May 13-17, 2018

Dear Friends:

I am pleased to extend my warmest greetings to everyone attending the International Conference on Pattern Recognition and Artificial Intelligence 2018, being held in Montreal, Quebec.

This conference brings together industry professionals from around the world to share the results of their research and discuss the latest trends and issues. I am certain that delegates will be inspired by the presentations and will benefit from this opportunity to network with their peers.

I would like to thank the organizers for putting together an informative and stimulating program for everyone in attendance. I hope that participants will also take advantage of this opportunity to explore everything that Montreal has to offer.

Please accept my best wishes for a productive conference.

Sincerely,

The Rt. Hon. Justin P.J. Trudeau, P.C., M.P.
Prime Minister of Canada
Welcome to the International Conference on Pattern Recognition and Artificial Intelligence ICPRAI-2018. This conference has been organized to celebrate the 30th Anniversary of CENPARMI (Centre for Pattern Recognition and Machine Intelligence) at Concordia University, Montreal, Canada, with the co-operation of all 4 universities in the city. The organization committee consists of well-known experts from all 6 continents of the world with a dream team of 9 world renowned Keynote Speakers to cover the PRAI frontiers in depth and in breadth with insights. The program includes 160 presentations in 4 tracks related to handwriting and document analysis, computer vision and image processing, machine learning and deep networks, biometrics and medical imaging, forensic science and video surveillance, classification techniques and various applications. In addition to a free public lecture, 3 satellite workshops have been organized focusing on image data mining, lognormality, e-learning and teaching methods.

Pattern Recognition and AI techniques and systems have been applied successfully to solve practical problems in many domains. In co-operation with CENPARMI members and colleagues at other institutions in Montreal, ICPRAI 2018 is co-sponsored by numerous organizations and industrial companies. It brings together a large number of scientists from all over the world to express their innovative ideas and report on their latest findings, as well as the fruitful results of numerous CENPARMI members and their students in handwriting recognition, forensic studies, face recognition, computer graphics, medical imaging, and classification techniques. In addition to the rich technical program, our conference also features practical sessions to encourage author-editor and inventor-investor interactions, and social events for CENPARMI alumni and conference participants.

For all the above activities, we would like to thank the co-operation of Conference Co-Chairs Xiaoyi Jiang, Seongwhan Lee, and Sabine Bergler, Track Co-Chairs Adam Krzyzak, Neamat El Gayar, Simone Marinai, and P. C. Yuen, Workshop Co-Chairs Renzhong Wang, Igor Gurevich, Vera Yashina, and Rejean Plamondon, Special Session Chairs Masaki Nakagawa and Nicole Vincent, publication and publicity chairs and their committee members, and our experts Peter Grogono, Leila Kosseim, Muna Khayyat, and Didier Guillevic, for a superb technical program and conference proceedings.

We would like to express our gratitude to the numerous committee members for taking care of financial and sponsorship matters (Olga Ormandjieva and Jun Tan). Special thanks go to Local Arrangements Chair Marleah Blom who takes care of the conference and hotel venues, and Nicola Nobile who has dedicated himself to a great variety of duties, ranging from conference secretariat businesses, to publications, software for handling paper submissions, reviews and selections, plus numerous other chores. They, together with Mary Tzanetakos, have created a very warm and comfortable environment to work in.
Thanks are also due to the organizations listed in the proceedings and those of the organizers of ICPRAI-2018, and the administration of Concordia University, which have given us strong support in finance and in various forms and means.

Finally, I hope you will find this conference a rewarding and memorable experience. We wish you a very enjoyable stay in beautiful Montreal and other parts of Canada.

Ching Y. Suen

General Chair, http://www.icprai2018.com
Director of CENPARMI, Concordia University
Montreal, Canada. http://www.cenparmi.concordia.ca

May 2018
À l’occasion de cette conférence internationale, je tiens à souligner l’apport significatif du Centre sur la reconnaissance des formes et l’intelligence artificielle de l’Université de Concordia dans l’essor de la recherche au Québec dans ces secteurs de pointe. L’intelligence artificielle détient d’ailleurs de profondes racines dans les universités québécoises, grâce auxquelles le Québec et sa métropole se sont vu reconnaître un leadership international en la matière.

Pour saisir pleinement les occasions liées au développement de l’intelligence artificielle, nous devons être proactifs et faire preuve d’audace. À ce titre, une véritable grappe industrielle se forme actuellement dans ce domaine au Québec, tant avec l’implantation, à Montréal, de laboratoires de géants de l’informatique tels que Google et Microsoft, qu’avec l’adoption de technologies novatrices par des entreprises d’ici.

Dans le cadre de la Stratégie québécoise de la recherche et de l’innovation, le Gouvernement du Québec a d’ailleurs annoncé des investissements de 100 millions de dollars d’ici les cinq prochaines années pour soutenir la création de la grappe. Celle-ci sera chargée de former, d’attirer et de retenir les meilleurs talents de partout dans le monde et de créer un environnement d’affaires favorisant l’essor de startups et l’intégration de l’intelligence artificielle par les entreprises et les organismes québécois. De plus, nous avons l’intention de créer sous peu une organisation mondiale en intelligence artificielle à Montréal, dont l’objectif sera de partager les connaissances et les réflexions éthiques sur ce sujet.

Nous comptons sur l’appui de nos précieux partenaires, dont le Centre sur la reconnaissance des formes et l’intelligence artificielle de l’Université de Concordia, pour contribuer au rayonnement de l’expertise du Québec et de sa métropole en la matière, grâce entre autres à des événements stimulants comme celui auquel vous êtes conviés cette semaine.

Je vous souhaite une excellente conférence à toutes et à tous !

Philippe Couillard

A MESSAGE FROM THE PREMIER

On the occasion of this international conference, I wish to acknowledge the significant contribution of Concordia University’s Centre for Pattern Recognition and Machine Intelligence to the growth of research in Quebec in these leading-edge sectors. Indeed, artificial intelligence has deep roots in Quebec universities, as a result of which Quebec and Montreal have gained international recognition for leadership in this area.

To take full advantage of the opportunities linked with the development of artificial intelligence, we must be proactive and act boldly. Along these lines, a true industrial cluster is currently taking shape in Quebec in this field, both with the installation of laboratories in Montreal by giants of information technology such as Google and Microsoft, and with the adoption of innovative technologies by firms from here.

Also, in the framework of the Quebec research and innovation strategy, the Gouvernement du Quebec has announced $100 million in investments over the next five years to support the creation of this cluster. Its role will be to train, attract and retain the best talents from around the world and to create a business environment fostering start-ups and artificial intelligence integration by Quebec firms and organizations. In addition, we intend to create a world organization of artificial intelligence in Montreal, with the goal of sharing knowledge and ethical thinking on this subject.

We count on the support of our valuable partners, including Concordia University’s Centre for Pattern Recognition and Machine Intelligence to contribute to broadening the expertise of Quebec and of Montréal in this field, owing, among other things, to stimulating events such as the one to which you have been invited this week.

I take this opportunity to congratulate the organizers, the volunteers and the partners, who make this experience accessible to all and thus contribute to the outreach of culture in all its nuances.

I wish you all a pleasant festival!
Mot de la mairesse / Message from the mayor – Valérie Plante

Je souhaite la plus chaleureuse des bienvenues à tous les participants et participantes à la Conférence internationale sur la reconnaissance des formes et l'intelligence artificielle.

Pour souligner son 30e anniversaire, le Centre d’études en reconnaissance des formes et en intelligence artificielle (CENPARMI) a rassemblé des conférenciers de renommée mondiale et offrira plus d’une centaine de présentations techniques qui sauront attirer l’attention tant du public que des experts sur les avancées en matière de techniques de pointe. Véritable plateforme d’échange et de partage, cette conférence regroupe des chercheurs internationaux armés d’expertises diverses qui sauront favoriser une grande collaboration. Cette collaboration est essentielle pour l’avancement de la science et de la technologie, une priorité chère à notre métropole.

Montréal est une ville de savoir, une ville d’innovation et une ville à l’avant-garde dans le domaine de l’intelligence artificielle. Comme administration publique, il nous importe de nous assurer que des formations à la fine pointe de la technologie soient disponibles et qu’elles correspondent aux réalités du marché du travail. À cet égard, un nouveau Bureau de l’enseignement supérieur a été créé par la Ville de Montréal. Ce bureau fera le pont entre les universités, les établissements d’enseignement supérieur et la communauté, afin d’assurer que les formations offertes répondent aux besoins de main d’œuvre de Montréal, chef de file mondial en intelligence artificielle.

Nous misons dès aujourd’hui sur notre savoir et cherchons à développer nos talents et nos compétences dans les domaines de l’intelligence artificielle, de la reconnaissance des formes et de l’analyse des données massives qui façonneront notre quotidien. Le réseau montréalais de chercheurs, d’universités, de centres de recherche, d’entreprises et d’organismes à but non lucratif favorise l’innovation et l’entrepreneuriat dans ces domaines. La sphère de l’enseignement supérieur est directement liée à l’épanouissement de cet écosystème à Montréal. À la fois utilisatrice de technologies et collaboratrice dans de multiples projets, la Ville de Montréal fait intégralement partie de cet écosystème.

Nous sommes audacieux et créatifs à Montréal et nous avons la capacité de penser autrement. Nous avons tout ce qu’il faut pour devenir une ville plus intelligente, plus simple et plus efficace.

Je remercie sincèrement CENPARMI et les organisateurs de cette conférence pour leur travail, qui saura renforcer les relations entre les institutions universitaires et les centres de recherches internationaux. Je vous souhaite une conférence riche en échanges et en créativité.

Ensemble, continuons ce bon travail!

I would like to extend a warm welcome to all the participants to the International Conference on Pattern Recognition and Artificial Intelligence.

In highlighting its 30th anniversary, the Centre for Pattern Recognition and Machine Intelligence (CENPARMI) has invited world-renowned speakers and will offer more than one hundred technical presentations geared both towards the general public, and towards experts in the area of cutting-edge technologies. Providing a platform for discussion and to share information, this conference will be a point of convergence for international researchers whose expertise will provide the grounds for future collaboration. This collaboration is essential for the advancement of science and technology, a priority that our metropolis holds dear.
Montréal is a city of knowledge and innovation and is at the forefront of the field of artificial intelligence. As a public administration, we wish to provide state-of-the-art training adapted to the realities of today's labor market. For this purpose, the city created its very own higher education bureau (Bureau de l'enseignement supérieur). The bureau will act as a bridge between universities, higher education institutions and the community, in order to ensure that the training provided meets the needs of Montréal's workforce, in keeping with our city's standing as a world leader in the field of artificial intelligence.

We rely on our knowledge and seek to develop our talents and competencies with respect to artificial intelligence, pattern recognition and the analysis of mass data, which are bound to impact our daily lives. Our city's network of researchers, universities, research centers, corporations and non-profit organizations promotes innovation and entrepreneurship in these fields. Higher education is directly linked to this flourishing ecosystem in Montréal. Both as a user of high technology and as a partner in the development of numerous projects, Montréal also forms an integral part of this ecosystem.

In Montréal, we are daring and creative and we know how to think outside the box. We have what it takes to become a smarter, simpler, more effective city.

We would like to sincerely thank the CENPARMI as well as all organizers of this conference for their work, as they will have contributed to reinforcing relationships between universities and international research centers. I wish you a conference filled with rewarding exchanges and creativity.

Together, let's continue the great work!

Valérie Plante
Mairesse de Montréal
Mayor of Montréal

Montréal
May 2018

Eva Nassif, Member of Parliament

It is my pleasure to welcome everyone to the 2018 International Conference on Pattern Recognition and Artificial Intelligence, and to congratulate the Centre for Pattern Recognition and Machine Intelligence of Concordia University for 30 years of innovation. As a Concordia University alumni, this great achievement attests to what we know to be true about this institution: it nurtures talent and hard work.

With at least 160 technical presentations and many great key speakers from around the world, this Conference is sure to be a success where there will be much to learn. I am proud to see such an important event taking place at our university campus, encouraging emerging and established academics and researchers in these disciplines, by offering a prominent platform for academics, knowledge, and breakthrough technology.

I want to congratulate the organizers, the key speakers, and everyone involved in this conference, for being a part of this fundamentally important intellectual experience.

Sincerely,

Eva Nassif, Member of Parliament

Vimy
As the Member of Parliament for Mount Royal, it gives me great pleasure to send my best wishes for a successful International Conference on Pattern Recognition and Artificial Intelligence. What better way to mark the 30th anniversary of the CENPARMI than by bringing people from all over the world together to discuss industry changes and developments.

It is a privilege to congratulate all the people who helped make this extraordinary event possible. As an alumnus from Concordia, I want to give a special thanks to Professor Ching Y. Suen and his team for co-ordinating this event along with the all the co-chairs who helped make this conference possible.

I wish you all many fruitful discussions and a wonderful event.

All the best,

Anthony
Dear participants,

It is with great pleasure that I welcome you all to the 2018 International Conference on Pattern Recognition and Artificial Intelligence, offered by the Centre for Pattern Recognition and Machine Intelligence.

The CENPARMI fosters collaboration in expanding our knowledge on important technologies that will shape the ways we think, interact and subsist in the future. Starting from a small group in 1988, it has since grown to bring together experts and professors from multiple universities such as l’École Polytechnique of Montreal and l’École de Technologie Supérieure. As a Member of Parliament, I am delighted to see such progress made in the interest of our social and economic well-being.

I would like to congratulate the CENPARMI on celebrating their 30th anniversary and thank you, the participants, for making this conference possible with 160 technical presentations by authors from 32 countries. Given the wealth of knowledge available, I encourage all of you to exercise your curiosity to the fullest extent.

From May 14th to May 17th, you will get to share in the work of some of the most brilliant minds in the Canadian and international scientific communities. I wish you all the best and hope you enjoy this amazing opportunity!

Sincerely,

David Lametti
Member of Parliament - LaSalle-Émard-Verdun
I am pleased to welcome all of the participants of the International Conference on Pattern Recognition and Artificial Intelligence to Montreal. You could not have chosen a better location for this conference as Canadians, and particularly Montrealers, continue to lead the way in the field of Artificial Intelligence.

The ideas discussed during this conference have the potential to transform the Canadian economy and improve the quality of life of Canadians. That's why our government is proud to support this exciting and rapidly growing field.

Your contributions to this field help build a bolder, brighter future for all Canadians. I wish all participants a successful and productive conference.

Marc Miller
Member of Parliament
Ville-Marie—Le Sud-Ouest—Île-des-Sœurs

3175 Saint-Jacques
Montreal, Quebec
H4C 1G7

514-496-4885
marc.miller@parl.gc.ca

mmiller.liberal.ca
Montreal, March 2nd, 2018

Professor Ching Y. Suen
Director
CENPARMI (Centre for Pattern Recognition and Machine Intelligence)
1455, boulevard de Maisonneuve Ouest, Suite EV003.403
Montréal (QC) H3G 1M8

Professor Suen,

On behalf of the MNA for the Mont-Royal riding and Minister responsible for Government Administration and Ongoing Program Review, and chair of the Conseil du trésor, Mr. Pierre Arcand, we acknowledge receipt of your letter dated February 15th regarding the International Conference on Pattern Recognition and Artificial Intelligence, organized by your center CENPARMI next May.

We would like to congratulate you for this promising international event.

You may rest assured that your request will be forwarded to Minister Arcand, and we are pleased to make available your brochures to the citizens of the Mont-Royal riding.

Best regards,

Charlotte Thierry
Political attachée
Mont-Royal riding office
Pierre Arcand  
Député de Mont-Royal  
Ministre responsable de l'Administration gouvernementale  
et de la Révision permanente des programmes  
Président du Conseil du trésor  
Ministre responsable de la région de la Côte-Nord

Montréal, le 2 mars, 2018

Dr. Ching Y. Suen  
Directeur général  
CENPARI (Centre for Pattern Recognition and Machine Intelligence)  
1455, boulevard de Maisonneuve Ouest, Suite EV003.403  
Montréal (QC) H3G 1M8

Dr. Suen,

Au nom du député de Mont-Royal, ministre responsable de l'Administration gouvernementale et de la Révision permanente des programmes, et président du Conseil du trésor, M. Pierre Arcand, nous accusons réception de votre correspondance transmise le 15 février, par laquelle vous lui présentez la Conférence internationale sur la reconnaissance des formes et l'intelligence artificielle, qui se tiendra en mai prochain.

Nous vous en remercions, et vous assurons que le tout sera porté à la connaissance du Ministre Arcand et que vos brochures seront mises à la disposition des citoyens de la circonscription de Mont-Royal.

Nous vous prions d’agréer, Dr. Suen, l’expression de nos sincères salutations.

Charlotte Thierry  
Attachée politique  
Circonscription de Mont-Royal
MESSAGE DU CHEF DU PARTI QUÉBÉCOIS

Je suis très heureux de vous souhaiter la bienvenue à l'édition 2018 de la prestigieuse International Conference on Pattern Recognition and Artificial Intelligence.

Le Parti Québécois a toujours valorisé et encouragé l’innovation et l’avancement technologique. Grâce aux politiques que nous avons mises en place, le Québec a vu émerger une grappe technologique de calibre mondial, notamment dans les domaines du multimédia et du développement des affaires électroniques.

La même vision doit s'appliquer à la reconnaissance de formes et à l'intelligence artificielle : il est fondamental que le Québec affirme sa présence, se présente en leader, ait l’audace d’explorer au-delà de ce qui est connu.

Puisque la technologie constitue désormais, et pour longtemps, un aspect incontournable de nos vies, cultivons la vision d'un Québec chef de file des secteurs de pointe. Nous avons tout l’aplomb, toute l’ingéniosité, tout le talent qu’il faut!

Bonne conférence à toutes et à tous et, aux gens de passage, un excellent séjour parmi nous!

Jean-François Lisée

MESSAGE FROM THE LEADER OF THE PARTI QUÉBÉCOIS

I am very pleased to welcome you to the 2018 edition of the prestigious International Conference on Pattern Recognition and Artificial Intelligence.

The Parti Québécois has always valued and encouraged innovation and technological advancement. Thanks to the policies we have put in place, Quebec has witnessed the emergence of a world-class technology cluster, particularly in the areas of multimedia and e-business development.

The same vision must apply to form recognition and artificial intelligence: it is fundamental that Quebec asserts its presence, introduces itself as a leader, with the audacity to explore beyond what is known.

Since technology is now, and for times to come, an essential aspect of our lives, let us cultivate the vision of a leading Quebec in high-tech sectors. We have all the confidence, all the ingenuity, all the talent required!

A great conference to all and, to the ones visiting, an excellent stay among us!
ICPRAI Organizers

General Chair
Ching Y. Suen (Canada)

Conference Co-Chairs
Sabine Bergler (Canada)
Xiaoyi Jiang (Germany)
Seong-Whan Lee (Korea)

Program Chairs
Adam Krzyzak (Canada)
Neamat El Gayar (Egypt)
Simone Marinai (Italy)
P. C. Yuen (HK, China)

Special Sessions Chairs
Réjean Plamondon (Canada)
Masaki Nakagawa (Japan)
Nicole Vincent (France)

Publication Chairs
Louisa Lam (Canada)
Patrick S. P. Wang (U.S.A.)

Publicity Chairs
Mohamed Cheriet (Canada)
Gernot A. Fink (Germany)
Luiz Oliveira (Brazil)
Umapada Pal (India)
Yuan Y. Tang (MO, China)
Jun Zhou (Australia)

Sponsorship Chairs
Robert Sabourin (Canada)
Jun Tan (China)

Competition Judges
Xiayi Jiang
Mohamed Cheriet
Donato Impedovo

Local Arrangements
Marleah Blom
Nicola Nobile
Mary Tzanetakos
Olga Ormandjieva (Finance)
Sponsors

Sponsored by CENPARMI, CSE & ENCS of Concordia University, and Industrial Partners.

Endorsed by: TC3 Neural Networks of IAPR (Int. Assoc. for PR)
ICPRAI Keynote Speakers

ICPRAI 2018 is proud to have had the following Keynote Speakers:

<table>
<thead>
<tr>
<th>Speaker</th>
<th>Topic</th>
<th>Home Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dr. Edwin Hancock</td>
<td>Network Analysis in Finance and Medicine</td>
<td><a href="http://videolectures.net/edwin_hancock/">http://videolectures.net/edwin_hancock/</a></td>
</tr>
<tr>
<td>U. K.</td>
<td>Editor-in-Chief - Pattern Recognition (Elsevier), Chair Professor of Computer Vision, York Univ., UK.</td>
<td></td>
</tr>
<tr>
<td>Dr. Diana Inkpen</td>
<td>Detecting Signs of Mental Illness from Social Media</td>
<td><a href="http://www.site.uottawa.ca/~diana/">http://www.site.uottawa.ca/~diana/</a></td>
</tr>
<tr>
<td>Canada</td>
<td>Director of the Natural Language Processing Lab, Univ. of Ottawa, Canada.</td>
<td></td>
</tr>
<tr>
<td>U.S.A.</td>
<td>Founder of The Henry C. Lee Institute of Forensic Science, Director of Forensic Research and Training Center, and Distinguished Chair Professor in Forensic Science of the Univ. of New Haven, USA.</td>
<td></td>
</tr>
</tbody>
</table>
**Descriptive Approach to Image Analysis**

**Dr. Igor B. Gurevich:** Deputy Editor-in-Chief of the English international journal "Pattern Recognition and Image Analysis (Pleiades/Springer)" of the Russian Academy of Sciences (RAS), and Vice-Chairman of the National Committee for PRIA of the Presidium of the RAS.

**Dr. Vera V. Yashina:** Lead Researcher of the Department of "Mathematical and Applied Problems of Image Analysis" at the Federal Research Centre, RAS.

---

**Improving Health-Care Through AI: Challenges and Opportunities for Reinforcement Learning**

**Dr. Joelle Pineau**

Canada

Director of the AI Lab of Facebook in Montreal
Co-Director of the Reasoning and Learning Lab, McGill Univ., Canada.


---

**How PRAI is changing the world - from Biometrics to Driverless Transportation**

**Dr. Patrick S. P. Wang**

U.S.A.

Founding Editor-in-Chief of "IJPRAI" (World Scientific) since 1987, former Adjunct Faculty at MIT and Harvard, and Professor Emeritus, Northeastern University, USA.

Home Page: [https://sites.google.com/site/mozart200/](https://sites.google.com/site/mozart200/)

---

**Steps Toward Humanlike Intelligence and Self-aware Systems**

**Dr. Patrick H. Winston**

U.S.A.

Ford Professor of Artificial Intelligence and Computer Science, Massachusetts Institute of Technology (MIT), author of numerous books on AI & programming languages, and former Director of the AI Lab of MIT for 25 years, U.S.A.

How Technologies for Mapping the Brain Could Lead to New Kinds of Artificial Intelligence

Y. Eva Tan Professor in Neurotechnology at MIT and Co-Director, MIT Center for Neurobiological Engineering

A recipient of the "Break-through Prize in life sciences", and numerous other awards. He has contributed to over 100 peer-reviewed papers and 170 granted patents, and given over 400 invited talks.

Home Page: http://syntheticneurobiology.org/people/display/71/11
On the Descriptive Approach to Image Analysis

I.B. Gurevich, V.V.Yashina

Department of Mathematical and Applied Problems of Image Analysis
Federal Research Center “Informatics and Control” of the Russian Academy of Sciences
Moscow, Russian Federation

igourevi@ccas.ru, weravashina@gmail.com

Abstract— The paper is devoted to Descriptive Image Analysis (DA) - an important line of the modern mathematical theory of image analysis. DA is a logically organized set of descriptive methods and models aimed for analyzing and estimating the information represented in the form of images, as well as for automating the extraction from images of knowledge and data needed for intelligent decision making about the real-world scenes reflected and represented by the images under analysis. The basic idea of DA consists in reducing all processes of analysis (processing, recognition, understanding) of images to: 1) construction of models/representations/formalized descriptions of images; 2) definition of transformations over image models; 3) construction of models/representations/formalized descriptions of transformations over models and representations of images; 4) construction of models/representations/formalized descriptions of schemes of transformations over models and representations of images that provide a solution for an image analysis problem. The main fundamental sources that predetermined the development of DA, or had a significant influence thereon, are considered. In addition, a brief description of the current state of descriptive image analysis is presented that reflects the main results of the descriptive approach (DA) to analysis and understanding of images, which is proposed and developed by the authors of this paper [13–15, 23, 26, 30].

Automation of image processing, analysis, estimation, recognition, and understanding is one of the crucial points of theoretical computer science that has a decisive importance for applications, in particular, for diversification and extension of types of solvable applied problem, as well as for improving the efficiency of problem solving and the precision of problem solutions.

The theoretical and methodical basis of automating the processing, analysis, and estimation of experimental data is constructed by the mathematical theory of pattern recognition and mathematical theory of image analysis.

Pattern recognition as a science has evolved from the needs to solve problems of processing, analyzing, and evaluating ill-structured, badly formalized, indistinct, incomplete, contradictory, semantically saturated, and noisy information by using computationally efficient mathematical methods. The initial information in these problems consists of numbers, symbols, and expert data; images; speech; arbitrary signals; texts; documents; diagrams and drawings; and random combinations thereof.

Pattern recognition tools and methods are designed for solving applied problems of intelligent decision making, diagnostics, identification, and prediction. The role of the image as an object of analysis and estimation is determined by its specific and integral information properties. An image is a certain totality of the initial pictured data and means of its representation, the results of the formation processes of representations of the image and their transformation procedures, the physical and logical aspects, and the models of objects, events, and processes represented in the image.

I. INTRODUCTION

The paper is devoted to Descriptive Image Analysis (DA) - an important line of the modern mathematical theory of image analysis. DA is a logically organized set of descriptive methods and models aimed for analyzing and estimating the information represented in the form of images, as well as for automating the extraction from images of knowledge and data needed for intelligent decision making about the real-world scenes reflected and represented by the images under analysis. The basic idea of DA consists in reducing all processes of analysis (processing, recognition, understanding) of images to: 1) construction of models/representations/formalized descriptions of images; 2) definition of transformations over image models; 3) construction of models/representations/formalized descriptions of transformations over models and representations of images; 4) construction of models/representations/formalized descriptions of schemes of transformations over models and representations of images that provide a solution for an image analysis problem.

The main fundamental sources that predetermined the origination and development of descriptive image analysis, or had a significant influence thereon, are considered. In addition, a brief description of the current state of descriptive image analysis is presented that reflects the main results of the descriptive approach (DA) to analysis and understanding of images, which is proposed and developed by the authors of this paper [13–15, 23, 26, 30].

Automation of image processing, analysis, estimation, recognition, and understanding is one of the crucial points of theoretical computer science that has a decisive importance for applications, in particular, for diversification and extension of types of solvable applied problem, as well as for improving the efficiency of problem solving and the precision of problem solutions.

The theoretical and methodical basis of automating the processing, analysis, and estimation of experimental data is constructed by the mathematical theory of pattern recognition and mathematical theory of image analysis.

Pattern recognition as a science has evolved from the needs to solve problems of processing, analyzing, and evaluating ill-structured, badly formalized, indistinct, incomplete, contradictory, semantically saturated, and noisy information by using computationally efficient mathematical methods. The initial information in these problems consists of numbers, symbols, and expert data; images; speech; arbitrary signals; texts; documents; diagrams and drawings; and random combinations thereof.

Pattern recognition tools and methods are designed for solving applied problems of intelligent decision making, diagnostics, identification, and prediction. The role of the image as an object of analysis and estimation is determined by its specific and integral information properties. An image is a certain totality of the initial pictured data and means of its representation, the results of the formation processes of representations of the image and their transformation procedures, the physical and logical aspects, and the models of objects, events, and processes represented in the image.

This work was supported in part by the Russian Foundation for Basic Research (projects no. 17-07-01482)
The specifics and complexity of image analysis and evaluation are due to the need for achieving a certain balance between such contradictory factors as the goals and problems of analysis; the nature of visual perception; the methods and tools for obtaining, forming, and representing images; and mathematical, computational, and technological tools of image analysis.

By now, image analysis and estimation has accrued much experience in applying mathematical methods from various areas of mathematics, computer science, and physics, in particular, algebra, geometry, discrete mathematics, mathematical logic, probability theory, mathematical statistics, mathematical analysis, mathematical theories of pattern recognition and image analysis, digital signal processing, and optics.

On the other hand, with all this diversity of methods used, we still need a firm basis to arrange and choose suitable methods and models of image analysis, represent, in an unified way, data for processing (images), meet the requirements of standard recognition algorithms imposed on initial information, construct mathematical models of images designed for recognition problems, and, on the whole, establish a universal language for unified description and representation of images and transformations over them.

Image analysis and understanding are quite difficult problems for mathematicians because the image is an extremely inconvenient form of information representation for mathematical processing. For a long time, this direction was underdeveloped because mathematicians had no interest in working with such unconventional forms of information. Serious efforts began in the 1950s and have been continuing since.

Presently, there is a wide set of mathematical methods allowing one to reduce the image to a form suitable for efficient recognition algorithms. Such formalized image representations (models: images reduced to a recognizable form) form the necessary basis for simulation, recognition, and computation of image characteristics, detection of regularities and properties, and intelligent decision making. These methods have generally been substantiated, developed, and many times tested in practice. In this respect, it is quite correct to speak about the creation of a mathematical theory of image analysis and a certain degree of its maturity; however, it still has to achieve the degree of development of the mathematical theory of image recognition, the essential elements of which are used when working with images.

The Russian school of mathematics has made an extremely significant contribution to the mathematical theories of image recognition and analysis. Its results, particularly, in the field of algebraic methods of image recognition and analysis, define its global level today [28, 68].

The Russian mathematical school also obtained some important original results on other algebraic tools for pattern recognition and image analysis, including algebraic multiple classifiers, algebraic committees of algorithms, combinatorial algorithms for recognition of 2D data, descriptive image models, and 2D formal grammars. The materials on image algebras (IAs) present the main approaches to creating a unified language for concepts and operations used in image processing and analysis. The most famous IAs are the standard IA by G. Ritter and the descriptive IA (DIA) by I. Gurevich.

II. THE DESCRIPTIVE APPROACH TO IMAGE ANALYSIS AND UNDERSTANDING

In this Section we briefly outline the DA, specifically: 1) its mathematical foundations and basic elements; 2) the model of the process of image recognition that is fundamental for the DA; 3) the sources of introducing the descriptive nature into mathematical methods for image analysis. We discuss the goals of the theoretical development in the framework of the DA, image analysis algebraization, and the problem of an image reduction to a recognizable form (IRRF). The main results of the DA are outlined.

The current state and development trends of DA are determined by the methods, models, and results of the DA obtained during its elaboration [13–30].

This section presents a methodology, as well as mathematical and computational techniques, for automation of image mining based on the DA.

The creation of the DA was significantly influenced by the following basic mathematical theories of pattern recognition: 1) the Algebraic Approach to Pattern Recognition and Algebra of Algorithms by Y.Zhuravlev [68]; 2) the Pattern Theory by U.Grenander [12], particularly, algebraic methods for representation of initial data in pattern recognition problems; 3) theory of categories techniques in pattern recognition (M. Pavel) [50]; 4) the Computational Theory of Vision by D.Marr [46].

Pattern Theory (U. Grenander [12]) provides techniques for pattern recognition for a case of data representation and transformation based on regular combinatorial structures and algebraic and probabilistic tools.

The Algebraic Approach to Recognition, Classification, and Forecasting Problems [68] includes mathematical setting of a pattern recognition problem, correctness and regularity conditions, algebras of algorithms, multiple classifiers (representation of a pattern recognition algorithm as an algebraic polynomial).

Aside from the basic researches by Y.Zhuravlev’s scientific school, there are a significant number of results concerned with algebraic methods for analysis and estimation of the information represented as images and signals, particularly, V.Labunetc [38], Y.Pytiev[51], I.Sinicyn [58], J.Furman [9, 10], V.Chernov [3]. The Soviet mathematical school obtained some important original results on other algebraic tools for pattern recognition and image analysis (for example, M.Schlesinger’s work (2D formal grammars) [60].

We should mention as the sources of the descriptive view on image analysis and consequently of DA the works of 1970 - 1990 years by F.Ambl er, G.Barrow, R Burstall [1], T.Evans [8], S.Kaneff [35], R.Kirsh [36], R.Narasimhan [48], A.Rosenfeld [56], A.Shaw [59].

Carried out by pattern recognition methods, image mining now tends to use multiplicity (multiple classifiers, and multi-
model representations) and fusion of the results. It means that several different algorithms are applied in parallel to process the same model and several different models of the same initial data in order to solve a problem; then, the results are fused to obtain the most accurate or optimal solution.

Multiple classifiers as well as multi-model and multiple-aspect image representations, are the common tools for implementing this multiplicity and fusion techniques. For the first time, the idea of a combination of classifiers optimized by algebraic correction was suggested and justified by Y. Zhuravlev [68].

In the English-language literature the multiple classifiers are very popular [52].

The idea of constructing a unified language for the concepts and operations used in image processing appeared for the first time in the works of Unger [66], who proposed to parallelize algorithms for image processing and analysis on computers with a cellular architecture. Mathematical morphology developed by Materon and Serra [43, 57] became a starting point of a new mathematical wave in image processing and analysis.

Serra and Sternberg [62, 63] were the first to succeed in constructing an integrated algebraic theory of image processing and analysis on the base of mathematical morphology. It is believed [54] that Sternberg was the one to introduce the term “image algebra” [62] in its current standard meaning.

There are many works devoted to the development of specialized algebraic constructions implementing or improving the methods of mathematical morphology.

From that time until the 1990s the interest in the descriptive and algebraic aspects of image analysis declined. The final view of the idea of the IA was the standard IA by G. Ritter [54, 55] (algebraic representation of the image analysis and processing operations). The DIA is created as a new kind of IA that provides a possibility to work with main image models and with the basic models of the transform procedure. The DIA was introduced by I. Gurevich and is now being developed by him and his followers [18, 19, 21, 24, 27].

The milestones of algebraization is as follows: 1) von Neumann [49] and Unger [66] (studies of interactive image transformations in a cellular space); 2) Duff, Watson, Fountain, and Shaw [5] (a cellular logic array for image processing); 3) Rosenfeld [56] (digital topology); 4) Minkowski and Hadwiger [31, 47] (pixel neighborhood arithmetic and mathematical morphology); 5) Matheron, Serra, Stermburg [43, 57, 62, 63] (a coherent algebraic theory specifically designed for image processing and image analysis: mathematical morphology); 6) Zhuravlev [68] (algebra of algorithms); 7) Stermburg [62] (the first use of the term “image algebra”); 8) Grenander [12] (Pattern Theory); 9) Maragos [44, 45] (introduced a new theory unifying a large class of linear and nonlinear systems under the theory of mathematical morphology); 10) Pavel [50] (theory of categories techniques in pattern recognition); 11) Davidson [4] (completed the mathematical foundation of mathematical morphology by embedding it into the lattice algebra (minimax algebra)); 12) Ritter [54, 55] (IA); 13) Gurevich [18, 19, 21, 24, 27] (DIA); 14) Haralick [32-34, 39], Shapiro [32-34, 39], Lee [33, 39], Joo Hyoman [34], Schafer [45], Dougherty [6, 7], Sinha [6], Gader [11], Khabou [11], Koldobsky [11], Radunacu [53], Grana [53], Albizuri [53], Sussner [64, 65], and Soille [61] (development of mathematical morphology and image algebras [37]).

When analyzing the existing algebraic apparatus [2, 40, 41, 42, 67], we came to the formulation of the following requirements on the algebraic language designed to describe the algorithms for solving the problems of image processing and understanding [18]: a) the new algebra must enable the processing of images as objects of analysis and recognition; b) the new algebra must enable operations on image models, i.e., arbitrary formal representations of images, which are objects and, sometimes, results of analysis and recognition; the introduction of image models is a necessary step in formalizing the initial data of algorithms; c) the new algebra must enable operations on the main models of procedures for image transformations; d) the procedures for image modifications may be used both as operations of the new algebra and as its operands to construct compositions of the basic models of the procedures.

During the development and implementation of the DA, a new class of image algebras—DIA—was introduced, defined, and investigated; the main types of image models were introduced, classified, and investigated; axioms of the DA were introduced; the general model of the image recognition process was defined and investigated; new settings of image analysis and recognition problems were introduced; a concept of image equivalence in the recognition problem was introduced and investigated; new classes of image recognition algorithms were defined and investigated; and an image formalization space was introduced, defined, and analyzed.

By the mid-1990s, it became apparent that, for the development of image analysis and recognition, it is crucial to: 1) understand the nature of initial information, i.e., images; 2) find the methods for image representation and description that make it possible to construct image models for recognition problems; 3) establish a mathematical language for unified description of image models and their transformations that make it possible to construct image models and solve recognition problems; 4) construct models to solve recognition problems in the form of standard algorithmic schemes that make it possible, in the general case, to transfer from an original image to its model and from the model to a sought-for solution.

The DA solves the fundamental problems of formalization and systematization of the methods and forms of information representation in the problems of image analysis, recognition, and understanding. In particular, the problems arise in connection with the automation of information extraction from images for intelligent decision making (diagnostics, prediction, detection, evaluation, and identification of patterns).

The final goal of this research is automated image mining: 1) the automation of the development, testing, and adaptation of methods and algorithms for analysis and evaluation of images; 2) the automation of the selection of methods and algorithms for analysis and evaluation of images; 3) the automation of the evaluation of quality and adequacy of initial data to solve the image recognition problem; 4) the
development of standard technological schemes for detecting, assessing, understanding, and retrieving images.

The axiomatics and formal structures of the DA provide methods and tools to represent and describe images for their subsequent analysis and evaluation.

It is established that the overall success and effectiveness of the analysis and evaluation of the information represented in the form of images are determined by the capabilities of the IRRF.

The DA requires that the processes of analyzing and evaluating the information represented in the form of images (the trajectory of problem solving), as a whole, be viewed as a sequence/combination of transformations and computations of a set of intermediate and final (defining the solution) evaluations. These transformations are defined on the equivalence classes of images and their representations. The latter are defined descriptively, i.e., by using the basic set of prototypes and the corresponding generative transformations that are functionally complete with respect to the equivalence class of admissible transformations.

The main intention of the DA is to structure different techniques, operations, and representations used in image analysis and recognition, the conceptual and mathematical bases for representing and describing images, as well as for its analysis and for its estimation.

The analysis of a problem is based on the investigation of the inner structure and content of an image as a result of the procedures “constructing” it from its primitives, objects, descriptors, features, tokens, and relations between them.

The main purpose of the DA is to structure and standardize the variety of methods, processes, and concepts used for analysis and recognition of images. The DA, its axiomatics and formal constructions are proposed and developed as conceptual and logical bases for extraction of information from images. It includes the following basic tools for image analysis and recognition: a set of methods for analysis and recognition of images, IRRF techniques, a conceptual system of image analysis and recognition, descriptive image models [23-26], classes, a DIA language [18, 19, 21, 24, 27], settings of image analysis and recognition problems, and a basic model of image recognition.

The main areas of research within the DA: 1) the creation of axiomatics for analysis and recognition of images; 2) the development and implementation of a common language to describe the processes of analysis and recognition of images (the study of the DIA); 3) the introduction of formal systems based on certain regular structures to determine the processes of analysis and recognition of images.

The mathematical foundations of the DA: 1) the algebraization of information extraction from images; 2) the specialization of Zhuravlev’s algebra [68] to the case of representation of the recognition source data in the form of images; 3) a standard language for describing the procedures for analysis and recognition of images (DIA) [18, 19, 21, 24, 27]; 4) the mathematical formulation of the problem of image recognition; 5) mathematical theories of image analysis and pattern recognition; 6) a model of the process for solving the standard problem of image recognition.

The main objects and means of the DA: 1) images; 2) a universal language (DIA) [18, 19, 21, 24, 27]; 3) two types of descriptive models, namely, an image model [23-26] and a model for an image recognition problem solving procedures and their implementation [23-26]; 4) descriptive algebraic schemes of image representation [19]; and 5) multimodel and multiaspect representations of images that are based on generating descriptive trees [20].

The basic methodological principles of the DA: 1) the algebraization of image analysis; 2) the standardization of the representation of problems of analysis and recognition of images; 3) the conceptualization and formalization of the phases of transforming the image while solving the recognition problem; 4) the classification and specification of admissible models of images (descriptive image models); 5) the IRRF; 6) the use of the standard algebraic language of the DIA for describing models of images and procedures for their construction and transformation; 7) the combination of algorithms in multiple algorithmic schemes; 8) the use of multimodel and multiaspect representations of images; 9) the construction and use of the basic model of the solution process for the standard problem of image recognition; 10) the definition and use of a nonclassical mathematical theory for the recognition of new formulations of problems of analyzing and recognizing images.

The main objects and means of the DA: 1) images; 2) a universal language (DIA) [18, 19, 21, 24, 27]; 3) two types of descriptive models, namely, an image model [23-26] and a model for an image recognition problem solving procedures and their implementation [23-26]; 4) descriptive algebraic schemes of image representation [19]; and 5) multimodel and multiaspect representations of images that are based on generating descriptive trees [20].

The basic methodological principles of the DA: 1) the algebraization of image analysis; 2) the standardization of the representation of problems of analysis and recognition of images; 3) the conceptualization and formalization of the phases of transforming the image while solving the recognition problem; 4) the classification and specification of admissible models of images (descriptive image models); 5) the IRRF; 6) the use of the standard algebraic language of the DIA for describing models of images and procedures for their construction and transformation; 7) the combination of algorithms in multiple algorithmic schemes; 8) the use of multimodel and multiaspect representations of images; 9) the construction and use of the basic model of the solution process for the standard problem of image recognition; 10) the definition and use of a nonclassical mathematical theory for the recognition of new formulations of problems of analyzing and recognizing images.

Note that the construction and use of the mathematical and simulation models of objects and procedures used for their transformation is a widely accepted method of standardization in the applied mathematics and computer science.

By now, in the framework of the DA, the following main results were obtained.

1. Algebraization of image analysis: a) the DA was characterized; b) DA axioms were introduced and substantiated; c) a mathematical object “DIA” was introduced and defined; d) descriptive image algebras with one ring (DIA1R) were introduced and investigated; e) the definition, method, and necessary and sufficient conditions for the construction of DIA1Rs were proposed; f) specialized versions of DIA1R over images, over image models and over image transformations were defined; g) the set of operations for the standard 1A that enable the construction of DIA1Rs was defined; h) the DIA1R classes generating the classes of image models were defined.

2. Efficient methods and tools for the description and representation of images in recognition problems: a) new settings of an image recognition problem were introduced and substantiated; b) an image formalization space (IFS) was introduced, defined, and studied; c) the structure of an IFS was defined; d) the topological properties of an IFS were studied; e) a mathematical object “Descriptive Algorithmic Image Transformation Scheme” (DAS) was introduced, defined, and studied; f) a classification of DAS was constructed; g) the standardization of representation was carried out and some examples of DAS were constructed for solving applied problems of biomedical images analysis; h) a mathematical object “Image Representation” (IR) was introduced and defined; i) types of IR were introduced and defined; j) a mathematical object “Descriptive Image Model” (DIM) was introduced and defined.
introduced and defined; k) types of DIM were introduced, defined and investigated; l) a model for solving an image recognition problem based on DIM was proposed and implemented; m) a method for selection of image transformations that takes into account the information nature of images was proposed and implemented in software; n) criteria for classification of the features used for description of images were formulated; o) a classification of image features as a tool for formal description of an IR was constructed; p) a method for constructing multiple-aspect image descriptions based on classifications of image features was proposed.

3. Efficient algorithms based on estimates calculation for image recognition problems (AEC); a) a multilevel model for combining algorithms and initial data in image recognition that is based on the combined use of multi-algorithmic classifiers and dual IR of the type “combinatorial structures of local neighborhoods/formalized models” was introduced and substantiated; b) AEC-class accepting spatial data was defined and studied; c) AEC-class with square support sets was defined and dual IR of the type “combinatorial structures of local neighborhoods/formalized models” was introduced and implemented; d) a method for efficient implementation of AEC-class with systems of rectangular support sets and proximity functions on pairs of support sets were proposed; f) a method for constructing and estimating the computational complexity of AEC-class with systems of support sets generated by arbitrary templates that are based on multistep procedures for finding templates on the raster was proposed.

4. Linguistic and Knowledge-oriented Tools for Supporting the Automation of Image Analysis: a) an informational web resource (its concept, architecture, functional scheme) for processing, analysis, and recognition of images was proposed and implemented; b) a classification of problems of image processing, analysis, and recognition was constructed; c) “Automated Knowledge Base System for Processing, Analysis, and Recognition” of Images was developed; d) Information Retrieval Thesaurus for Image Analysis was developed; e) an experimental version of an image analysis ontology in the OWL language was developed;


III. CONCLUSIONS

The DA, as demonstrated by the results of its development and implementation, is a promising base for creating the Descriptive Theory of Image Analysis. The way to its creation involves research and development in the following directions. 1) Mathematical settings of new image recognition problem. 2) DIA. 3) DIM and image features classification. 4) IFS including its topologies, IR and problem solving trajectories. 5) DAS. 6) Generating descriptive trees and multiple models and representations of images. 7) Linguistic and knowledge-oriented tools. 8) Image equivalence. 9) Image metrics. 10) Pattern recognition algorithms accepting 2D and 3D data. 11) Combined use of the multiplicity of IR and the hierarchical model of multiple classifiers.

We hope that, after elaborating these topics we shall be able to formulate the axiomatic and basic statements of the Descriptive Theory of Image Analysis.

References


[57] I. N. Sinicyn, Calman and Pugachev Filters, 2nd ed. (Logos, Moscow, 2007) [in Russian].


ICPRAI Public Lecture

Artificial Intelligence and Pattern Recognition
Trendy Technologies in Our Modern Digital World

Public Lecture, Concordia University, Sunday May 13, 2018, 2:00 - 4:00 p.m.

The general public and ICPRAI conference participants were invited to Concordia University for a very special event that focused on AI, PR, Natural Language Processing, and Machine Learning (including deep learning). Through ample illustrations and live demonstrations, our team of experts unfolded the myths about these technologies in simple English, showed ways of transferring human intelligence to a computer, highlighted the state-of-the-art research, and provided answers to the following questions:

- What are they and how they have changed the world?
- How do they work and why they work?
- What are the successful applications and pitfalls?
- What are the challenges and limitations?

Speakers included experienced experts who are model professors and recognized winners of teaching and Research Awards, distinguished industrial practitioners, and the Vice-President of the Canadian Association of AI. As CENPARMI's gift to embrace the city of Montreal, the purpose of this public lecture was to inform, stimulate and excite individuals about these subjects while promoting the development of intelligent and learned citizens. The event was hosted in conjunction with the International Conference on Pattern Recognition and Artificial Intelligence at Concordia University, May 14-17, 2018 (http://www.icprai2018.com).

Convener: Dr. Ching Suen, Director of CENPARMI, Concordia University, Winner of numerous Teaching and Research Excellence Awards, gold medalist, Author/Editor of many books, journals, and technical papers on pattern recognition and machine intelligence, organizer of numerous international conferences and General Chair of ICPRAI 2018. http://www.cenparmi.concordia.ca/CENPARMI/Suen/index.html

Invited Speaker: Dr. Peter Grogono, Author of several books, including the best-selling textbook "Programming in Pascal". During his 38 years at Concordia University, he received teaching awards from the University, the Faculty, and Student Associations. https://users.enscs.concordia.ca/~grogono/

Invited Speaker: Dr. Leila Kosseim, Co-Director of the CLAC Lab and Vice President of the Canadian Artificial Intelligence Association, specialist in Artificial Intelligence, Natural Language Processing and Discourse Analysis, and author of numerous technical papers. https://users.enscs.concordia.ca/~kosseim/, and https://www.caiac.ca/en/executive

Invited Speaker: Dr. Muna Khayyat, Technology Associate, Morgan Stanley. Author of numerous technical papers on neural networks and pattern recognition, specialist in language models, hierarchical classifiers, word spotting and recognition of cursive scripts. https://www.linkedin.com/in/muna-khayyat-7a71a653/?ppe=1/

Table of Contents

Message from the Chair .......................................................................................................................... ii
Message du Premier Ministre Du Québec .......................................................................................... iv
Message from the Mayor of Montreal ............................................................................................... v
Message from Eva Nassif, Member of Parliament – Vimy .............................................................. vii
Message from Anthony Housefather, Member of Parliament – Mount Royal ................................. viii
Message from David Lametti, Member of Parliament - LaSalle-Emard-Verdun ............................... ix
Message from Marc Miller, Member of Parliament - Ville-Marie-Le Sud -Ouest-Ile-des-Soeurs .......... x
Message du Pierre Arcand, Député de Mont-Royal ........................................................................ xi
Message du Chef Du Parti Québécois Jean-François Lisée ........................................................... xiii
List of Organizers .............................................................................................................................. xiv
Sponsors ........................................................................................................................................... xv
Keynote Speakers ............................................................................................................................... xvi

Dr. Edwin Hancock: Network Analysis in Finance and Medicine
Dr. Patrick Winston: Steps Toward Humanlike Intelligence and Self-aware Systems
Dr. Ed Boyden: How Technologies for Mapping the Brain Could Lead to New Kinds of Artificial Intelligence
Dr. Patrick Wang: How PRAI is changing the world - from Biometrics to Driverless Transportation
Drs. Igor Gurevich/Vera Yashina: Descriptive Approach to Image Analysis
Dr. Diana Inkpen: Detecting Signs of Mental Illness from Social Media
Dr. Joelle Pineau: Improving Health-Care Through AI: Challenges and Opportunities for Reinforcement Learning
Dr. Henry C. Lee: Pattern Recognition & AI in Forensic Investigation

Free Public Lecturers .............................................................................................................................. xxv

Oral Presentation: A. Handwriting Recognition and Document Analysis

The Handwritten Chinese Character Recognition use Convolutional neural networks with the GoogLenet
Jiahao Chen, Ning Bi, Kang Yang and Tan Jun ................................................................. 2
Table-based Document Classification in Historical Document Images
Sara Zhalehpour, Andrew Piper, Chad Wellmon and Mohamed Cheriet ....................................... 8
An overview on on-line handwriting analysis for the assessment of AD and PD
Donato Impedovo, Giuseppe Pirlo and Gennaro Vessio ........................................................... 14
Image Quality Enhancement for Document Extraction with Mobile Devices
Van Cuong Kieu, Florence Cloppet, Vincent Poulain D’andecy, Saddok Kebairi and Nicole Vincent ......... 19
Improving Chinese Writer Identification by Fusion of Text-dependent and Text-independent Methods
Yu-Jie Xiong, Li Liu, Patrick S. P. Wang and Yue Lu ................................................................. 25

Oral Presentation: B. Computer Vision and Image Processing
Real-time interpretation of geometric shapes for digital learning
Omar Krichen, Nathalie Girard, Eric Anquetil and Mickaël Renault ........................................... 31
Generalization of Parameter Recovery in Binocular Vision for a Planar Scene
Marcus Valtonen Örnhag and Anders Heyden ............................................................................... 37
Developing Image Processing Algorithms in Julia
Michael Wirth .................................................................................................................................. 43
Illumination Invariant Face Recognition Via Dual-Tree Complex Wavelet Transform in Logarithm Domain
Guangyi Chen, Tien D. Bui and Adam Krzyzak ............................................................................. 49
Dissimilarity-based representation for radiomics applications
Hongliu Cao, Simon Bernard, Laurent Heutte and Robert Sabourin ........................................... 53

Oral Presentation: C. Machine Learning and Deep Networks
Integrating Learning and Reasoning Services for Explainable Information Fusion
Dan Harborne, Chris Willis, Richard Tomsett and Alun Preece ................................................... 60
A New Approach to Train Convolutional Neural Networks for Monocular Visual Odometry
Mahdi Abolfazli Esfahani, Keyu Wu, Shenghai Yuan and Han Wang ........................................ 66
Segmentation-Free Cell Phenotype Classification using Deep Residual Neural Networks
Qicheng Lao, Haoran Sun and Thomas Fevens ............................................................................. 72
Hybrid Deep Neural Network for Visual Phrase Detection
Lin Bai, Lina Yang, Yuanyan Tang, Lin Huo and Taoshen Li ........................................................ 78
Training Quantized Nets with Adaptive Shared Exponents Based on Statistical Distributions
Katsuhiro Yoda, Wataru Kanemori, Mitsuru Tomono, Makiko Ito and Teruo Ishihara .................. 83

Oral Presentation: D. Hierarchical Image Representation and Vision
Hybrid Image Representation Method based on Bag of Edge Tokens from Octaves of Edge Elements
Elham Etemad and Qigang Gao ........................................................................................................ 90
Distributed Component Forests: Hierarchical Image Representations Suitable for Tera-Scale Images
Michael H.F. Wilkinson and Simon Gazagnes ............................................................................... 96
Recent Developments from Attribute Profiles for Remote Sensing Image Classification
Minh-Tân Pham, Sébastien Lefèvre, Erchan Aptoula and Lorenzo Bruzzone .................................. 102
Algorithms for hierarchical segmentation based on the Felzenszwalb-Huttenlocher dissimilarity
Edward Cayllahua, Yukiko Kenmochi, Jean Cousty, Arnaldo de Albuquerque Araujo and Guillermo Cámara Chávez .............................................................................................................. 108
Intrinsic Quality Analysis of Binary Partition Trees
Tianatahina Jimmy Francky Randrianasoa, Camille Kurtz, Pierre Gançarski, Eric Desjardin and Nicolas Passat ........................................................................................................................................ 114

Oral Presentation: E. Automatic Marking of Descriptive Answers
Clustering Offline Handwritten Mathematical Answers for Computer-Assisted Marking
Vu Tran Minh Khuong, Huy Quang Ung, Cuong Tuan Nguyen and Masaki Nakagawa ....................... 121
Bag-of-features for clustering online handwritten mathematical expressions
Huy Quang Ung, Vu Tran Minh Khuong, Anh Duc Le, Cuong Tuan Nguyen and Masaki Nakagawa ....... 127
A New Roadmap for Evaluating Descriptive Handwritten Answer type
Palaiahnakote Shivakumara, Umapada Pal, Tong Lu, Tapabrata Chakraborti and Michael Blumenstein ........ 133

AI-based Automated Japanese Short-answer Scoring and Support System
Tsunenori Ishioka and Masayuki Kameda ................................................................. 138

An Investigation of Discrete Hidden Markov Models on Handwritten Short Answer Assessment System
Hemmaphan Suwanwiwat, Abhijit Das, Miguel Ferrer, Umapada Pal and Michael Blumenstein ............ 144

Oral Presentation: F. Tracking and Motion

Toward Scalable Visual Digital Evidence Visualization and Multimodal Interaction using Computer Vision Techniques
Serguei Mokhov, Miao Song, Jashanjot Singh, Joey Paquet, Mourad Debbabi and Sudhir Mudur ............. 151

Improving Multiple Object Tracking with Optical Flow and Edge Preprocessing
David-Alexandre Beaupre, Guillaume-Alexandre Bilodeau and Nicolas Saunier .................................. 158

Cubes3D: Neural Network based Optical Flow in Omnidirectional Image Scenes
André Apitzsch, Roman Seidel and Gangolf Hirtz ............................................................................. 164

Evaluation of trackers for Pan-Tilt-Zoom Scenarios
Yucao Tang and Guillaume-Alexandre Bilodeau ................................................................................ 170

Enhancing Human Action Recognition through Temporal Saliency
Vida Adeli, Ehsan Fazl-Ersi and Ahad Harati .................................................................................. 176

Oral Presentation: G. Classification

Support Vector Machine As Graph Theory Problems
William Brendel and Luis Marujo ......................................................................................... 183

On dynamic ensemble selection and data preprocessing for multi-class imbalance learning
Rafael Menelau Oliviera E Cruz, Robert Sabourin and George D. C. Cavalcanti ............................... 189

Multi-label Pixelwise Classification for Reconstruction of Large-scale Urban Areas
Yuanlie He, Sudhir Mudur and Charalambos Poullis ........................................................................ 195

Naive Bayes Classification for Subset Selection in a Multi-label Setting
Luca Mossina and Emmanuel Rachelson .................................................................................... 204

Semi-supervised product quantization for approximate nearest neighbor search
Xiao Bai, Yuan Tian and Jun Zhou ........................................................................................... 210

Oral Presentation: H. Biometrics and Medical Applications

Facial Expression Recognition Using a Multi-level Convolutional Neural Network
Hai-Duong Nguyen, Soonja Yeom, Il-Seok Oh, Kyoung-Min Kim and Soo-Hyung Kim ......................... 217

Machine learning and feature selection for the analysis of Alzheimer Metabolomics Data
Nabil Belacel and Miroslava Cuperlovic ....................................................................................... 222

Sequential Backward Spatio-Spectral Filter Optimization for Motor Imagery Classification
in Ear-EEG Brain-Computer Interface
Yong-Jeong Kim, No-Sang Kwak and Seong-Whan Lee .................................................................... 227

Finger-vein quality assessment by joint representation learning from grayscale and binary images
Huafeng Qin and Mounim A. El Yacoubi ...................................................................................... 232

Failure modelling and anomaly detection of a propulsion subsystem
Catherine Cheung, Julio Valdes, Richard Salas Chavez and Alejandro Lehman Rubio ......................... 238

Oral Presentation: I. Deep Learning and Applications

Identification of Bruised Apples Using Deep Learning and 3D Near-infrared Imaging
Zilong Hu, Jinshan Tang, Ping Zhang and Babu Patlolla ................................................................... 245
Deep Error Correcting Output Codes
Guoqiang Zhong, Hongxu Wei, Yuchen Zheng, Junyu Dong and Mohamed Cheriet ........................................... 250
Deep Learning-Based Corresponding Points Fast Matching
Mohammad Ali Bagheri Orumi, Mahmoud Famouri, Zohreh Azimifar and Azadeh Nazemi ....................... 256
Cardiac Murmur Classification in Phonocardiograms using Deep Convolutional Neural Networks
Pengcheng Xi, Rafik Goubran and Chang Shu .............................................................................................. 261
Object cosegmentation using deep Siamese network
Prerana Mukherjee, Brejesh Lall and Snehith Lattupally ........................................................................... 267

Oral Presentation: J. Forensic Science

Counterfeit Coin Detection Using Stamp Features and Convolutional Neural Network
Ali K. Hmood, Tamarafinide V. Dittimi and Ching Y. Suen ..................................................................... 273
A Novel Way of Identifying Cyber Predators
Dan Liu, Ching Yee Suen and Olga Ormandjieva ....................................................................................... 279
Counterfeit Coin Detection Based on Image Content By Fuzzy Association Rules Mining
Maryam Sharifi Rad, Saeed Khazaei and Ching Y. Suen .......................................................................... 285
Objective Identification of Bullets Based on 3D Pattern Matching and Line Counting Scores
Danny Roberge, Alain Beauchamp and Serge Levesque .......................................................................... 290
Visual Detection of Fake Coins Using Fisher Vectors
Li Liu, Yue Lu and Ching Suen ..................................................................................................................... 296

Oral Presentation: K. Video Processing and Surveillance

"How to rate a video game?" - A prediction system for video games based on multimodal information
Vishal Batchu, Varshit Battu, Dakkannagari Mohana Murali Krishna Reddy and Radhika Mamidi .......... 303
Automatic video stream selection method by on-air microphone detection
Florent Lefevre, Vincent Bombardier, Nicolas Krommenacker, Patrick Charpentier and Petat Bertrand ...... 309
Classification of the Levels of Consciousness within Non-Rapid Eye Movement Sleep
Minji Lee, Benjamin Baird, Olivia Gossseries, Jaakkko Nieminen, Ji-Hoon Jeong, Giulio Tononi and Seong-Whan Lee ........................................................................................................... 315
Robust and Adaptive Vehicle Detection System Using Surveillance Videos
Yan-Lin Chou and Daw-Tung Lin .............................................................................................................. 319
Animation Generation with a Low-Dimensional Simplicial Complex
Brian Gauch and Richard Peters .................................................................................................................. 325

Posters

The existence of oscillatory solutions for a complex-valued neural network model with delay
Chunhua Feng ............................................................................................................................................... 333
Distance Estimation Based Energy Efficient Protocol for Wireless Sensor Network
Rajesh Loganathan, Bhoopathy K and Abdur Rahman Mohamed Ismail ..................................................... 339
LS- SVM's no-reference video quality assessment model considering the network packet loss
under the internet of things
Yibin Hou and Jin Wang ................................................................................................................................ 344
SAF-AFIS: SIFT-based Alignment Free Approach for an Automatic Fingerprint Identification System
Hamza Djebli and Sammy Ait-Aoudia ........................................................................................................... 350
Unsupervised Anomaly Detection in Sewer Images with a PCA-based Framework
Dirk Meijer, Mitchell Kesteloo and Arno Knobbe ....................................................................................... 354
Survey of Issues with Text to Speech Synthesis of Multilingual Indian Texts
Suban Krishnamoorthy and Ching Suen ....................................................................................................... 360
Mobile Based Assistive Technologies for Authentication of Banknotes
<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tamarafinide Dittimi, Ching Suen and Ali Hmood</td>
<td>366</td>
</tr>
<tr>
<td>A Robust Palmprint Matching Method</td>
<td></td>
</tr>
<tr>
<td>Yea-Shuan Huang and Charles Djimy Slot</td>
<td>372</td>
</tr>
<tr>
<td>A novel female facial beauty predictor</td>
<td></td>
</tr>
<tr>
<td>Elham Vahdati and Ching Y. Suen</td>
<td>378</td>
</tr>
<tr>
<td>Restoring height-map images of shiny coins using spline approximation to detect counterfeit coins</td>
<td>383</td>
</tr>
<tr>
<td>Saeed Khazaee, Maryam Sharifi Rad and Ching Y. Suen</td>
<td></td>
</tr>
<tr>
<td>Automatic High-Speed Compressive Tracking with Motion Prediction</td>
<td>388</td>
</tr>
<tr>
<td>Hongjun Li and Wei Hu</td>
<td></td>
</tr>
<tr>
<td>Off-line Handwritten Chinese Recognition and Improvement Based on Caffe Framework</td>
<td>394</td>
</tr>
<tr>
<td>Ning Bi, Haoli Sun and Tan Jun</td>
<td></td>
</tr>
<tr>
<td>Attention-Based Recurrent Neural Networks in Chinese Short Text Classification</td>
<td>399</td>
</tr>
<tr>
<td>Xin Lin, Jiahao Chen, Jun Tan and Ning Bi</td>
<td></td>
</tr>
<tr>
<td>Quaternion Linear Regression for Color Face Recognition</td>
<td></td>
</tr>
<tr>
<td>Yulong Wang, Yuan Yan Tang, Cuiming Zou, Lina Yang and Patrick Wang</td>
<td>403</td>
</tr>
<tr>
<td>Computational Optimization for Fast and Robust Automatic Segmentation in Virtual Microscopy using Brute-force-based Feature Selection</td>
<td>407</td>
</tr>
<tr>
<td>Clement Bouvier, Cédric Clouchoux, Nicolas Soudet, Anne-Sophie Hérard, Zhenzhen You, Caroline Jan, Philippe Hantraye, Gilles Mergoil, B. Helena Rodriguez and Thierry Delzescaux</td>
<td></td>
</tr>
<tr>
<td>Novel Graph-based Image Segmentation: Application to Medical Imaging</td>
<td>413</td>
</tr>
<tr>
<td>Sarada Dakua, Gajendra Mourya, D. Bhatia, Julien Aïnabahed and Abdul Al-Ansari</td>
<td></td>
</tr>
<tr>
<td>Fractal-windowed based Empirical Mode Decomposition Scheme for Protein Sequence Analysis</td>
<td>419</td>
</tr>
<tr>
<td>Lina Yang, Pu Wei, Lin Bai, Cheng Zhong, Yuan Yan Tang and Jianjia Pan</td>
<td></td>
</tr>
<tr>
<td>A new ConvNet architecture for heartbeat classification</td>
<td>424</td>
</tr>
<tr>
<td>Donato Impedovo, Giuseppe Pirlo and Vincenzo Dentamaro</td>
<td></td>
</tr>
<tr>
<td>Fractal Modeling of Big Data Networks</td>
<td>429</td>
</tr>
<tr>
<td>Mahdi Barat Zadeh Joveini, Javad Sadri and Hoda Alavi Khoushal</td>
<td></td>
</tr>
<tr>
<td>Application of Pattern Recognition in Mineral Segmentation and Identification</td>
<td>433</td>
</tr>
<tr>
<td>Hossein Izadi and Javad Sadri</td>
<td></td>
</tr>
<tr>
<td>Sequential Transformation with Geometric Constraints for Matching Oblique Aerial Images</td>
<td>439</td>
</tr>
<tr>
<td>Woo-Hyuck Song, Hong-Gyu Jung and Seong-Whan Lee</td>
<td></td>
</tr>
<tr>
<td>Stochastic Modeling of Camera Errors for Stereo Image Processing</td>
<td>445</td>
</tr>
<tr>
<td>Florian Particke, Christian Hofmann, Markus Hiller, Lucila Patino-Studencki and Jörn Thielecke</td>
<td></td>
</tr>
<tr>
<td>Posture and Fall Detection System Using 3D Motion Sensors</td>
<td>450</td>
</tr>
<tr>
<td>Hamza Qassoud, Miadrag Bolic and Sreeraman Rajan</td>
<td></td>
</tr>
<tr>
<td>Fall Detection Based on Deep Saliency Images</td>
<td>455</td>
</tr>
<tr>
<td>Hongjun Li, Yupeng Ding, Chaobo Li and Ze Zhou</td>
<td></td>
</tr>
<tr>
<td>Online Semi-Supervised Learning with Adaptive Vector Quantization</td>
<td>461</td>
</tr>
<tr>
<td>Yuanyuan Shen, Xu-Yao Zhang and Cheng-Lin Liu</td>
<td></td>
</tr>
<tr>
<td>Pattern Structure of Human Motion Using Single Channel CW Doppler Radar: An Unsupervised Perspective</td>
<td>467</td>
</tr>
<tr>
<td>Julio Valdes, Zachary Baird, Sreeraman Rajan and Miadrag Bolic</td>
<td></td>
</tr>
<tr>
<td>Sequential Minimal Optimization Extended to General Quadratic Programming</td>
<td>473</td>
</tr>
<tr>
<td>William Brendel and Luis Marujo</td>
<td></td>
</tr>
<tr>
<td>Stacked Kernel Extreme Learning Machine for Hyperspectral Image Classification</td>
<td>481</td>
</tr>
<tr>
<td>Yantao Wei, Peng Zhang, Huang Yao, Jiazheng Xu and Xinge You</td>
<td></td>
</tr>
<tr>
<td>A Block-based Path Recognition of Slag Removal Using Convolutional Neural Network</td>
<td>487</td>
</tr>
<tr>
<td>Jeong-Soo Kim, Geon-Tae Ahn and Soon-Yong Park</td>
<td></td>
</tr>
<tr>
<td>Local Binary Pattern Mapping on Graph-based Image Representation for Texture Classification</td>
<td>492</td>
</tr>
<tr>
<td>Srisupang Thewsuvan and Keiichi Horio</td>
<td></td>
</tr>
</tbody>
</table>
Ontology-driven Acquisition of Verbal and Nominalization Patterns for Criminal Events
José A. Reyes-Ortiz, Maricela Bravo and Leonardo Sánchez .............................................................. 498

Classification of Keyphrases using Random Forest and Latent Semantic Analysis
Mireya Tovar Vidal, Gerardo Flores, Azucena Montes, Meliza Contreras González and Ana Patricia Cervantes ................................................................. 506

Efficient fine-grained road segmentation using superpixel-based CNN and CRF models
Farnoush Zohourian, Jan Siegemund, Mirko Meuter and Josef Pauli ........................................... 512

Image Classification using Collaborative Mean Attraction with Sparse Optimization
Hiroti Ogihara and Masayuki Mukunoki ................................................................................................ 518

Saliency and Object Detection
Phutphalla Kong, Matei Mancas, Seng Kheang and Bernard Gosselin ........................................... 523

Comparing Presentation Attack Detection Methods using Convolutional Neural Networks and Local Binary Patterns
Justin Spencer, Deborah Lawrence, Kaushik Roy, Prosenjit Chatterjee, Albert Esterline and Jung-Hee Kim .......................... 529

SDRN: Scalable Deep Rectifier Network for opinion spam detection
Zeinab Sedighi, Hossein Ebrahimpour-Komleh and Ayoub Bagheri .............................................. 535

Learning classifier predictions: is this advantageous?
Vitaliy Tayanov, Adam Krzyzak and Ching Suen .............................................................................. 541

Hybrid RUSBoost Versus Data Sampling to Address Data Imbalance for Breast Cancer
Cytological Malignancy Grading
Muneera Alsaeedi, Thomas Fews, Adam Krzyzak and Lukasz Jelen .................................................... 545

A generalized unified discrete linear method for edge detection by antisymmetric FIR kernels
A. Raji ..................................................................................................................................................... 552

Fast Hierarchical depth map computation from stereo
Vinay Kaushik and Brejesh Lall ........................................................................................................... 558

Fast Context-Annotated Classification of Different Types of Web Service Descriptions
Serguei Mokhov, Joey Paquet and Arash Khodadadi ..................................................................... 562

Classification of Human Activity Level Using Single Channel CW Doppler Radar
Zachary Baird, Julio Valdés, Sreeraman Rajan and Miodrag Bolic ................................................... 571

Comparative Study of a Shape-Based and a Texture-Based Feature Extraction Technique for Mass Classification in Digital Mammograms
Temitope T Adeyemo, Adeola O Olowoye, Temitola M Adepoju, Elijah O Omidiora and Stephen O Olabiyi ................................................................. 576

Comparative Study of Iterative Back Projection and Discrete Algebraic Reconstruction Techniques for Reconstruction of Low Resolution Images
Adebola O. Olowoye, Stephen O. Olabiyi, Elijah O. Omidiora, Temitayo M. Fagbola and Temitope T. Adeyemo ................................................................. 582

Workshop: Image Mining - Mathematical Theory and Applications – Session 1

Training a classifier for automatic flash detection in million images from camera-traps
Bernd Radig and Patrick Follmann ....................................................................................................... 589

Multimodal image analysis for power line inspection
Bushra Jalil, Davide Moroni, Maria Antonietta Pascali, Ovidio Salvetti, Giuseppe Riccardo Leone and Massimo Martinelli .............................................................................. 592

Big Data Application for Smart Features Formation in Medical Diagnostic Tasks
Nataly Ilyasova, Alexandr Kupriyanov, Rustam Paringer, Dmitriy Kirsh, Alexandr Shirokanev and Victor Soifer ................................................................................................. 597

A post-processing method for 3D fluorescence microscopy images
Andrey Nasonov, Yakov Pchelintsev, Alexandra Nasonova and Andrey Krylov .................................................. 602
Automatic Video Mining in Animal Behavior Study using Statistical Shape Models
Yaoguang Zhong, Shawn Horvatic, Takuya Kaneko, Bing Ye and Jie Zhou ........................................................ 607

Separation of arterial and venous vessels images from computer tomography
Alexei Baev, Vasili Dubrovin, Aleksei Rozhentsov and Anna Eruulanova ........................................................... 612

On Conditionality of Pairwise Comparisons in Machine Learning
Sergey Dvoenka and Denis Pshenichny  .............................................................................................................. 618

Dataless Black-Box Model Comparison
Christoph Theiß, Clemens-Alexander Brust and Joachim Denzler ................................................................. 622

In Defense of Active Part Selection for Fine-Grained Classification
Dimitri Korsch and Joachim Denzler .................................................................................................................... 627

Workshop: Image Mining - Mathematical Theory and Applications – Session 2

The features of the recognition of the region of efficiency of microwave devices
Galina Antonova ................................................................................................................................................ 632

Approximation-based transformation of color signal for heart rate estimation with a webcam
Mikhail Kopeliovich and Mikhail Petrushan ................................................................................................................ 638

Development of the filtering algorithm for doubly stochastic images based on models
with multiple roots of characteristic equations
Nikita Andriyanov, Vitaliy Dement’Ev and Konstantin Vasiliev ......................................................................... 643

The solution of the problem of simplifying the images for the subsequent minimization of the image bit depth
Evgenii Semenishehev, Viacheslav Voronin, Igor Shraifel and Dmitrii Chernishov .............................................. 648

Nonlinearity of iris structure as way to improve recognition methods
Viktor Chigrinskiy and Ivan Matveev ...................................................................................................................... 652

Iris Segmentation in Challenging Conditions
Mikhail Korobkin, Gleb Odinokikh, Yuri Efimov, Ivan Solomatkin and Ivan Matveev ........................................... 656

Using optimal circular path method to match piecewise iris templates
Ivan Matveev and Vladimir Novik .......................................................................................................................... 661

Iris Anti-spoofing Solution for Mobile Biometric Applications
Gleb Odinokikh, Yuri Efimov, Ivan Solomatkin, Mikhail Korobkin and Ivan Matveev ........................................... 667

Underwater Image Enhancement Algorithm Based on Logarithmic Transform Histogram Matching
With Spatial Equalization
Viacheslav Voronin, Evgenii Semenishechev and Sos Agaian .............................................................................. 672

Workshop: Image Mining - Mathematical Theory and Applications – Session 3

Computational Graphology Applied to Handwriting Images
Afnan Garoot, Maedeh Safar, Nicola Nobile and Ching Suen .............................................................................. 677

Deep Learning is not a Matter of Depth but of Good Training (Extended Abstract)
Björn Barz and Joachim Denzler .......................................................................................................................... 683

Dynamic-HOG Descriptor for Structured Object Recognition: Case Study on Coins
Ali K. Hmood, Ching Y. Suen and Louisa Lam ................................................................................................................ 688

Increase efficiency of simple images segmentation using detectors based on doubly stochastic random fields
Nikita Andriyanov and Vitaliy Dement’Ev .............................................................................................................. 694

Missing data imputation based on stochastic neighbor embedding
Igor Petrov and Vasily Ryazanov .......................................................................................................................... 698

Shape of Basic Clusters: Finding Coherent ELR-2s via Hough-type Transform
Elena Nelyubina, Vladimir Ryazanov and Alexander Vinogradov ........................................................................ 702
Precedent-based Low Count Rate Image Intensity Estimation using Maximum Likelihood
Distribution Descriptions
Viacheslav Antsiperov ...................................................................................................................... 707

A new mathematical method for automated identification of neurons on microscopic images
Igor Gurevich, Vera Yashina, Adil Tleubaev, Ayat Ospanov and Anton Vladimirov .............................. 712

New Mathematical Methods for Automation of Angiographic Image Analysis of the Human Fundus
Igor Gurevich, Vera Yashina, Ayat Ospanov, Adil Tleubaev and Anatoly Fedorov .................................. 718

Intellectual Information Technology for Symbol Extraction from Ill-structured Graphical Documents

Workshop: Lognormality Principle and its Applications – Session 1

Some observations on lognormality and motor control in handwriting
Roberto Parisi, Antonio Parziale and Angelo Marcelli ........................................................................... 732

On the use of Interval Arithmetic to Bound Delta Lognormal Rapid Human Movements Models
Simon Pierre Boyogueno Bidias, Jean Pierre David, Yvon Savaria and Réjean Plamondon ..................... 736

The Delta-Lognormal model in 2.5D
Saad Chidami, Mylène Archambault-Caron and Réjean Plamondon ..................................................... 743

Extending the Sigma-Lognormal Model of the Kinematic Theory to Three Dimensions
Roman Schindler, Manuel Bouillon, Réjean Plamondon and Andreas Fischer ......................................... 748

A New Experimental Set-up To Run Neuromuscular Tests
Nadir Faci, Simon Pierre Boyogueno Bidias, Réjean Plamondon and Nicolas Bergeron .......................... 753

Gesture Synthesis for Human-Computer Interaction
Luis A. Leiva, Daniel Martin-Albo and Réjean Plamondon .................................................................. 758

Kinematic Reconstruction of Calligraphic Traces from Shape Features
Daniel Berio, Frederic Fol Leymarie and Réjean Plamondon ................................................................ 762

Workshop: Lognormality Principle and its Applications – Session 2

Improving on-line signature skillfulness
Miguel A Ferrer, Moises Diaz, Cristina Carmona-Duarte and Réjean Plamondon....................................... 768

Modeling the Complexity of Biomechanical Tasks using the Lognormality Principle: Applications
to Signature Recognition and Touch-screen Children Detection
Ruben Vera-Rodriguez, Ruben Tolosana, Javier Hernandez-Ortega, Aythami Morales, Julian Fierrez
and Javier Ortega-Garcia ...................................................................................................................... 774

Can the Sigma-lognormal Modeling Help to Monitor Child Graphomotor Skill Progress?
Rémi Céline, Jean Vaillant, Nagay Jimmy and Réjean Plamondon ............................................................. 780

Kinematic Analysis of Fast Pen Strokes in Children with ADHD using the Sigma-lognormal Model
Patricia Laniel, Nadir Faci, Réjean Plamondon, Miriam H. Beauchamp and Bruno Gauthier ...................... 784

Lognormality In Children With Mild Traumatic Brain Injury: A Preliminary Pilot Study
Nadir Faci, Naddley Désiré, Miriam H. Beauchamp, Isabelle Gagnon and Réjean Plamondon .................. 790

Workshop: Lognormality Principle and its Applications – Session 3

A common framework to evaluate Parkinson’s disease in voice and handwriting
Cristina Carmona-Duarte, Miguel Ferrer, Pedro Gómez-Vilda, Arend. W. A. Van Gemmert
and Réjean Plamondon ...................................................................................................................... 795

Exploiting the Lognormality Principle: Three Ongoing Projects
Alexandra Nadeau, Karina Lebel, Joan Carbo, Arnaud Boré, Ovidiu Lungu, Julien Doyon, Hung Nguyen,
Christian Duval, Patrick Boissy, Nicole Vincent, Zoi Kapoula and Réjean Plamondon .............................. 800
Workshop: CALL on e-Learning and Teaching Chinese as a Foreign Language

Application of Virtual Reality Classroom with Panoramic Cameras into Chinese Language Teaching
Fei Song ............................................................................................................................................................... 807

Create Attractive Teaching by Using Domestic Animation
Hongying Gao ..................................................................................................................................................... 813

Applying Internet Thinking in Teaching Chinese as a Foreign Language
Hong Luo ............................................................................................................................................................. 816

List of Papers ......................................................................................................................................................... 820
Oral Session A

Handwriting recognition and document analysis

Monday May 14, 2018, 10:00 AM

Jiahao Chen, Ning Bi, Kang Yang and Tan Jun
The Handwritten Chinese Character Recognition use Convolutional neural networks with the GoogLenet

Sara Zhalehpour, Andrew Piper, Chad Wellmon and Mohamed Cheriet
Table-based Document Classification in Historical Document Images

Donato Impedovo, Giuseppe Pirlo and Gennaro Vessio
An overview on on-line handwriting analysis for the assessment of AD and PD

Van Cuong Kieu, Florence Clopket, Vincent Poulain D'andecy, Saddok Kebairi and Nicole Vincent
Image Quality Enhancement for Document Extraction with Mobile Devices

Yu-Jie Xiong, Li Liu, Patrick S. P. Wang and Yue Lu
Improving Chinese Writer Identification by Fusion of Text-dependent and Text-independent Methods
The Handwritten Chinese Character Recognition use Convolutional neural networks with the GoogLenet

1st Jiahao Chen  2nd Ning Bi  3rd Kang Yang  4th Jun Tan
School of Mathematics School of Mathematics School of Mathematics School of Mathematics
Sun Yat-sen University Sun Yat-sen University Sun Yat-sen University Sun Yat-sen University
Guangzhou, China Guangzhou, China Guangzhou, China Guangzhou, China
chenjh47@mail2.sysu.edu.cn mcsbn@mail.sysu.edu.cn EzioDeYK@hotmail.com mcstj@mail.sysu.edu.cn

Abstract—With the outstanding performance in 2014 at the ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC14), an effective Convolutional neural network(CNN) model named GoogLenet has drawn the attention of the mainstream machine learning field. In this paper we plan to take an insight into the application of the GoogLenet in the Handwritten Chinese Character Recognition(HCCR) on the database HCL2000 with several necessary adjustments and also state-of-the-art improvement methods for this end-to-end approach. By the experiment we have found that the application of the GoogLenet for the Handwritten Chinese Character Recognition (HCCR) results into significant high accuracy, to be specific more than 99 percent for the final version, which is encouraging for us to the further research and improvement.

Index Terms—CNN, GoogLenet, Handwritten recognition, Chinese character

I. INTRODUCTION

For many reasons the Handwritten Chinese Character Recognition(HCCR), especially the off-line HCCR, is still a sophisticated challenge, leaving many unsolved problems such as the deficient recognition accuracy [1]. Unfortunately rare improvement by the traditional “pre-treatment + feature-extraction + classification-machine” approaches are proposed, leaving significant margin to the human capability. In 2012 a new combining CNN model named Multi-column deep neural network was proposed by Dan et al. [2] from the IDSIA laboratory made the first recorded successful realization of the deep learning models by the HCCR with enormous vocabulary. In the ICDAR 2013, the Graham et al. [3] won the first reward by the online HCCR with the spatially-sparse CNN. Similarly the researchers from Fujitsu R.D Center made the best performance at the off-line HCCR with the alternately trained relaxation CNN (ART-CNN) [4]. By then the accuracy rate for the HCCR both online and off-line with CNN models has made important advances. Other tricks such as ReLU and Dropout [5] are used to make a fast convergence, in addition to omit the over-fitting problem. However the CNN models on hand all seem to be not slim or deep enough for the more complicated problems. On the other hand, all of them require a huge size for the storage. As reported, the ATR-CNN requires 2.46GB storage size for the dictionary and the wide-used AlexNet [5] needs 60 million parameters to be trained.

In the ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC14), a new network model named GoogLenet [6], developed by the Google Inc. and named after the well-known Lenet [7] by LeCun, became the winner against all the other approaches. Meanwhile, this model requires only 7 million parameters, which is almost 30x fewer than its competitor VGGNet [8]. In this paper, we will focus on this efficient deep CNN model with its application on the off-line HCCR. There are many previous experiments and some of them also made great successes. For going “deep” into the field of HCCR, we take such attempt by own and also make some state-of-the-art adjustments to this model, which are good to gain a budget for the computation and a faster convergence.

The remainder of this paper is organized as follow. Section II gives a brief introduction of the CNN and its application on the HCCR, also introduces the inside of the GoogLenet and the powerful Inception model. The methods for the adjustments of the GoogLenet with diverse versions are shown in Section III. Experimental results are given in Section IV and conclusions are drawn in section V.

II. RELATED WORKS

A. The Development of CNN

Convolutional neural networks(CNN) involve many more connections than weights; the architecture itself realizes a form of regularization. In addition, a convolutional network automatically provides some degree of translation invariance. This particular kind of neural network assumes that we wish to learn filters, in a data-driven fashion, as a means to extract features describing the inputs [9].

Since then, CNNs have typically a standard structure that stacked convolutional layers are followed by one or more fully-connected layers. The first CNN models designed by LeCun et al. in such way was called LeNet and applied in various assignments such as handwritten zip code as well as the document recognition missions [7]. In addition, Convolutional networks combine three architectural ideas to ensure some degree of shift, scale, and distortion invariance: 1) local receptive fields; 2) shared weights (or weight replication); and 3) spatial or temporal sub-sampling.

Copyright © 2018 by CENPARMI, Concordia University
B. Theoretical Parts for the CNN

1) Convolution and Max-Pooling: Typically convolutional layers are interspersed with sub-sampling layers to reduce computation time and to gradually build up further spatial and configure invariance. We will briefly introduce CNN algorithm for convolution and max-pooling.

Given \( x^l_j \in R^{M_l \times N_l} \), represents the \( l \)-th map in the \( j \)-th layer, \( j^{th} \) kernel filter in the \( l^{th} \) layer connected to the \( i^{th} \) map in the \((l-1)^{th} \) layer denoted \( K_{ij}^l \in R^{K_l \times K_l} \) and index maps set \( M_l = \{ i \}^{N_l} \)in the\((l-1)^{th} \)layer map connected to the \( j^{th} \) map in the \( l^{th} \) layer. So the convolution operation can be given by equation(1),

\[
x_j^l = f(\sum_{i \in M_l} x_i^{l-1} \ast k_{ij}^l + b_j^l)
\]

where \( f(\cdot) \) is ReLU non-linearity activation function \( f(x)=\max(0,x),b_j^l \) is bias. And pooling equation can be described in the equation(2),

\[
x_j^l = \text{down}(x_i^{l-1})
\]

where down(.) is sum-sampling function to compute the max value for each \( n \times n \) region in \( x_i^{l-1} \) map.

2) Loss-function Introduction: Softmax regression is an extended model for logistic regression [10] and applied as an effective method for multi-class classification problem.

Suppose we have \( T \) categories and the training data for the each category are denoted as \((x_i,y_i)\), where \( i = 1,...,N \), with \( x_i \in R^e \) and \( y_i \in \mathcal{R} \) being the feature vector and label apart. CNN aims to minimize the following cross-entropy loss function:

\[
J(\theta) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} l(y_i = t) \log \sum_{t=1}^{T} e^{\theta^T x_i}
\]

where \( \theta \) is model parameter, \( \sum_{t=1}^{T} e^{\theta^T x_i} \) is a factor of normalization, and \( l(.) \) is an indicating function. The loss function of \( j(\theta) \) can be minimized by using stochastic gradient descent(SGD) algorithm, during the training process of CNN.

3) ReLU: Compared with the traditional neural network, where the activation pots are in the form of the followed functions:

\[
f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}
\]

or

\[
f(x) = \text{sigmoid}(x) = \frac{1}{1 - e^{-x}}
\]

Krizhevsky et al. [5]have proposed a inspired non-linear activation function

\[
f(x) = \max(0,x)
\]

, which is named as ReLU. With regards to the deep network, the sigmoid or tanh function changes very slowly near the saturation area and results in the derivative being closed to 0.

4) Dropout: The trick “Dropout” is also proposed by Krizhevsky [5]as a practical regularization method for the training of the deep models, in order to avoid the over-fitting problem.

C. Overview of the GoogLenet

GoogLenet [6] is 22-layer deep network when counting only layers with parameters, while all the convolutions, including those inside the inception modules, use rectified linear activation. At the bottom of the model stand the convolutional layers, which are 7x7 with stride 2 and 3x3 with stride 1 followed by max-pooling layers respectively. The main stem of the GoogLenet is 9 repeated reception modules, which will be explained in details later. These blocks are separated into types, 3(a,b), 4(a,b,c,d,e)and 5(a,b).The use of average pooling before the classifier is based on the work by Lin et al [11].It was found that a move from fully connected layers to average pooling improved the top-1 accuracy by about 0.006, however the use of dropout remained essential even after removing the fully connected layers.

We would expect to encourage discrimination in the lower stages in the classifier, increase the gradient signal that gets propagated back, and provide additional regularization. These classifiers take the form of smaller convolutional networks put on top of the output of the Inception (4a) and (4d) modules.During training, their loss gets added to the total loss of the network with a discount weight (the losses of the auxiliary classifiers were weighted by 0.3). At inference time, these auxiliary networks are discarded.

The exact structure of the extra network on the side, including the auxiliary classifier, is as follows:

- An average pooling layer with 5x5 filter size and stride 3, resulting in an 4x4x512 output for the (4a), and 4x4x528 for the (4d) stage.
- A 1x1 convolution with 128 filters for dimension reduction and rectified linear activation.
- A fully connected layer with 1024 units and rectified linear activation.
- A dropout layer with 0.7 ratio of dropped outputs.
- A linear layer with softmax loss as the classifier (predicting the same 1000 classes as the main classifier, but removed at inference time).

D. The Inception Module

The most aspired part of the GoogLenet is the inception module, which is based on finding out how an optimal local sparse structure in a convolutional vision network can be approximated, and be a decreasing number of patches over larger and larger regions. In order to avoid patch-alignment issues, current incarnations of the Inception architecture are restricted to filter sizes 1x1,3x3 and 5x5, however this decision was based more on convenience rather than necessity. It suggests that adding an alternative parallel pooling path in each such stage should have an additional beneficial effect, too.

Meanwhile, in order to avoid the inevitable increase in the number of the outputs from stage to stage by using several convolutional layer in one module, it leads to the seconde idea of the proposed architecture: judiciously applying dimension reductions and projections wherever the computational requirements would increase too much otherwise. This is based
on the success of embedding: even low dimensional embedding might contain a lot of information about a relatively large image patch. That is, 1x1 convolutions are used to compute reductions before the expensive 3x3 and 5x5 convolutions.

With the inception module, we can extract local feature representation using flexible convolutional kernel filter sizes with layer-by-layer structure, which was proved to be robust and effective for the large scale high-resolution images.

E. The Adjustment and Improvements

GoogLeNet employed around 7 million parameters, which represented a 99 reduction with respect to its predecessor Alex-Net, which used 60 million parameters. Furthermore, VGG-Net employed about 3x more parameters than Alex-Net. The computational cost of Inception is also much lower than VGG-Net or its higher performing successors. This has made it feasible to utilize Inception networks in big-data scenarios. Still, the complexity of the Inception architecture makes it more difficult to make changes to the network and much harder to adapt it to new use-cases while maintaining its efficiency, while the original paper for the GoogLenet does not provide a clear description about the contributing factors that lead to the various design decision of the GoogLenet architecture. In the later parts, we are going to take steps forward to some state-of-the-art improvements for this model.

1) General Design Principles: we start with describing a few general principles and optimization ideas that proved to be useful for scaling up convolution networks in efficient ways. Still, grave deviations from these principles tend to result in deterioration in the quality of the networks and fixing situations where those deviations were detected and resulted in improved architectures [12].

- Avoid representational bottlenecks, especially early in the network.
- Higher dimensional representations are easier to process locally within a network.
- Spatial aggregation can be done over lower dimensional embedding without much or any loss in representational power.
- Balance the width and depth of the network.

2) Factorizing Convolutions with Large Filter Size: Since Inception networks are fully convolutional, each weight corresponds to one multiplication per activation. Therefore, any reduction in computational cost results in reduced number of parameters. This means that with suitable factorization, we can end up with more disentangled parameters and therefore with faster training.

   a) Factorization into Smaller Convolutions: Convolutions with larger spatial filters (e.g. 5x5 or 7x7) tend to be disproportionately expensive in terms of computation. So a reduction of the geometric size of the filters comes at a large cost of expressiveness. It seems natural to exploit translation invariance again and replace the fully connected component by a two-layers convolutional architecture: the first layer is a 3x3 convolution, the second is a fully connected layer on top of the 3x3 output grid of the first. In practice, this setup clearly reduces the parameter count by sharing the weights between adjacent tiles.

   b) Spatial Factorization into Asymmetric Convolutions: The above results suggest that convolutions with filters larger than 3x3 might not be generally useful as they can always be reduced into a sequence of 3x3 convolutional layers. In theory, we could go even further and argue that one can replace any nxn convolution by a 1xn convolution followed by a nx1 convolution and the computational cost saving increases dramatically as n grows (shown in Fig.1). In practice, we have found that employing this factorization does not work well on early layers, but it gives very good results on medium grid-sizes (On mxm feature maps, where m ranges between 12 and 20).

   Fig. 1. substitution of the inception module

3) Efficient Grid Size Reduction: Traditionally, convolutional networks used some pooling operation to decrease the grid size of the feature maps. In order to avoid a representational bottleneck, before applying maximum or average pooling the activation dimension of the network filters is expanded. However, this creates a representational bottleneck as the overall dimensionality of the representation drops resulting in less expressive networks. Instead of doing so, researchers suggest another variant the reduces the computational cost even further while removing the representational bottleneck, shown in Fig.2.

   Fig. 2. efficient grid size reduction
4) Batch-Normalization: Training Deep Neural Networks are complicated by the fact that the distribution of each layers inputs changes during training. This phenomenon is referred as internal co-variate shift. Since the full whitening, white is well-acknowledged [13], of each layers inputs is costly, it is suggested to normalize each scalar feature independently. Thus, the researchers introduce, for each activation $x^{(k)}$, a pair of parameters $\gamma^{(k)}, \beta^{(k)}$, which scale and shift the normalized value:

$$y^{(k)} = \gamma^{(k)} \hat{x}^{(k)} + \beta^{(k)}$$

These parameter are learned along with the original model parameters, and restore the representation power of the network. Indeed, by setting $y^{(k)} = \sqrt{\text{Var}[x^{(k)}]}$ and $\beta^{(k)} = E[x^{(k)}]$, we could recover the original activations, and in practice we use mini-batches in stochastic gradient training, each mini-batch produces estimates of the mean and variance of each activation. To illustrate, we have $m$ values of the activation in the mini-batch:

$$B = x_{1...m}$$

Let the normalized values be $\hat{x}_{1...m}$, and their linear transformations be $y_{1...m}$. We refer to the transform

$$BN_{\gamma,\beta} : x_{1...m} \rightarrow y_{1...m}$$

as the Batch Normalizing Transform [14]. The BN Transform is presented in Algorithm 1. In the algorithm, $\epsilon$ is a constant added to the mini-batch variance for numerical stability.

1 Batch Normalizing Transform
2 Input:
3 Value of x over a mini-batch $B = x_{1...m}$;
4 Parameters to be learned: $\gamma, \beta$
5 Output: $y_i = BN_{\gamma,\beta}(x_i)$
6 $\mu_B \leftarrow \frac{1}{m} \sum_{i=1}^{m} x_i$
7 $\sigma_B^2 \leftarrow \sum_{i=1}^{m} (x_i - \mu_B)^2$
8 $\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sigma_B + \epsilon}$
9 $y_i \leftarrow \gamma \hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i)$

During training we need back-propagate the gradient of loss $L$ through this transformation, as well as compute the gradients with respect to the parameters of the BN transform.

$$\frac{\partial L}{\partial x_i} = \gamma \frac{\partial L}{\partial y_i}$$

$$\frac{\partial L}{\partial \sigma_B^2} = \sum_{i=1}^{m} \frac{\partial L}{\partial \hat{x}_i} \cdot (x_i - \mu_B) \cdot -\frac{1}{2} (\sigma_B^2 + \epsilon)^{-\frac{3}{2}}$$

$$\frac{\partial L}{\partial \mu_B} = \sum_{i=1}^{m} \frac{\partial L}{\partial \hat{x}_i} \cdot \frac{1}{\sqrt{\sigma_B^2 + \epsilon}}$$

$$\frac{\partial L}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial L}{\partial y_i} \cdot \hat{x}_i$$

$$\frac{\partial L}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial L}{\partial y_i}$$

For the output depending only on the input, deterministically. Once the network has been trained, we use the normalization.

$$\hat{x} = \frac{x - E[x]}{\sqrt{\text{Var}[x]} + \epsilon}$$

III. The Adoption of the GoogLenet for HCCR and the Adjustments for the Improvement

The original structure of the GoogLenet, which was used for the ILSVRC-2014, is proposed by the following map. For the application on the dataset HCL2000, we have made three steps for the adaptation of the Googlenet with state-of-the-art research findings and thus proposed 3 versions of the networks for the HCCR with the continues improvement on the accuracy as well as the gain of the time-consuming.

A. Version 1: The Changes by the Input and Output Layers

With the original structure of the GoogLenet, we have made small changes on the layer both on the top and bottom of the network, followed the available achievement by experienced researchers [15]. Same as the former statement, the size of the input for the models is 64x64. Therefore, we have changed the kernel-size of the first convolution layer into 9 and remove the pooling layer before the inception 3a. So the inputs of the inception 3a become 28x28, which is uniform to the input of this inception module of the original structure. Meanwhile, by reducing the number of pooling layers we can avoid unnecessary loss of the feature information. On the other hand, we changed the parameter–output number–of the inner-product layer before the classification layer because of the classification number being 3755.

B. Version 2: Equipped with Batch-Normalization and Fraction of the Convolutional Layer

After the version 1, we thought about putting some practical tricks into the models. Firstly, we have add layers for Batch-normalization after every convolutional layers, which proved to be beneficial to the convergency of the hyper-parameters. And then, we have fractionated all the convolutional layers with large kernel-size into several smaller convolutions, to be specific, we alter the convolution layer with kernel-size 9 into 3 overlying kernel-size-3 convolutions and the one with kernel-size 5 into 2. Although the depth of the models seems to be significantly increased, we have reduced not only the number of parameters but also the computation cost.

C. Version 3: Spatial Fraction

With the accomplishment of the version 2, we decide to make deep transformation for the network taking guidance of the published essays. For grid size reduction, we have adopted the max-pooling layers after each group of the inception module, remove the insolated 1x1 convolutional layers in the inception modules before these max-pooling layers and changed the strides of the layer before the concatenation layer in these inception modules into 2. The reduction of the pooling-layer number provide advantages against information loss. Thereby we maintain a good balance between the depth
and width of the model as well. In addition, we replaced the 3x3 convolutional layers in the inception 4(a-e) and 5(a,b) with both the 1x3 and 3x1 convolutional kernels.

IV. EXPERIMENTS AND ANALYSIS

A. Experimental Data

We used the off-line HCL2000 data set [16] for training and testing. HCL2000 contains 3,755 frequently used simplified Chinese characters in GBI written by 1,000 different subjects (sample images shown in Fig. 3), and a total image sample number of 3,755,000. The whole data set is separated into training set and test set with the ratio 7:3.

![Fig. 3. HCL Chinese handwritten data set](image)

B. Pre-processing and Experimental Settings

First we shuffle our training data, after a visual inspection of several characters rescaled to various sizes and suitable to our designed experimental CNN architecture, we decided to normalize off-line characters image 64x64 as inputs for the models. Before resizing, we reversed the grey values of the image in order to ensure fast computation. We conducted our experiments on an open CNN platform called Caffe [17] using a GTX TITAN BLACK GPU card.

C. The Training of Different Visions of the GoogLeNet

For the testing of 3755 classes of HCL, the recognition rates with three versions of the GoogLeNet was adequate. It can be seen that GoogLeNet improved the recognition performance significantly, which demonstrates that deeper layer-by-layer architecture helps to extract the more nature and abstract representation of off-line Chinese characters and improve their accuracy.

Firstly, we have trained all three models with a learning rate by 0.01, with 90 percent decay every 20000 steps, and then plot the graphs to represent the changing-path of the training loss as well as the testing accuracy for the last two versions (the training of the first version takes much more time then we decide not to present it in the following), to find the inflection points where the losses do not continue to decrease. Then we have fine-tuned the trained models with 0.00001 learning rate and showed them by the following Fig. 4. It clearly indicates that the effectiveness of using adjustment tricks can further improve the performance of GoogLeNet for HCCR, not only by the training velocity but also for the recognition accuracy.

![Fig. 4. accuracy and loss figure of the 1st version](image)

By the meantime we have also compared the recognition capacity of our GoogLenet models as well as other parameters in Table. I.

V. CONCLUSION

In this paper, we presented a new deep learning model, GoogLeNet, for the recognition of handwritten Chinese character. GoogLeNet uses several Inception modules to construct an efficient deep network, taking the advantage of finding optimal local construction and allowing to repeat it spatially. The GoogLeNet is designed to be very deep yet slim, with a total of no more than 30 layers (counting for all convolutional layers by the final version, pooling layers, fully connect layers and softmax output layer). We employ state-of-the-art improving knowledge, such as the Batch-normalization, Convolutional Layer's Factorization and etc, to enhance the performance of GoogLeNet. Yet we have tremendously improved the recognition capacity of the network as well as the time-consuming for the model-training. Although the storage size for the models’ parameters is apparently too large for practical use, while the output of the inner-product layer was changed to 3755 and for the addition of the BN layers. The further research will be focused on the storage size reduction.

ACKNOWLEDGEMENT

This work was supported by the Guangdong Provincial Government of China through the “Computational Science Innovative Research Team” program and Guangdong Province Key Laboratory of Computational Science at the Sun Yat-sen University, and the National Science Foundation of China (11471012).
### TABLE I
THE BEST RECOGNITION ACCURACY OF OUR PROPOSED METHOD AGAINST PREVIOUS METHODS

<table>
<thead>
<tr>
<th>System</th>
<th>Top1</th>
<th>Top5</th>
<th>training iteration</th>
<th>fine-tune iteration</th>
<th>Die.size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Googlenet(v1)</td>
<td>0.976</td>
<td>0.994</td>
<td>120000</td>
<td>n/a</td>
<td>85.5MB</td>
</tr>
<tr>
<td>Googlenet(v2)</td>
<td>0.983</td>
<td>0.998</td>
<td>60000</td>
<td>50000</td>
<td>500MB</td>
</tr>
<tr>
<td>Googlenet(v3)</td>
<td>0.992</td>
<td>0.998</td>
<td>70000</td>
<td>30000</td>
<td>693MB</td>
</tr>
</tbody>
</table>

### REFERENCES


Abstract—Layout analysis of historical document images is considered a challenging task due to their various structures and complex layouts. Tables are one of the common contents of these documents and detecting them is an essential step for scholars to analyze, explore and mine the valuable information encoded in these documents. In this paper, we propose a new application for Mel-frequency cepstral coefficients (MFCCs) by using them for table detection in historical document images. MFCCs give a good discrimination of lower frequency components of the signal compared to the higher frequency components. The proposed system first converts the image to a one-dimensional signal. Then, MFCC-based features are extracted from that signal. Classification is performed using support vector machine (SVM) and random forest (RF) classifiers. The proposed method shows promising results using the images from two different databases with varying layouts.

Index Terms—MFCC, table detection, historical documents classification, SVM, RF

I. INTRODUCTION

In the past decades, research in the field of automatic analysis and recognition of historical documents has experienced considerable attention. In order to retrieve and understand visual information from these documents, their content needs to be characterized in a meaningful way [1]. Document contents can take different textual and graphical forms like text, symbols, footnotes, figures, lists and tables. Tables are found in diverse document classes, i.e. journals, newspapers, forms and invoices. They are used for summarizing and presenting correlational information of documents in a compact and organized way [2].

Generally, tables in documents are categorized into two main groups according to their structure: ruling-line tables (RL-Ts) and non-ruling-line tables (NRL-Ts) [2]. RL-Ts are the most common type of tables. Their regions are either bound by a bounding box or separated from other parts using a number of vertical or/and horizontal lines. They can further classify into smaller groups of closed tables, non-closed tables and parallel tables according to the lines they contain. NRL-Ts are generated using neither bounding box nor ruling lines. NRL-Ts are usually limited to the reports and letters and rarely scientific journals. Fig. 1 depicts some examples of each category in historical documents from eighteenth century.

There are several previous works developed for table detection in scanned document images. One of the first works in this field was done by Kieninger et al. [3]. They developed a table recognition system called T-Recs. This system locates and extracts the structures of the tables using the words bounding box geometry. Applying a bottom-up approach, a segmentation graph is built and these bounding boxes are clustered into candidate table regions. This system has low accuracy in the case of multi-column layouts since it is only based on bounding boxes. Cersarini et al. [4] proposed a method for table detection in scanned document images based on searching for parallel lines. The perpendicular lines and white space near the lines are then located to verify the hypothesis. However, relying only on horizontal or vertical lines limits this method only to the RL-Ts recognition. Gatos et al. [5] focused on finding intersections in tables with both horizontal and vertical ruling lines. Then, table detection is achieved by finding the intersection pairs. The main drawback of this system is its limitation to tables with vertical or horizontal ruling lines. Mandal et al. [6] presented a method considering this hypothesis that the word gaps are smaller than the gaps between table columns. Using this hypothesis, they selected the text lines which are likely from a table. Shafait and Smith [7] developed a table detection algorithm for documents with multi-columns. Their method first determines the page layout by finding the text column partitions and candidate some of them as table partitions. Next, table partitions are clustered into table columns. Finally, the horizontal ruling is used to locate the table. Kumar et al. [8] introduced a method for classification of the document images into two groups of with or without tables. They modeled the spatial relationship of patches using the statistic of codewords. The images are partitioned horizontally and vertically and a histogram of codewords is computed for each partition and trained/tested using a random forest classifier. Soe et al. [9] presented an algorithm for locating tables and their cells in camera-captured documents. Their method detects the cell junctions and labels their connectivity. The encoded connectivity information is used for designing a cost function and this cost function is minimized via a belief propagation algorithm. Their proposed method can detect even skewed tables or multiple tables in a page. However, it is only limited to tables with ruling lines. Abdullah and Jasim [10] proposed a method for table detection using improved Hough transform to find vertical and horizontal lines. This method is also limited to RL-Ts.
A reliable and robust feature extraction method is essential for table detection tasks. The irregular and various layouts of tables make finding the proper features harder. So far, a variety of features have been used by conventional methods for table detection in document images; e.g., ruling lines, gaps between words and columns, table boundaries and junctions. The most common disadvantage of these features is that they are often not applicable to both RL-Ts and NRL-Ts. This paper introduces a practical method for classifying documents with various layouts into classes of with and without table using MFCCs. The fact that the occurrence of words in tables follow a column-wise harmonic manner which has a lower frequency compared to the other textual contents of the pages brings up the idea of using a spectral approach. MFCCs are widely used in speech recognition applications. They are produced through cepstral analysis and then wrapped according to Mel scale which emphasizes the lower frequency component over the higher ones. As far as we know, using MFCCs for image processing purposes so far has been limited to identification of satellite images and leukemia cells also face, palm print and gesture recognition [11]–[14]. The proposed method is based on converting the image into a one-dimensional signal and perform the same process for MFCCs extraction as speech. Then, some statistical features are extracted from each signal and fed to our classifiers. The rest of paper is organized as follows. Section II provides the detailed explanation of the proposed table detection method. The proposed system is evaluated in section III. Finally, conclusion and future works are given in section IV.

II. PROPOSED METHODOLOGY

Our proposed method first extracts MFCCs related features and then feed these features to a classifier to decide the presence of a table in the images. Bellow, we will go into more details of each step.

A. MFCC based features

Steps for extraction of MFCCs for our method are the same as the steps in voice recognition. However, first, the document image is converted from a two-dimensional image to a one-dimensional signal since MFCC algorithm only accepts one-dimensional signals as an input. In this research before vectorizing the images they all scale to 700x450 pixels for making the images comparable. Fig. 2 shows the steps for extracting MFCCs features from document images.

![Fig. 2: MFCCs extraction process.](image)

a) Frame the signal: In order to have a reliable spectral analysis, the signal needs to be stationary. Therefore, the signal is considered statistically stationary in short periods of samples called frames. Shorter frames result in not having enough samples to calculate the power spectrum and longer frames cause having a frame with too many changes throughout the frame. The one-dimensional signal is divided into successive frames of $N$ samples with $O$ samples overlap to avoid loss of information. Values for $N$ and $O$ used in this research are $N = 100$ and $O = 60$ samples.

b) Windowing: Windowing equals to the point-wise multiplication of the frame and window function in the time domain. This is equivalent to frequency convolution, where all the frequencies are replaced with Fourier transform of the window function. Windowing is usually applied to minimize the disruptions at two ends of the frame and increase the continuity of the neighbor frames. The convolution distributes the actual frequency to the bins around it. Therefore, by using a proper window function except for rectangular one, fewer side lobes appear compared to the main lobe. This results in having less artificial spectral disruptions for each frame and makes the frames suitable for spectral analysis. The most common window function for carrying out the windowing is the Hamming window. This function has a narrow main lobe with low-level side lobes, which is usually represented as:
n is the window function with sample length of \( s \) the power spectral estimate for the frame the process is faster and less complex. The periodogram-based computation, each frame is divided into small DFTs. Thus, for applying the Discrete Fourier Transform (DFT). For FFT coefficients of computing a 512-point FFT and keeping the first 275 is the length of FFT. Here we follow the common practice of computing a 512-point FFT and keeping the first 275 coefficients.

\[ w(n) = 0.54 - 0.46 \cos \left( \frac{2\pi n}{N-1} \right), \quad 0 \leq n \leq N - 1 \] (1)

c) Fast Fourier Transform (FFT): The next step is calculating the periodogram estimate of the power spectrum by transforming the windowed signal from the time to the frequency domain using FFT. FFT is a fast processing algorithm for applying the Discrete Fourier Transform (DFT). For FFT computation, each frame is divided into small DFTs. Thus, the process is faster and less complex. The periodogram-based power spectral estimate for the frame \( s_i(n) \) or the periodogram estimate of the power spectrum is given by:

\[ P_l(k) = \frac{1}{N} \sum_{n=1}^{N} s_i(n)h_i(n)e^{-j2\pi kn/N}^2, \quad 0 \leq k \leq K - 1 \] (2)

where \( s_i(n) \) is the \( i^{th} \) framed signal with \( 1 \leq n \leq N \), \( h_i(n) \) is the window function with sample length of \( N \) and \( K \) is the length of FFT. Here we follow the common practice of computing a 512-point FFT and keeping the first 275 coefficients.

d) Mel Filter banks: The calculated power spectrums are then mapped into the Mel scale using triangular overlapping windows called Mel filter banks (see Fig.3(a)). With the help of these bandpass filter banks, energies in various Mel-scaled frequency regions are obtained. The Mel scale defines the space and wideness of each filter. The first filter indicates the amount of energy near 0 Hz. As the frequencies get higher, there is less concern about the variations in the amplitude and the filters get wider (see Fig.3(b)). The most popular formula for mapping the frequency \( f \) Hz to the \( m \) Mel is given in Eq. (3).

\[ m_f = 2595 \log_{10} \left( \frac{f}{700} + 1 \right) \] (3)

Each filter bank is multiplied by the power spectrum, then the coefficients are summed up which results in having the same number of coefficients as the number of filter banks for each frame. The standard number of filters is 20-26. In this study, the frequency axis of the spectrogram is mapped into Mel-scaled frequency axis by filtering the frequency bins from 64 Hz to two kHz and 23 Mel filter banks into 15 equally spaced Mel-bands.

e) Logarithm: Once the energies are estimated in the previous step, we take their logarithm. This compression operation allows the cepstral mean subtraction for channel normalization. The output of this step is the spectro-temporal representation of the signal called logarithmically scaled Mel-spectrogram (LogMS). Fig. 4 shows the behavior of the LogMS for different structural layouts. As we can see in the Fig. 4 (a and b) when a document contains text lines, the energy concentrations are almost in the same range with this difference that for the two columns page, concentration is around higher coefficients. Fig. 4 (c) depicts an example of having both figure and text in a same document. In this case, coefficients for the figure are not following any order, but as soon as the text lines are starting the energy concentration is around the same amount. Finally, Fig. 4 (d) shows the case that we have both table and text on the page. For the table also the energy concentration is in the same range, however, as we expected, the table energy peaks occur in lower coefficients compared the main text.

f) Discrete Cosine transform (DCT): The final step for extracting MFCCs is converting the logarithm of Mel cepstrum to the time domain. This step is done to decorrelate the filter bank energies. This correlation is due to their overlapping regions. The reason for favoring DCT over inverse DFT in this part is that DCT contains more information with a smaller number of coefficients. The MFCCs are computed as below:

\[ c_i(m) = \sum_{l=1}^{L} (\log S_i(l)) \cos \left[ m \left( l - \frac{1}{2} \right) \frac{\pi}{M} \right], \quad m = 0, 1, \ldots, M-1 \] (4)

where \( M \) is the number of MFCCs and \( L \) is the number of filter banks. \( S_i(l) \) are the Mel filter banks energies. Only 12-20 coefficients out of all DCT coefficients are usually selected since the higher coefficients indicate faster changes in the energies which can lead to performance degradation of the system. The first coefficient is also excluded since it represents the mean value of the frame and carries little information.

Since MFCCs only describe the power spectrums envelopes for each frame, the signal may lose the local dynamics information i.e., trajectories of the MFCCs over the time. Therefore, it is a common practice to calculate the delta and double delta coefficients. These coefficients are also known as differential and acceleration coefficients, respectively. Double delta coefficients are obtained in the same way as delta coefficients except instead of using static coefficients, deltas are used.

For this research, nine MFCCs along their delta and double delta coefficients and 15 LogMS coefficients are calculated. Then, nine statistical functions, namely, maximum, minimum, maximum position, minimum position, range, mean, variance,
Fig. 4: Examples of different behavior of the LogMS coefficients vs frames in presence of different structural layouts. Each layout regions are marked with a red dashed line.
median, mode, entropy, kurtosis and skewness are applied to these 42 MFCC based coefficients. The distribution of coefficients in each frame suggests using some form of fitting. Here, we used symmetric generalized Gaussian distribution (GGD) and asymmetric generalized Gaussian distribution (AGGD) fitting. The shape, variance and entropy information are extracted from the given MFCC-based features by fitting them with a GGD. Also, shape, mean, right and left variances, skewness and kurtosis and entropy are extracted in four orientations (horizontal, vertical, main diagonal and secondary diagonal) by an AGGD fitting. The final feature vector size for this research is 1516.

B. Classification

For our experiments, we used two classifiers, SVM and RF. LIBSVM toolbox [15] is used for implementing the SVM with a degree three polynomial kernel function. The hyperparameters for SVM are calculated using a validation set equals to 10% of the training set. For RF we used 1000 trees and the number of attributes selected for each tree is equal to the square root of the number of features.

III. PERFORMANCE AND EVALUATION RESULTS

We used two databases to conduct our experiments. The first database is used to evaluate our method against the proposed method by Kumar et al. [8]. The database used by them for evaluating their table detection method contains 618 Arabic document images from a collection of hand-drawn and printed tables (216 images) and a set of mixed-form documents (402 images). This database is chosen since it is one of the rare publicly accessible databases with the fair number of scanned document images similar to our actual database.

The second database is created using a large variety of labeled historical document images with different layouts from Eighteenth Century Collections Online (ECCO) database. ECCO is an online eighteenth-century document archive in any language that is printed in the United Kingdom between the years 1701 and 1800. The prepared database contains 6412 document images with different and complicated layouts. Among these images, 888 images are from RL-Ts category, 144 images from NRL-Ts category. The rest of the images are from various structural groups, e.g., pages with one or more than one column, pages containing figures, index/content pages, etc. All the experiments in this study are done using a 5-folds cross-validation setup where all the images are randomly divided into five partitions. In each testing round, one fold is assigned to testing process and the rest is used for training. The average result of all testing rounds indicates the final performance. One fold of the training set was used as the validation set in case we needed to find any parameters.

Table. I presents the comparison results of our method and the proposed method by Kumar et al. [8] using their database. Their proposed features are called horizontal-vertical pooling (HVP) with 6300 dimensions and the accuracy of their detection system is tested using two classifiers; SVM and RF. From the table, it is obvious that our proposed algorithm outperforms their method in terms of accuracy using fewer features and same classifiers.

The analysis of the errors made by the proposed algorithm indicates two points. The first point is related to the complexity of our second database. For example, our database includes index and content pages which are structurally similar to the tables specially NRL-T ones but it is hard for the systems working only according to layout analysis to detect them. There are also figures that have a tabular scheme in them and are detected as false positives. Moreover, pages with several columns are sometimes detected as tables since they have a tabular property. The second point is related to the limitation our proposed algorithm. In the presence of noises or huge white spaces our proposed system can fail, which is one of the drawbacks of MFCCs itself. Also, the first preprocessing step of resizing images can cause problems when there is an image with a larger horizontal axis. False alarms in documents with pure text are quite rare unless there is a big font size difference with an unusual gap between the words occurs in the image. Some examples of the false positive and false negative images made by the proposed algorithm are given in Fig. 6 (a and b).

IV. CONCLUSION

We have proposed a new method for classifying historical document images according to the existence of tables in them. This method relies on using a well-known speech feature

TABLE I: Experimental results and comparison with the proposed method by Kumar et al. on the first database.

<table>
<thead>
<tr>
<th></th>
<th>HVP SVM</th>
<th>MFCC based SVM</th>
<th>HVP RF</th>
<th>MFCC based RF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>93.2%</td>
<td>94.4%</td>
<td>97.4%</td>
<td>97.9%</td>
</tr>
<tr>
<td>Forms</td>
<td>97.6%</td>
<td>98.51%</td>
<td>98.9%</td>
<td>99.50%</td>
</tr>
</tbody>
</table>

TABLE II: Experimental results using the second database.

<table>
<thead>
<tr>
<th></th>
<th>SVM</th>
<th>RF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recall</td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>RL-Ts</td>
<td>80.42%</td>
<td>89.51%</td>
</tr>
<tr>
<td>RL-Ts + NRL-Ts</td>
<td>71.15%</td>
<td>84.71%</td>
</tr>
</tbody>
</table>
called MFCCs. The implementation of the proposed method using two different databases demonstrates that our algorithm can detect tables with relatively high accuracy especially considering the challenging nature of our second database. Also, against most of the state of the art methods, it is capable of detecting both table categories of RL-Ts and NRL-Ts.

We plan to improve our proposed method in several ways. Adding image enhancement in the preprocessing step can help in dealing with the noisy images, which is a common problem in the historical document images. Conducting the experiments with respect to characteristics of the image and operational parameters also can lead to improvement of the results. Another path is to directly feed MFCCs and LogMS features as the input to a system that can handle three-dimensional features like deep learning architects. It is also good to expand our feature selection to consider other properties beyond the spectral representation. Moreover, we are planning to extend the algorithm to be able to locate the tables as well.

ACKNOWLEDGMENT

This publication was made possible by a grant from NSERC Canada and SSHRC Canada for “The Visibility of Knowledge” project.

REFERENCES


Abstract— The relation between handwriting and Alzheimer’s Disease (AD) as well as Parkinson’s Disease (PD) has been studied so far. However it is just in the last 5-6 years that Computer Aided Diagnosis system have been considered. Of course the current stage of development is still in the direction of healthy vs. non healthy classification. This work intends to provide a hands-on description from a pattern recognition perspective: state of the art on on-line data acquisition, feature extraction and classification is described. The paper also highlights the most profitable research direction in order to arise the community interest within the specific task.

Keywords—on-line handwriting Analysis, AD assessment, PD assessment

I. INTRODUCTION

Handwriting deals with cognitive and perceptual-motor tasks [28]. Parkinson’s Disease (PD) and Alzheimer’s Disease (AD) result in changes on handwriting. Just for instance, micrographia (reduction in writing size) is typically associated with PD while dysgraphia (a progressive disorganization and degeneration of the various components of handwriting) has been observed in AD patients [17].

In the on-line domain the handwritten trait is represented as a sequence \(\{S(n)\}_{n=0, \ldots, N}\), where \(S(n)\) is the signal value sampled at time \(n\Delta t\) of the writing process \((0 \leq n \leq N)\), \(\Delta t\) being the sampling period (see fig. 1). The on-line acquisition gives the possibility to acquire useful dynamic info of the writing process.

Fig. 1. On-line acquisition ("●": pen-down; "•": pen-up; black dots: on pad samples; red dots: on air movement samples)

Studies on motor control in healthy and unhealthy people are available so far, but only recently, a growing research interest has arisen towards the possibility to automatically discriminate between impaired subjects and healthy controls (HC) having the aim of developing research in the direction of a Computer Aided Diagnosis (CAD) system.

This review is intended from a Pattern Recognition perspective based on data acquisition, feature extraction and classification. The most valuable results are reported along with the conclusions and future research directions.

II. DATA ACQUISITION

The data acquisition phase mainly deals with the selection of the acquisition device and of the writing/drawing tasks.

A. Device

Many devices are available ranging from electronic pens [19] to professional tablets, however since in some situation the use of an electronic pen on a digital screen could be unusual or unfamiliar to patients, writing with an inking pen on paper fixed to the tablet has been also considered [23], [33].

The main attributes acquired depends upon the specific tool, however typically acquired parameters are: \((x,y)\)-coordinates of the pen position, time stamps, pen orientation (azimuth and altitude) and pressure. Moreover, pen tablets also detect in air movements (when the tip is not in contact with the writing surface). Kinematics in the air can be obtained by taking into account the so-called button status, which is a binary variable for pen-up state (in-air movement) and for pen-down state (on-surface movement). Electronic (smart) pen can also acquire pressure of the fingers holding the pen [19].

B. Tasks

Acquisition tasks adopted by researchers can be classified as follow.

1) Simple tasks. Straight lines, spirals, as well as meanders and circles have been frequently used for motor performance evaluation in both PD and AD [8, 20, 25].

Non-sense words have been used: “eeee”, “el”, “l”, “lll”, “lln” [4]. Such characters are easy to be written in a recursive and continuous way; moreover, and their use minimizes the linguistic-comprehension processes. Simple words and short sentences have been also widely...
adopted: “hello hello”, “lektora”, “mamma” [3, 10]. Typically, words/sentences used in these tasks are chosen based on their simple orthography and easy syntax. In same situation the sentence contains words having a common “core” (e.g. “Ein helles grelles Licht” [29] or “The leveler leveled all levels” [31]) in order to verify how a common pattern is modified having or not prefix or postfix. Sentences include words with ascendant and descendent traits (e.g. “g” and “l”). A sentence requires a high degree of simultaneous processing, and it offers the possibility to widen evaluate motor-planning activity and hesitation or pause between two words [7]. A sentence also allows to record the effect of fatigue during writing [5, 6].

Handwritten signatures have been taken into account too [21].

2) Complex tasks. The drawing/handwriting task is part of a more complex task involving motor, cognitive and functional issues. Van Gemmert et al. [31] required to write a the already mentioned sentence sentence under four different conditions. Constrains on time duration of the writing task [68] as well as on the stroke dimension [27] have been considered too. In [2] Broeder et al. requested participants to count the number of heard tone while writing loops. Copying tasks have included the copying of the fields of a bankcheck [23, 33] as well as addresses, phone number, grocery list, etc. [9, 23, 33].

Complex tasks acquire a relevant importance in the AD case since AD is primarily characterized by cognitive deficits. For instance the Clock Drawing Test (CDT) [9, 15] is able to reveal visual-spatial deficits: in some case of dementia this deficit is evident since early stages. CDT, as well as many others complex tasks, involves various neuro-psychological functions: auditory perception, auditory memory, abstraction capacity, visual memory, visual perception, visual-space functions, programming and execution capacity.

### III. FEATURES

Function and parameter features can be considered. When function features are used, the handwritten trials are characterized in terms of a time function. When parameter features are used, the trials is characterized as a vector of elements, each one representative of the value of a feature.

#### A. Function Features

The most common function features are: (x,y) coordinates, time stamp, button status, pressure, azimuth, altitude, displacement, velocity and acceleration. Some of these features are directly conveyed by the acquisition device whereas others are numerically derived (Table 1). It is not surprising to note that the most used are velocity (often called speed) and acceleration: the first conveys information related to the slowness of PD and AD movements, while changes on the acceleration profile are able to reveal tremor.

In air based features (coordinates, azimuth, altitude, velocity, acceleration) can be considered for timestamps having the button status b(t)=0. This is a very important aspect since it has been recently demonstrated to convey very useful information [15, 16] since the pen must be hold without a support: tremor and hesitation are much more evident than in the case of the pen on the pad.

#### B. Parameter Features

Parameter features are obtained by means of transformations upon the function features (see Table 2 for details). Some parameters have been specifically designed with the aim to perform AD and PD analysis. Among the others, movement in air and on the pad while performing a task has been observed to increase as task length and difficulty increase while other values (e.g. pressure) remain constant. When a copy task is considered, the in-air time reflects hesitations of AD patients.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Source</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>Device</td>
<td>[5, 6, 7, 19, 23, 33]</td>
</tr>
<tr>
<td>Button Status</td>
<td>Device</td>
<td>[5, 6, 7, 23, 33]</td>
</tr>
<tr>
<td>Pressure</td>
<td>Device</td>
<td>[5, 6, 7, 9, 30, 23]</td>
</tr>
<tr>
<td>Azimut</td>
<td>Device</td>
<td>[5, 6, 7, 23, 30, 33]</td>
</tr>
<tr>
<td>Altitude</td>
<td>Device</td>
<td>[5, 6, 7, 23, 30, 33]</td>
</tr>
<tr>
<td>Velocity</td>
<td>Calculated</td>
<td>[2, 3, 4, 9, 10, 21, 24, 25, 29, 33, 35]</td>
</tr>
<tr>
<td>Acceleration</td>
<td>Calculated</td>
<td>[4, 8, 9, 29, 32]</td>
</tr>
<tr>
<td>Gyroscope</td>
<td>Device</td>
<td>[19]</td>
</tr>
</tbody>
</table>

Parameter features can be evaluated at global (task level) or even at local level (typically at stroke level). A stroke is generally considered as a single component of the handwritten trait which is connected and continue: (see fig. 2).

![Fig. 4. On-pad strokes of the word in fig. 1.](image)

The number of strokes per second can be considered to be representative of the handwriting frequency: in AD patients a significantly low writing frequency has been observed.

Jerk (which characterizes PD) can be measured in terms of Number of Changes in Acceleration (NCA); it is often taken into account with the Number of Changes in Velocity (NCV). Since jerk/tremor introduces “noise” on the handwriting signal, entropy and energy features have used and valuated starting from the (x,y) coordinates adopting well known Shannon and Rény operators.

Empirical Mode Decomposition has been applied [5]. EMD is able to decompose a signal within finite and small number of components which convey info related to the most oscillating (high-frequency) part of the signal. Although many well-known transforms could be considered (e.g. Fourier, cepstrum, etc.), and although they could be useful for a deep frequency analysis of signals, their use seems to have not been considered up to date.

Features based on the kinematic theory of rapid human movement have been also considered by adopting the Sigma-Lognormal model to represent the information of both the motor commands and timing properties [21].

Finally, in order to have statistical representation of the available function features, max, min, means, standard deviation, range and median have been considered too.
IV. Classification

For the sake of simplicity results are discussed separately for PD and AD.

A. PD CAD design

PD is usually diagnosed by the first motor symptoms in particular slowness [3, 27, 31], reduction in amplitude of repeated actions (bradykinesia) and micrographia [2, 20, 32], tremor and rigidity [13, 20, 26, 13] are observed. PD patients, if compared to controls, generally write smaller letters apply less pressure and require more performance time. However it must be underlined that not all the mentioned characteristics have been simultaneously observed under whatsoever task. Just for instance micrographia or reduction in letter size has been observed within longer words or within signatures or sentences [32] and not on a sequence of few characters [1].

PD also results in cognition, planning and execution impairments [12]. Complex tasks can be used to reveal such characteristic [2, 31].

Although studies about the correlation of handwriting and PD are so far available, it is only in the last 5-6 years that these evidences have been applied in order to get a Computer Aided Diagnosis (CAD) system. In general the task has been the classification of PD patients vs. healthy. In air and on surface features have been considered as well as Support Vector Machine – SVM [5, 6, 7], Discriminant Analysis [23], Convolutional Neural Network [19] and Naïve Bayes [13] have been successfully used. Table 4 summarizes results.

B. AD CAD design

It has been observed that at the beginning, fine motor control and coordination [22, 34] are impaired. Moreover the maxima speed value is almost regular in healthy persons while it results to be strongly reduced at the beginning of the disease and completely lost in the advanced stadium [10]. In general, AD patients results in slower, less smooth, less coordinated and less consistent handwriting movements than their healthy counterparts.

Details about features, classifiers and results are reported in table 5. Among the others, Handwritten signatures have been demonstrated to be a useful, an ERR (Equal Error Rate) of 3% has been achieved [21]. Other interesting tasks are copying tasks (including two and three dimension figures) [9].

V. Conclusion and future work

Handwriting based tasks can be successfully used for the assessment of AD and PD. The pattern recognition community has provided an effort in the direction of a CAD system just in the last 5-6 years: there is a call for research still completely open.

First of all the most part of experiment has been led on private datasets. These are different in tasks, size, acquisition device, etc.. Only very few and dataset are currently available [5, 11, 19, 21]. The lack of a big dataset strongly limits the research development. In general task already considered only refers to draw or write: finger taps [18] should also be considered: just think to the daily use of smartphones and the connected potentialities.

In general, the pattern recognition community has provided a minor contribution to the field, and many issues still can be inspected: segmentation, and classifiers.

Acknowledgment

This work is supported by the Italian Ministry of Education, University and Research within the PRIN2015 - Handwriting Analysis against Neuromuscular Disease - HAND Project under Grant H96J16000820001.
Participants

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Features</th>
<th>Classifier</th>
<th>Results</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>37 PD 38 EC</td>
<td>Spiral drawing, repetition of “f”, “le”, “les”, three Czech words, the sentence “Tramvaj dnes už nepo-jede”</td>
<td>SVM</td>
<td>Accuracy = 88.1%</td>
<td>[5]</td>
</tr>
<tr>
<td>24 PD 20 EC</td>
<td>Entropy, signal energy, empirical mode decomposition (on-surface) + feature selections</td>
<td>SVM</td>
<td>AUC = 89.09%</td>
<td>[6]</td>
</tr>
<tr>
<td>20 PD 20 EC</td>
<td>Stroke height/width, duration, writing length, NCP, Energy, EMD</td>
<td>SVM</td>
<td>Accuracy = 82.5%</td>
<td>[7]</td>
</tr>
</tbody>
</table>

TAB. IV
PD CAD SYSTEMS
ABBREVIATIONS: PD = PARKINSON’S DISEASE PATIENTS; EC = ELDERLY CONTROLS; T = TABLET; ST = SHEET OF PAPER FIXED ON THE TABLET; EP = ELECTRONIC PEN; AUC = AREA UNDER THE ROC; DA = DISCRIMINANT ANALYSIS; SVM = SUPPORT VECTOR MACHINES; CNN = CONVOLUTIONAL NEURAL NETWORKS; NB = NAÏVE BAYES.

TABLE V
AD CAD SYSTEMS
ABBREVIATIONS: AD = ALZHEIMER’S DISEASE PATIENTS; EC = ELDERLY CONTROLS; T = TABLET; ST = SHEET OF PAPER FIXED ON THE TABLET; DA = DISCRIMINANT ANALYSIS.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Features</th>
<th>Classifier</th>
<th>Results</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>23 AD 17 EC</td>
<td>Dictated sentence writing, free sentence writing, two and three dimensions drawing, clock drawing</td>
<td>Pressure, time, velocity, acceleration, energy, complexity</td>
<td>DA</td>
<td>Accuracy = 72%</td>
</tr>
<tr>
<td>20 AD 20 EC</td>
<td>Time-in-air, time-on surface, total time</td>
<td>Logistic regression</td>
<td>AUC = 0.925</td>
<td>[15]</td>
</tr>
<tr>
<td>20 AD 20 EC</td>
<td>Time-in-air, time-on surface, total time</td>
<td>Logistic regression</td>
<td>Accuracy = 87.2%</td>
<td>[16]</td>
</tr>
<tr>
<td>22 AD 41 EC</td>
<td>Copying: a phone number, a grocery list, the details of a check, the alphabet sequence and a paragraph</td>
<td>Size, duration (on-paper time and the in-air), pressure, mean velocity, mean pressure</td>
<td>DA</td>
<td>Accuracy = 97.5%</td>
</tr>
</tbody>
</table>

REFERENCES
Conf. on e-Health and Telemedicine, 2014, pp. 171-175.


Image Quality Enhancement For Document Extraction with Mobile Devices
Van Cuong Kieu\textsuperscript{1}, Florence Cloppet\textsuperscript{1}, Vincent Poulain d’Andecy\textsuperscript{2}, Saddok Kebairi\textsuperscript{2}, Nicole Vincent\textsuperscript{1}
1. LIPADE laboratory, Paris Descartes University, 75006, Paris, France
2. Research team, Yooz-ITESOF company, Parc d’Andron - Le Séquoia, 30470, Aimargues, France

Abstract—Mobile devices simply and personally enable a new image acquisition to build large scale datasets for research and application. However, it is challenging to any automatic process by introducing degradation such as blur, distortion, or illumination defect. Therefore, we present a novel method to extract document in captured images. This method includes techniques to deal with the variability of complex background, illumination defect, blur, and to reconstruct documents for document analysis and recognition. The method is evaluated on two captured datasets.

Keywords—document image quality, blur correction, smart camera acquisition, document extraction, dominant color, background removal

I. INTRODUCTION

At the moment, the mobile devices are more and more used and they nearly all enable to snap some photos with a large number of pixels. Unlike a fixed scanner, mobile devices are always at hand in any environment to capture a digital image of any document. However, the lack of movement and illumination control leads to various degradations due to acquisition process such as complex background, character deformation\textsuperscript{[1]}\textsuperscript{[2]}\textsuperscript{[3]}, document distortion\textsuperscript{[4]}, and blur\textsuperscript{[5]}. Numerous current techniques have failed to deal with such degradation\textsuperscript{[6]} \textsuperscript{[7]}.

To understand degradation conducted by mobile devices, we carried out an analysis on a private database\textsuperscript{1} which consists of 196 images from different devices: Ipad2, Iphone5, Nexus4 and GalaxyS3. Types of documents include invoices, A4 forms and receipts. Fig. 1 shows degradation distribution in the corpus. Unsurprisingly, background problem is present in almost all of document images, perspective in 50\% of these images, illumination defect and blur follows with 37\%, and distortion and cropped document share the same proportion of 18\%. We classified only the dominant degradation, but in fact, several problems of degradation locally and heterogeneously appear in one document image. Such degradation is introduced during image acquisition. This analysis correlates with the two published databases DIQA\textsuperscript{[6]} and SmartDoc\textsuperscript{[7]}.

The impact of degradation both on visualization and document analysis has been reported in\textsuperscript{[1]}\textsuperscript{[2]}\textsuperscript{[3]}\textsuperscript{[4]}\textsuperscript{[5]}. Therefore, document degradation model and document image quality enhancement have been investigated for decades\textsuperscript{[8]}. Recently, several works are dedicated to degradation on captured document images\textsuperscript{[6]}\textsuperscript{[7]}\textsuperscript{[9]}. Few studies are concerned by mobile acquisition itself\textsuperscript{[10]}. As mentioned in\textsuperscript{[9]}, the high variability of illumination effect (\textit{e.g.} glossy backgrounds and background with flash effect) leads to the difficulty in extracting the document. Therefore, in this paper, we focus on illumination correction on non-uniform background for extracting document and on blur correction in order to improve the performance of document analysis and recognition.

Fig. 1: Degradation frequency occurrence (in red) within our corpus

Fig. 2: Pipeline of document quality improvement for document analysis with mobile devices.

II. ILLUMINATION CORRECTION

Illumination correction is the first and crucial step in document image pre-processing. Several global classic methods can be used such as contrast improvement.
Nevertheless, they cannot handle local and heterogeneous effects (e.g. Fig. 4). Thus, in this section, we propose to consider the optical system modeling to correct illumination defect.

The acquisition process is modeled as shown in Fig. 3. In general, the source light $\vec{L}$ comes to the document surface. One part $\vec{L}$ of the light reflexes and comes to the optic system of camera, the other part – the transmission part $\vec{T}$ comes through the document. The information retrieved by camera is the intensity of coming light $\vec{L}$ and the color of all objects. The very high intensity of $\vec{L}$ will overshadow the color of object (the red ellipse in Fig. 4-a) whereas the low value produces a dark region (the blue ellipse in Fig. 4-a).

![Fig. 3: Document image acquisition process: $\vec{L}$ is light coming to document, $\vec{L}'$ is light coming to camera, $\vec{T}$ is transmission light.](image)

Let $(l_0, p_0)$ and $(l, p)$ be the intensity of the coming light and object color before and after sampling for a pixel P, respectively (Fig. 3). The study [13] already proved the existence of an exponential factor $\lambda$ which satisfies the following equation:

$$ l = M - M \times (1 - \frac{l_0}{M})^\lambda $$

Where $M$ is the number of values of luminosity, here $M$ is equal to 256. From (1), the factor $\lambda$ can be calculated as follows:

$$ \lambda = \frac{\ln(1 - l/M)}{\ln(1 - l_0/M)} $$

Therefore, the relation between the intensity of luminosity and the color of the pixel $P$ is as follows:

$$ p = M - M \times (1 - \frac{p_0}{M})^\lambda $$

![Fig. 4: (a) Original image](image)

![Fig. 4: (b) The illumination correction for the image (a)](image)

Equations (2) and (3) in [13] suggest that by estimating the value $l_0$ we can approximate the true color of the pixel $P$. Normally, luminosity is homogeneous in a good quality image and it is equal to the average of all bright pixels in the image (i.e. a bright pixel has luminosity value greater than 50). As a result, we can choose this average value as the value of $l_0$ in order to correct illumination defect.

Fig. 4-b shows the transformation of the initial image in Fig.4. The white region inside the red ellipse and the dark region inside the blue ellipse are corrected.

Thanks to the illumination correction, we can now extract the document in the image by methods presented in the next section.

### III. DOCUMENT EXTRACTION

Several basic extraction methods will be presented in the first sub-section. In the second one, we show how a document shape is rectified with respect to the rectangular shape.

In general, a document image $I$ can be modeled as follows:

$$ I = \bigcup_{i=1}^{F_i} F_i \cup B $$

Where $F_i$ is a zone of interest (e.g. a document), $B$ is the background where $F_i$ lies on. Let $D$ and $T$ be the set of rectangular documents and the transformation operation (i.e. rotation) to obtain a corrected document.

$$ D = \bigcup_{i=1}^{T} T(F_i) $$

Normally, $k$ is small. In this paper, we make the assumption that there is only one document in the center of the image (i.e. $k=1$).

#### A. Document coarse detection

The document usually has a dominant background color (e.g. white). Thus, it can be extracted by color segmentation.

1) **Binarization:** a simple way to extract the document is to use a binarization method. Thanks to the illumination correction, the global threshold methods such as OTSU is appropriate to separate the document and background because the document is a large object in the image.

2) **Dominant color segmentation:** the dominant color is extracted from the 3D color histogram where a distance has been defined. The mode of this raw 3D histogram gives the most present color $c_R$, but the neighboring colors are also considered as belonging to the dominant color. A threshold that has been empirically defined on a base of 338 document images defines the neighborhood. As in [20], the distance of a color $c$ to $c_R$ in the image $I$ is defined by:

$$ d_{l,c_R}(c) = 1 - \exp\left(-\frac{\|c-c_R\|_2}{2m}\right) $$

where $m$ is the mean of the squared distances between pixels' color and $c_R$. Thus, the size of the neighborhood is varying according to the image content. The domain associated with the dominant color ($c_R$ chosen as the histogram mode) can be defined as:

$$ D_R = \{ x \in I | d_{l,c_R}(c(x)) < \delta \} $$

where $c(x)$ gives the x pixel color.
Fig. 5: (a) initial image, (b) illumination correction, and (c) the extracted document thanks to dominant color.

Fig. 5-b and c show the illumination correction image and the extracted document of the image Fig. 5-a, respectively.

B. Refinement of document detection

At this stage, an image is decomposed as I = F ∪ B where F is the document lying on the background B. We have developed three ways to get a quadrilateral shape of F: the first is a local approach based on the contour approximation, the second is based on the optimization of a global criterion, and the third is to take advantage of regularity and symmetry of the contours.

1) Polygonal approximation: from the binary shape of the document, the contour pixels are extracted and sequenced. Then a polygonal approximation is performed using Douglas-Peucker method [21]. To obtain a limited number of segments, the threshold involved in the polygonal approximation has to be tuned. This is the main drawback of local approaches; a low threshold value will keep too much irregularities of the document mask contour, while a higher value will get rid of points that are useful to describe correctly the shape. This can be difficult to handle, hence we propose a global approach.

2) Optimization by Genetic Algorithm (GA): as global approach, we have to express the criterion optimized in order to find four points P={P1, P2, P3, P4}, the candidate vertices of the document. The quadrilateral defined by the four points, denoted as Q(P), must contain as many pixels of the document as possible and as few pixels of the background as possible. Then we defined the function:

\[ f(P) = \lambda \cdot \frac{\mathcal{A}(F \cap Q(P))}{\mathcal{A}(F)} + (1 - \lambda) \cdot \frac{\mathcal{A}(Q(P))}{\mathcal{A}(Q(P))} \]

To define the best four corners, we made use of a genetic algorithm. The chromosomes are binary with size of (28 x 4) and interpreted as eight real numbers. The crossover operator is a one-point crossover. At each generation, the parents are selected through a three member tournament. From experiments, the quadrilateral corresponding to the best member in the population after fifty generations is extracted to limit computation time.

Fig. 6: in (a) the initial document mask, in (b) polygonal approximation in red, in (c) quadrilateral obtained with the GA in green.

3) Symmetric document contour approximation (SCA): a rectangular and flat document has vertical and horizontal symmetry axes which pass through the symmetry center of this document. If the document is distorted by a small local distortion, the document symmetry property can be recovered.

The document contour is approximated to be a quadrangle \( P_1 P_2 P_3 P_4 \) by Douglas-Peucker method [21]. The quadrangle divides the contour into four parts: two ‘horizontal’ contour parts (\( C_1 \) and \( C_2 \)) and two ‘vertical’ parts (\( C_3 \) and \( C_4 \)). We define \( M_{ij} \) as the center of the segment \( P_i P_j \) where \( P_i \in C_1 \) and \( P_j \in C_2 \) in providing that \( P_i P_j \parallel P_2 P_3 \). The horizontal symmetric axis \( \theta_{12} \) between \( C_1 \) and \( C_2 \) can be extracted by linear regression from the set of \( M_{ij} \) (see Fig. 7).

A pair \( P_i P_j \) is said ‘correct’ if the distance from the center \( M_{ij} \) to \( \theta_{12} \) is less than \( \varepsilon \). If the distance is greater than \( \varepsilon \), let \( M'_{ij} \) be the intersection of \( P_i P_j \) with \( \theta_{12} \). Let \( P_a P_b \) be the nearest ‘correct’ pair of \( P_i P_j \). If \( |P_i M'_{ij} - P_a P_b/2| < |P_j M'_{ij} - P_a P_b/2| \), we define \( P_i' \) so that \( P_i P_i' \) is a correct pair. Inversely, we define \( P_i' \) so that \( P_i' P_j \) is a correct pair.

Fig. 7: Horizontal symmetric axis detection to correct document contour: \( C_1 \) and \( C_2 \) are two horizontal parts, \( \theta_{12} \) is the horizontal symmetric axis, and the green part of \( C_2 \) is the corrected part.

In the same way, the vertical symmetric axis \( \theta_{34} \) is extracted and two vertical contour parts \( C_3 \) and \( C_4 \) are corrected. Fig. 8-d shows a result of the correction contour process.

Fig. 8: Example of contour correction: (a) original image in SmartDoc database, (b) illumination corrected image, (c) binarized image, and (d) corrected document contour (green).

C. Rectified document

The three methods enable to retrieve the document vertices. According to the position of the paper with respect to the camera, the deformation of the document is more or less important. Some edges can be either convex or concave. The bended support, where the document lies on, and the
Blurring process comes from the color of textual pixels diffusing to the region around and vice-versa. Basically, this process is modeled by a mathematical linear model $I = H \times I_0 + N$ where $I$ is a blurred image, $H$ is a blur model, $I_0$ is the non-blurred image, and $N$ is an additive noise. Many studies have been done by using this model [5][12]. But it does not solve completely the blur problem because this global and linear model is not suited to local and heterogeneous blur. Thus, in this section, we describe briefly a local and non-linear blur model (refer to [14] for more details of the method). The method concentrates on the blur that affects the textual zones as this blur has the most harmful consequences.

A. Blur model

In a document image, the black color of textual pixels spreads over its neighbors; therefore, we first localize blur regions by detecting textual zones $Z$ in the image. To do it, we define a binary mask resulting from binarizing the image by NICK method [11]. Each connected component in the binary image is dilated in an extended component, and each extended component defines a textual zone in the original image (Fig. 10).

Thus, in this section, we describe briefly a local and non-linear blur model (refer to [14] for more details of the method). The method concentrates on the blur that affects the textual zones as this blur has the most harmful consequences.

In a document image, the black color of textual pixels spreads over its neighbors; therefore, we first localize blur regions by detecting textual zones $Z$ in the image. To do it, we define a binary mask resulting from binarizing the image by NICK method [11]. Each connected component in the binary image is dilated in an extended component, and each extended component defines a textual zone in the original image (Fig. 10).

In order to cluster this zone into blur and non-blur pixels, we define a model of the two classes. Each pixel in the zone more or less belongs to the blur class according to a membership value. This value is considered as its blur level. In order to estimate the membership value of a pixel to blur class $\mu(P)$, we extract a blur feature named $\alpha$-feature:

$$\alpha(P) = \begin{cases} 
\frac{p - G_{\min}}{G_{\max} - G_{\min}} & \text{if } p < G \\
\frac{G_{\max} - p}{G_{\max} - G_{\min}} & \text{if } p \geq G 
\end{cases} \quad (3)$$

where $p$ is the gray value of $P$ pixel, $G_{\min}$ and $G_{\max}$ are the minimum and maximum gray values in the region, respectively. The threshold $G$ gives the highest value to the $\alpha$-feature and models the highest blur level. Thus, this threshold can be estimated as the threshold of a binarization method for the textual zone. This is because the pixels having this value may belong to text or to background with some ambiguity, in other words, such pixels are strongly blurred.

A fuzzy-c-means clustering (FCM) with two classes (blur and non-blur) is initialized by $\alpha$-feature. At the final state of FCM, the $G_{\min}$ and $G_{\max}$ determine two values $\mu_{BG}$ and $\mu_{FG}$ with respect to the blur class (Fig. 11). Those values enable to divide the textual zone into three regions: background (BG), foreground (FG), and transient regions. The transient region contains mostly pixels having high membership values to the blur class, therefore it is considered as blur region.

Let $BL$ be the set of pixels in the blur region. Fig. 12 shows three examples of blur regions (blue).

- Non-blur region $BG \cup FG$ where $BG = \{P, \mu(P) < \mu_{BG} \& p > G \}$ and $FG = \{P, \mu(P) < \mu_{FG} \& p < G \}$.
- Blur region $BL = Z \setminus (BG \cup FG)$.

In document image, the blur correction process is often to determine which pixels belong to background and which ones belong to foreground with respect to OCR accuracy [6]. In this trend, we define a threshold $T$ linked to a local window $W$ (w x w) the center of which is a blurred pixel in the blur region BL. The threshold $T$ corresponds to the solution of the minimization of the following function:

$$T = \text{Argmin}_T \{ |AB \cdot \Delta F| \}$$

$$\Delta F = \sum_{P \in BL} \chi_t(p)(p - t)$$

$$\chi_t(p) = \begin{cases} 
1 & \text{if } p > t \\
0 & \text{otherwise}
\end{cases}$$
Fig. 13. Example of the correction process on motion blur document image (a): result with OTSU (b) and with local window (c), (d).

Table 1: The number of detected documents of 4 methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>#Found document</th>
<th>#Missed document</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polygonal approximation</td>
<td>141 (71.9%)</td>
<td>55 (28.1%)</td>
</tr>
<tr>
<td>GA optimization</td>
<td>151 (77.1%)</td>
<td>45 (22.9%)</td>
</tr>
<tr>
<td>ABBYY11 mobile</td>
<td>158 (80.6%)</td>
<td>38 (19.4%)</td>
</tr>
<tr>
<td>Illumination + SCA</td>
<td>184 (94%)</td>
<td>12 (6%)</td>
</tr>
</tbody>
</table>

The two first lines which do not involve illumination correction are not efficient when the image background has a similar color to the document background itself (i.e. notice that ABBYY-11-mobile has similar failures on clear backgrounds). Hence the dominant color detection merges both information. Beside the management of the number of colors enabling to recover the document is difficult, in some cases, the second dominant color is more relevant than the first. The second dominant color helps to detect the background. In last line, we obtain better results than ABBYY due to the illumination correction and spatial document properties that enables a binarization to recover the document in a correct way.

C. Blur correction

1) Measure: the OCR accuracy of DIQA by Tesseract 3.02 before and after blur correction is used to evaluate the performance. To compare with another method, we implement three adaptive binarization methods as three naïve blur correction methods (Niblack [15], Sauvola [16] and Wolf [17]), two classic non-blind deconvolution methods (Richardson-Lucy [19] and Wiener [18]) and a recent deblurring [5] method. The three adaptive methods are applied with the same window size of 33 as in our method. The two non-blind deconvolution methods use a same Gaussian kernel (i.e. its structure size is nine and its standard deviation is equal to 2.5). The authors in [5] suggest to use a large kernel size of 135 and a Gaussian filter for real database such as DIQA.

2) Blur estimation: in order to analyze the results, we also applied the local blur estimation proposed in [14]. For more details on this estimation, we refer to [14]. The estimated blur level in the 25 datasets of DIQA varies from lowest level of 0.232 to highest level of 0.981.

3) Blur correction: Fig. 14 shows the OCR accuracy baseline in black and that of 7 methods on DIQA dataset after the blur correction. The results report an overall improvement of 11% by our method in comparison with the baseline. The maximal retrieved improvement is equal to 30.6%. The proposed method improves significantly OCR accuracy of document images whose blur level varies from low to relatively high. Indeed, Fig. 14 shows that from the blur level of 0.232 to 0.679, the improvement of OCR accuracy is about 9.75%. The proposed method gives a large improvement of 12.85% even with high blur level (i.e. from 0.679 to 0.981).

The use of binarization methods in document analysis and recognition leads to some loss of information. Indeed, the OCR accuracy of binarized images by Niblack and Sauvola methods is rapidly degraded when the blur level increases. The Wolf’s binarization method appears to be suited for blurred images in DIQA dataset when the blur level is small (i.e. it improves the OCR accuracy about 5% on average when blur level is smaller than 0.45). However, it cannot deal with the high blur levels.

VI. CONCLUSION

The efficiency and ease of mobile devices used in data collection change the way of image acquisition, especially for documents. However, the document analysis and recognition methods adapted to the old acquisition fashion can fail because of the lack of image quality control in the new acquisition way. Therefore, we presented in this paper a
pipeline to deal firstly with the illumination defect and distortion in order to extract documents and then with blur to improve the performance of OCR. Thanks to local approaches, the illumination correction based on the relation between color and luminosity at pixel level can deal with the heterogeneity and variability of light sources in the image. To correct blur, we proposed a local non-linear model based on a fuzzy clustering. Indeed, the illumination correction helps to extract documents correctly up to an accuracy of 94%, and the blur correction improves the accuracy of OCR up to 11%.

**Fig. 14:** OCR accuracy vs. the increase of blur level in the DIQA database

**REFERENCES**


Improving Chinese Writer Identification by Fusion of Text-dependent and Text-independent Methods

Yu-Jie Xiong\textsuperscript{1}, Li Liu\textsuperscript{2}, Patrick S. P. Wang\textsuperscript{3} and Yue Lu\textsuperscript{1}

\textsuperscript{1}Shanghai Key Laboratory of Multidimensional Information Processing  
Department of Computer Science and Technology  
East China Normal University, Shanghai 200062, P. R. China  
\textsuperscript{2}School of Information Engineering  
Nanchang University, Nanchang 330031, P. R. China  
\textsuperscript{3}Northeastern University, Boston, MA 02115, United States

xiong@stu.ecnu.edu.cn, liuli_033@163.com, patwang@ieee.org, ylu@cs.ecnu.edu.cn

Abstract—A novel method for Chinese writer identification is proposed in this paper, which takes the advantage of both text-independent and text-dependent characteristics. The contour-directional features are extracted from the whole image. They are used to calculate the text-independent similarity between the query and reference handwriting images. Meanwhile, character pairs appearing in both the query and reference handwriting images are utilized for computing the text-dependent similarity. We propose an effective method to measure the similarity of character pairs. It is rooted from image registration. The displacement field used to align two characters is calculated using Log-Demons algorithm, and is utilized for similarity measurement. The final similarity between the query and reference handwriting images is the fusion of text-independent and text-dependent similarities. The proposed method is evaluated on the HIT-MW and CASIA-2.1 datasets. The best Top-1 identification accuracy on the HIT-MW and CASIA-2.1 datasets reaches 97.1\% and 98.3\% respectively, which outperforms other previous approaches.

Index Terms—Chinese writer identification, text-independent, text-dependent, Log-Demons, similarity fusion.

I. INTRODUCTION

Writer identification is to identify the authorship of handwritings. It can be generally classified into two categories: text-independent and text-dependent [1]. Text-independent methods are based on the handwritten data with unrestricted text content. They do not focus on particular characters or words, but try to solve the problem by analyzing the writing style of handwriting. Thus, text-independent methods seem to be more applicable. Furthermore, text-independent methods treat each character of handwriting equally, in which all characters contribute to the writing style. However, text-independent methods require a certain minimum amount of text to produce a reliable decision. On the other hand, text-dependent methods require that the text content of the query and reference handwriting images are the same. These methods take the advantage of direct comparison between characters of the same text content in both the query and reference handwriting images.


When we concentrate on the issue of Chinese writer identification, we find that some characters of the same text content appear in both the query and reference handwriting images in most cases. Two characters containing the same text content in both the query and reference handwriting images are defined as a character pair. In this study, our motivation is to improve text-independent Chinese writer identification utilizing the text-dependent information, i.e, character pairs appearing in both the query and reference handwriting images. Thus, the proposed method takes the advantages of both text-dependent and text-independent methods. In the text-independent part, we extract the contour-directional feature to represent the writing style of handwriting. The weighted Chi-squared metric is used to measure the text-independent similarity between the query and reference handwriting images. In the text-dependent part, the character images of character pair are treated as the fixed image and the moving image, and the displacement field of them are calculated using Log-Demons algorithm. The energy of the
II. THE PROPOSED METHOD

The text-independent similarity between the query and reference handwriting images is calculated using contour-directional feature. Meanwhile, the text-dependent similarity is calculated by energy based similarity measurement. Both similarities are fused to obtain the final similarity between the query and reference handwriting images.

A. Text-independent method

Text-independent writer identification extracts robust features from the handwriting for pattern representation, which is unrestricted to text content. We utilize the contour-directional feature presented in our previous work [10] to describe the individual writing style of handwriting.

1) Contour-directional feature (CDF): The contour-directional feature utilizes the distribution of pixel pairs based on the directional information to represent the individual trait of the writer. The contour-directional feature is extracted from the contour image, thus contour detection needs to be performed at first. The Sobel operators are employed to generate the contour image from the query handwriting image or reference handwriting image. Then, the distribution of edge pixel pairs in the contour image is recorded as the feature of handwriting. To obtain edge pixel pairs, the contour image is divided into a number of grids of size $n \times n$, and the center of each grid is every edge pixel. A grid of $5 \times 5$ is illustrated in Fig. 1(a). The black square is denoted as an edge pixel $P$, and gray squares are denoted as edge pixels connected to $P$. Each pixel $S$ in the grid is marked with a symbol $G(S) = D_i$, where $D$ denotes the larger distance in the horizontal and vertical distances between $S$ and $P$. For each value of $D$, there are $8+D$ pixels around the center $P$ and they are assigned from $D_1$ to $D_{8+D}$. All the pixel pairs $(\alpha, \beta)$ which satisfy the following conditions are recorded.

$$
\begin{align*}
\alpha & \text{ and } \beta \text{ are edge pixels,} \\
G(\alpha) & = D_i, \ G(\beta) = D_j, \text{ and } i < j, \\
G(\gamma) & = D_k, \ i < k < j, \text{ and } \gamma \text{ is not an edge pixel.}
\end{align*}
$$

We define the direction $Dir(S)$ as: $\arctan \frac{S_y - P_y}{S_x - P_x}$, where $(S_x, S_y)$ and $(P_x, P_y)$ are coordinates of $S$ and $P$. Afterwards, we update the symbol of each pixel in the grid according to the direction of the pixel. The new symbol of $S$ is denoted as $C(S)$, and rules of updating are below:

$$
\begin{align*}
& \text{If } Dir(S_1) \text{ is unique, and } G(S_1) = D_i, \text{ then } C(S_1) = G(S_1) = D_i. \\
& \text{If } Dir(S_1) = Dir(S_2) = \cdots = Dir(S_n), \ G(S_1) = D^{a_1}, G(S_2) = D^{b_1}, \ldots, G(S_n) = D^{c_k}, \text{ and } D^{a} < D^{b} < \cdots < D^{c}, \\
& \text{then } C(S_1) = \cdots = C(S_n) = G(S_1) = D^{a_1}.
\end{align*}
$$

As shown in Fig. 1(b), changed symbols are labeled with red color, and the edge pixel pairs $(1_2, 1_4), (1_2, 1_4), (1_4, 2_10), (2_10, 2_12), (2_12, 1_7)$ are recorded as the contour-directional feature. When the center of the $n \times n$ grid has traversed every edge pixel in the contour image, all edge pixel pairs appearing in the grid are also recorded. After the normalization, the frequency histogram of edge pixel pairs is generated as the contour-directional feature vector.

2) Similarity calculation: After the contour-directional features are extracted from the query and reference handwriting images, the weighted Chi-squared metric is used to calculate the similarity. Assume that there are $L$ reference handwriting images, and the query handwriting image and reference handwriting images are denoted as $Q$ and $R_i \ (1 \leq i \leq L)$, respectively. Let $CDF_{Q}$ ($\{a_1, a_2, \ldots, a_N \}$) and $CDF_{R_i}$ ($\{b_1, b_2, \ldots, b_N \}$) denote their contour-directional features. The distance $D_C$ between $Q$ and $R^i$ is computed by:

$$
D_C = \sum_{j=1}^{N} \frac{(a_j - b_j)^2}{(a_j + b_j) \sigma_j},
$$

where

$$
\sigma_j = \sqrt{\frac{1}{L - 1} \sum_{i=1}^{L} (b_j - \mu_j)},
$$

and

$$
\mu_j = \frac{1}{L} \sum_{i=1}^{L} b_j.
$$

B. Text-dependent method

Text-dependent method is a one-to-one comparison of words or characters. In this study, we propose the geometry based similarity measurement and energy based similarity measurement.
measurement to calculate the similarity of character pairs appearing in both the query and reference handwriting images.

1) Energy based similarity measurement (EBSM):
Image registration is the process of transforming different images into the same coordinate system. Many popular approaches [11]-[15] for image registration are based on Demons algorithm, which is quite efficient and non-parametric. The proposed energy based similarity measurement derives from the idea of image registration, and introduces Log-Demons algorithm [12] into text-dependent writer identification for the first time. At first, we give a brief introduction of image registration and Log-Demons algorithm.

Image registration is to find a displacement field \( s \) which can provide a good alignment of the images \( C_Q \) and \( C_R \), and it is usually treated as an optimization problem. A widely used energy function \( E(c, s) \) [16] for optimizing is defined as:

\[
E(c, s) = \frac{1}{\lambda_i} \| C_Q - C_R \circ c \|^2 + \frac{1}{\lambda_x} \| c - s \|^2 + \frac{1}{\lambda_T} \| \nabla s \|^2,
\]

where \( \lambda_i \) accounts for the noise on the image intensity, \( \lambda_x \) stands for the spatial uncertainty, and \( \lambda_T \) controls the amount of regularization. The variable \( c \) is exact spatial transformation of \( s \). This function introduces a hidden variable in the registration process: correspondences. This auxiliary variable \( c \) is added to decouple the complex minimization into simple and efficient alternate optimization. The optimization of \( E(c, s) \) contains two steps. The first step is to optimize \( \frac{1}{\lambda_i} \| C_Q - C_R \circ s \|^2 + \frac{1}{\lambda_x} \| c - s \|^2 \) with respect to \( c \) and with fixed \( s \). The second step is to optimize \( \frac{1}{\lambda_T} \| \nabla s \|^2 \) with respect to \( s \) and with fixed \( c \).

Log-Demons algorithm is also about the optimization procedure on \( s \). The auxiliary variable \( u \) that represents the update displacement field of \( s \) is added, and \( c = s \circ \exp(u) \).

The correspondence energy of \( u \) and \( s \) is defined as:

\[
E^{corr}_s(u) = \| C_Q - C_R \circ s \circ \exp(u) \|^2 + \frac{\lambda^2_x}{\lambda_T} \| u \|^2,
\]

By adapting Gauss-Newton-like approaches, the update field \( u \) at each pixel \( p \) with the displacement field \( s \) is calculated:

\[
u(p) = -\frac{C_Q(p) - C_R \circ s(p)}{\| J(p) \|^2 + \frac{\lambda^2_x}{\lambda_T}(p)} J(p)^T,\]

where \( J = \nabla^T M \circ s \), and \( \nabla \) is the gradient. After that, the obtained \( u \) is used to update \( s \) for iterations. The overview of Log-Demons algorithm is summarized in Alg. 1.

Fig. 2 and Fig. 3 show image registration results of character pair from the identical writer and different writers. The difference of character pairs from the identical writer is not as significant as that of the character pairs from different writers \((Q_1, R_1)\). As a consequence, the displacement field of character pair from the identical writer is smoother than that of character pair from different writers. For our task of text-dependent writer identification, a character pair \((C_Q, C_R)\) which can be treated as the fixed image and the moving image are provided. \( C_Q \) is the character image from the query handwriting and \( C_R \) is the character image from the reference handwriting. We obtain the displacement field \( s \) which aligns \( C_R \) to \( C_Q \) best by Log-Demons algorithm and calculate the score of energy:

\[
E(s) = \| C_Q - C_R \circ s \|^2 + \| \nabla s \|^2.
\]

It is assumed that the character pair of the identical writer has a lower score of energy than the character pair of different writers. In the classification phase, character pairs are ranked in decreasing order of the energy scores. Based on this ranking, we can verify whether a character pair has actually been written by the identical writer.

2) Similarity calculation: All character pairs of the same text content are used to calculate the similarity. Assume that there are \( \text{Num}_{cp} \) character pairs in both the query handwriting image \( Q \) and reference handwriting image \( R \), and are denoted as \((C^Q_i, C^R_i)\). The energy of the displacement field \( s_i \) between \( C^R_i \) and \( C^Q_i \) is denoted as \( E(s_i) \). The similarity between \( Q \) and \( R \) is defined as the average distance and average energy of all \( \text{Num}_{cp} \) character pairs:

\[
\text{Similarity}(Q, R) = \frac{1}{\text{Num}_{cp}} \sum_{i=1}^{\text{Num}_{cp}} E(s_i).
\]
\[ D_E = \frac{1}{\text{Num}_{cp}} \sum_{i=1}^{\text{Num}_{cp}} E(s_i). \] (8)

C. Similarity fusion of text-dependent and text-independent methods

After text-independent similarity \( D_C \) and text-dependent similarity \( D_E \) between the query and reference handwriting images are calculated, they are normalized into interval [0, 1]. Then, we summarize \( D_C \) and \( D_E \) together to measure the final similarity \( S_Q^R \) between \( Q \) and \( R \):

\[ S_Q^R = \delta \ast D_E + (1 - \delta) \ast D_C, \] (9)

where \( 0 < \delta < 1 \) is the weight parameter to balance the contribution of text-independent and text-dependent methods, and it can be determined by cross-validation.

III. EXPERIMENTAL RESULTS

We evaluated the proposed method on the HIT-MW [17] and CASIA-2.1 datasets [18]. The HIT-MW dataset is built for off-line Chinese handwritten text recognition, and contains 853 Chinese handwriting samples. 254 images from 241 writers are labeled with writer information. In our experiments, the handwritings of 240 writers are employed by our experiments. We also use another Chinese handwriting database, viz. the CASIA database [18]. This database contains off-line and on-line Chinese handwritings. For the off-line part, there are three sub-datasets (CASIA-2.0-2.2). We use one of them, i.e., CASIA-2.1, to evaluate our method and report experimental results. The CASIA-2.1 dataset contains two sub-datasets, we use the larger one which contains handwritings of 240 writers. Both datasets are divided into the query and reference set, and every writer has only one image in each set. Given a query handwriting image \( Q \), the system sorts all the images in the reference set based on their similarities compared with \( Q \). Ideally, the reference handwriting image with the minimum distance should be created by the same writer of \( Q \). Ranking list (Top-N) is used to measure the performance of the proposed method. For the Top-N criterion, a correct hit is accumulated when at least one handwriting in the first \( N \) place of the ranking list is created by the correct writer. In our experiments, we use the identification accuracy of Top-1, Top-5, and Top-10 to evaluate the performance of our method. The size of grid for the contour-directional feature extraction is set equal to \( 15 \times 15 \). The ground truth of the HIT-MW and CASIA-2.1 datasets is utilized to find character pairs in both the query and reference handwriting images.

A. Weight parameter \( \delta \)

We carry out the experiment to find the optimal weight parameter \( \delta \) of two datasets. The value of \( \delta \) is selected from 0 to 1 incrementally. For each value, we evaluate the Top-1 accuracy to investigate its effect on the performance. The optimal weight parameter \( \delta \) for the HIT-MW dataset is 0.27, and the optimal weight parameter \( \delta \) for the CASIA-2.1 dataset is 0.21.

B. Comparison of the proposed method with others

We compare the proposed method with previous methods of text-independent Chinese writer identification. Tab. I and Tab. II show the performance of our method and previous ones on the HIT-MW and CASIA-2.1 datasets, respectively. The best Top-1 identification accuracy on the HIT-MW and CASIA-2.1 datasets reaches 97.1% and 98.3% respectively, which outperforms other previous approaches. It demonstrates that text-independent and text-dependent methods characterize the handwriting from different aspects, and the combination of both characteristics enhance the identification performance.

IV. CONCLUSION

In this paper, we propose a novel method for Chinese writer identification taking the advantage of both text-independent and text-dependent characteristics. In order to exploit text-dependent information, we propose a new strategy to measure the similarity of character pairs appearing in both the query and reference handwriting images. We borrow the idea from image registration and introduce the Log-Demons algorithm into writer identification for the first time. We calculate the displacement field which is used to align the character pairs, and utilize the energy of displacement field for similarity measurement. Experimental results show that the proposed method enhances the identification performance of text-independent methods, and outperforms other previous approaches on two public datasets.

REFERENCES


Oral Session B

Computer vision and image processing

Monday May 14, 2018, 10:00 AM

Omar Krichen, Nathalie Girard, Eric Anquetil and Mickaël Renault
Real-time interpretation of geometric shapes for digital learning

Marcus Valtonen Örnhag and Anders Heyden
Generalization of Parameter Recovery in Binocular Vision for a Planar Scene

Michael Wirth
Developing Image Processing Algorithms in Julia

Guangyi Chen, Tien D. Bui and Adam Krzyzak
Illumination Invariant Face Recognition Via Dual-Tree Complex Wavelet Transform in Logarithm Domain

Hongliu Cao, Simon Bernard, Laurent Heutte and Robert Sabourin
Dissimilarity-based representation for radiomics applications
Real-time interpretation of geometric shapes for digital learning

Omar Krichen  
Univ Rennes, CNRS, IRISA  
F-35000 Rennes  
omar.krichen@irisa.fr

Nathalie Girard  
Univ Rennes, CNRS, IRISA  
F-35000 Rennes  
nathalie.girard@irisa.fr

Eric Anquetil  
Univ Rennes, CNRS, IRISA  
F-35000 Rennes  
eric.anquetil@irisa.fr

Mickaël Renault  
Univ Rennes, CNRS, IRISA  
F-35000 Rennes  
mickael.renault@irisa.fr

Abstract—In the context of the ACTIF project that aims for active and collaborative learning promotion, this paper presents a pattern recognition and analysis system for Geometry learning in middle school. The goal is to allow students to draw geometric shapes on a touch-tablet, given a teacher’s instruction. To make the student active, the system has to recognize and analyze on the fly the student’s productions in order to produce real-time visual, corrective, and guidance feedback. We base our work on the visual grammar CD-CMG [1] (Context Driven Constraints Multi-set Grammar), to model the domain prior knowledge and interpret the hand-drawn sketches on the fly. Our first contribution lies in adapting this grammar to the Geometry domain to cover the geometric objects taught in middle school curriculum. Although being expressive enough to model this large scope, the formalism could not cope with the exigence of real-time analysis, given that the multiple interactions between geometric objects generate combinatorial issues. Our second contribution lies in extending the formalism which resulted in having an acceptable performance for a real-time user interaction system. The first experiments show that the proposed approach allows simplicity and interpretation time reduction.

Index Terms—On-line Recognition, Hand-drawn stroke analysis, Digital learning

I. INTRODUCTION

Our work is in the context of ACTIF project, which aims to use pen-based tablets in an educational context, mainly in French middle schools, to foster active learning [14]. In this paper, we focus on learning Geometry by drawing freely on a touch-tablet. Dynamic Geometry Software products are now an important part of teaching geometry. Their goal is to make geometric concepts understanding easier for the student by graphical construction, manipulation and visualization of figures. To our knowledge, the tools used in middle schools, such as Geogebra [2], rely on a drag-and-drop approach to manipulate geometric objects. Indeed, in order to compose a figure, a student must choose from a graphical panel the object he wants to create then has to place its components in the interface. This tends to limit the creative process of the user. In [13], Fiorella and Mayer demonstrate that "generative drawing", i.e learning by drawing, has a positive impact on students learning abilities in the classroom. In [15], Kluger and DeNisi show the impact of feedback intervention on learning performance. These two points represent the pedagogical foundation of our project. We propose a pen-based system that simulates the traditional pen and paper figure sketching and enriches it by real-time visual, corrective and guidance feedback. This paper presents the first works done in the project, and focuses on the online recognition method of the system. In the literature several works have been done for hand-drawn sketches recognition, the following Sec. II presents an overview of existing approaches. Based on this overview and the application domain, we introduce the formalism and the modelling of the geometric knowledge in Sec. III. Since the system has to recognize hand-drawn sketches in real-time, some optimizations are needed, Sec. IV describes the formalism extension and its impact on the analysis process. Sec. V presents our experiments and results while conclusion and perspectives are given in Sec. VI.

II. RELATED WORKS

In this work, we are interested in on-line recognition of handwritten structured documents. We distinguish between two types of handwritten documents interpretations methods: lazy [8] and eager [10]. Lazy interpretation means that the analysis process begins after completion of the user’s production. Eager interpretation means that the handwritten strokes are analyzed on the fly, which is more relevant to our objective of having real-time corrective and guidance feedback to prevent error propagation. There are two major approaches for document analysis: statistical and structural. Statistical approaches [3] rely on learning on large labelled databases to discriminate between symbols and are well suited for isolated shapes recognition. However, they do not allow the modelling of the document’s structure. Since we are in Geometry context, the system has to recognize not only the geometric shapes, but also the structural relations between the objects. Structural approaches consider a symbol in terms of its constituents, the graphical primitives, and the structural relations between them. For instance, a triangle is considered as three segments related by spacial relations. Structural approaches rely on modelling prior domain knowledge by visual grammars. We distinguish between two classes of structural recognition methods. The former is based on graph grammars. For example, Zannibi et al. [4] use labelled graphs to recognize handwritten mathematical expressions. One problem with graph-based methods is that they are complex to manipulate for the designer, especially if the production rules number is high. The latter is based on bi-dimensional grammars. For example, in [5] Hammond...
and Davis proposed Ladder, a generic description language, and applied it for the interpretation of Truss diagrams in a digital learning context [6]. In [7], a bi-dimensional extension to the Stochastic Context Free Grammar is proposed for handwritten mathematical expressions analysis. In this work, to model the geometry domain knowledge, we prefer to use Context Driven Constraints Multi-set Grammar (CD-CMG) [1], a generic formalism for easier interpretation of hand drawn documents. Indeed, in this grammar, the context is explicitly specified in the production rules, which reduces the search space. Moreover, this formalism is the combination between a statistical approach (to locally recognize a shape) and a structural approach (to model the global structure of the document). Finally, CD-CMG has been applied on various types of documents such as architectural plans [11] or electrical sketches [1]. All these features show this formalism is well adapted for our purpose. Thereafter, our contribution lies in two folds: adapting this grammar to the Geometry domain for e-education (see Section III), and extending this formalism to match the constraint of real-time analysis of geometric productions (see Section IV).

III. GEOMETRY DOMAIN MODELLING

In this section, we present the formalism, and illustrate it through its adaptation to the Geometry domain.

A. Context Driven Constraints Multi-Set Grammar

As an extension of the well-known grammar CMG [12], CD-CMG is formally defined as follows:

**Definition 1.** A CD-CMG is a tuple \( G=(V_N, V_T, S, P) \) with:

- \( V_N \): the set of non terminal symbols = symbol classes;
- \( V_T \): the alphabet, here \( V_T = \{ \text{stroke} \} \);
- \( S \): the first symbol, or axiom;
- \( P \): the set of production rules.

And where a production rule \( p \in P \) is composed of three blocks allowing different levels of vision on the document. The **precondition** and the **postcondition** blocks stand for the global vision of the document while the **constraint** block stands for the local vision of the analyzed strokes. Therefore, a production rule \( p \) is denoted as follows:

\[
\alpha \rightarrow \beta \{ \text{Preconditions} \} \{ \text{Constraints} \} \{ \text{Postconditions} \} \mid \alpha \in V_N^+, \beta \in (V_T \cup V_N)^+
\]

Preconditions and postconditions are based on the concept of **Document Structural Context**, which models a zone in the document and the awaited elements in it, defined as follows:

**Definition 2.** A DSC is defined by \( (\lambda)[\text{position}](\gamma)[\text{part}] \) where:

- \( \lambda \) is a set of reference elements;
- **position** is a zone (i.e a position) related to \( \lambda \);
- **\( \gamma \)** is a set of awaited symbols in this zone;
- **\( \text{part} \)** is a part of the awaited symbol that has to intersect the zone.

The **preconditions** are a set of DSC that have to be satisfied and represent the context in which \( \beta \) can be replaced by \( \alpha \). The **postconditions** are a set of DSC that represent the objects that can be created from the new reduced elements \( \alpha \). This formalization enables to drive the analysis process by the context. Indeed, the preconditions represent the verification step while the postconditions represent the prediction step. The **constraints** model a local vision on the analyzed elements \( \beta \). They have two purposes: checking that the shape of \( \beta \) is consistent with the production, and decide if it is pertinent to reduce \( \beta \) into \( \alpha \).

B. Adaptation of CD-CMG to Geometry

We consider the main geometric objects taught in French middle schools: segments, arcs, circles, angles, all the types of triangles and quadrilaterals. We defined around 20 productions rules to model these objects productions as well as the interactions between them (e.g intersection and orthogonality).

Let’s illustrate this with two production rules.

Fig. 1 presents a part of a **segment production rule**, while Fig. 2 and Fig. 3 illustrate a segment composition. In this example, the red stroke in Fig. 2 is transformed into a segment if the stroke' shape is close to a segment' shape (red rectangle in Fig. 1) models the fact that a bisector production rule will be triggered if a straight line intersects the center zone of the new created segment (res).

Fig. 1: Segment production rule in CD-CMG

**Segment**: res → stroke: \( t \) where:

- **Preconditions:**
  
  \( \{\text{Segment}:s1\}[\text{InitialExtremity}]\)
  
  \( \{t\}[\text{first}] \)

- **Constraints:**
  
  Recognizer\((t, \text{segment})\)

- **Postconditions:**
  
  \( \{\text{res}\}[\text{center}]\) \( \{\text{Straight}:d\} \)
  
  \[\text{[one]} \implies \{\text{Bisector} \rightarrow d\}\]

Fig. 2: Drawn stroke (in red) Fig. 3: Interpreted segment

Fig. 4 illustrates the **production rule of a scalene triangle**. The precondition block is composed of one precondition which is a conjunction of three DSC modelling the fact that each segment intersects a zone related to the other two. The
Triangle: \text{res} \rightarrow \text{segment: } s_1, s_2, s_3 \text{ with:}

**Preconditions:**

(S1) [Zone] (s2) [one] & (S2) [Zone] (s3) [one] & (S3) [Zone] (s1) [one]

**Constraints:**

LinkedSegments(s1, s2, s3)

Fig. 4: Triangle production rule

constraint here is structural, such that it verifies that the three segments are linked by their extremities.

Since the structural context is the same for all types of triangles (and quadrilaterals as well), we established a hierarchy between production rules, from general to specific, in order to prune the search space and speed up the analysis process. For example, a triangle can be reduced into an isosceles triangle if two of its sides are equal.

C. Analysis process associated to CD-CMG

The analysis process, extensively explained in [1], is a combination of a bottom-up strategy (guided by the reduced elements) and a top-down strategy (guided by the postconditions DSCs). For each new element, the parser searches the DSC it satisfies and vice versa. Consequently, a production is triggered if its \(\beta\) elements contain at least a new element and its precondition block contains at least a new DSC. Let’s consider the scene illustrated in Fig. 5 composed of a new stroke \(t\) (in red) and 3 segments (s1, s2, s3).

![Fig. 5: Stroke analysis process](image)

The analysis of \(t\) leads to the construction of the analysis (or derivation) tree presented in Fig. 6. The root represents the stroke \(t\). The nodes and the leaves represent the triggered rules, while the blue path is the sequence of reduced production rules, *i.e.* the analysis result. As shown in Fig. 6, \(t\) is first reduced into a segment, denoted thereafter \(s_4\). Then, several rules are tested among which the production rules that led to the correct interpretation of the user’s drawing.

![Fig. 6: Analysis tree](image)

D. Limits of CD-CMG in geometry

As we can see in Fig. 6, the triangle production rule is triggered three times for each possible combination of segments: \((s_4, s_1, s_2), (s_4, s_1, s_3)\) and \((s_4, s_2, s_3)\), even if there is no coherent context for creating a triangle in this scene. This is due to the fact that these productions contain a new element \((s_4)\) and one of their preconditions DSC is satisfied (*c.f.* Fig. 5). Only the preconditions block is checked in this case, since not all DSC are satisfied. The impact on the combinatorics is not important here, but when the document is complex, the analysis becomes costly. Even though the formalism is generic and expressive enough to model the prior geometry knowledge, the multiple possible interactions between geometric objects, *e.g.* creating sub-figures from existing ones (*c.f.* Fig. 7), also generate combinatorics problems in the analysis process.

![Fig. 7: Sub-figures creation](image)

Let’s consider the triangle production rule (*c.f.* Fig. 4). A direct consequence of adapting CD-GMC to the geometry domain is that the \(\beta\) elements (here the 3 segments) are not really replaced by the \(\alpha\) elements (here the triangle). They contribute to create the triangle but they remain considered in the analysis process in order to create other new elements. This has a big impact on the applicable rules search space size. We distinguish two factors producing the combinatorics explosions: the format of the DSCs, and the computation of equivalent interpretations. We will explicit these factors and our proposed solutions in the next section.

IV. REVISION AND FORMALISM EXTENSION

In this section, we present the problems we faced in terms of analysis process complexity and our proposed solutions.

A. The DSC problematic

The expressivity of the formalism in terms of describing the document structure with the DSC allows to formalize that all the components of a polygon are linked by their extremities. Unfortunately, as we have seen in Section III.C, the fact that a production can be triggered even if only one of its preconditions DSC is validated generates a combinatorial problem. Indeed, the more segments a polygon contains, the more DSC there are in the polygon production. Fig. 8 illustrates the composition of a new stroke \(t\) in the context of three already interpreted segments. \(t\) will be recognized as a new segment called thereafter \(s_2\). The fact that \(s_2\) is linked to \([AB]\) (blue zone in Fig. 8) will activate the DSC:

\[
[AB] \text{ [InitialExtremity] } s_2 \text{ [one] } \implies \text{ triangle } \rightarrow [AB], s_2, s_3.
\]
The parser will search the third segment (i.e. s3) that completes the triangle rule with [AB] and s2. There is no contextual information in this DSC about the segment [BC] that completes the triangle since it is not concerned by the zone [AB] [InitialExtremity]. In consequence, for this scene composed of 3 segments besides [AB] and s2, the triangle production rule will be tested three times (for s3=[BC], s3=[ED] and s3=[EF]) instead of once. Thus, the analysis time can be very long, especially if the document is complex. In fact, this issue relies on a CD-CMG limitation. The formalism does not allow to have more than one zone in a DSC, which would enable positioning many awaited elements in relation to one reference element.

To resolve this problem, we propose to refine the constraints on the zones such that a zone can cover all the awaited symbols in the same DSC. The DSC related to the triangle production rules will then be:

\[ [AB] [TotalLengthSegment] s2, s3 [zone], \]

where TotalLengthSegment is the zone that covers the length of [AB] (in blue in Fig. 9). This formulation allows to have a contextual information on all the segments composing a triangle. The loss in focus of the zone (from covering an extremity to covering all the segment) is balanced in the Constraint block by verifying that the segments are structurally linked by their extremities. For the scene illustrated in Fig. 9, the triangle production rule is triggered only once, which reduces the analysis complexity.

**B. Equivalent interpretations problematic and formalism extension**

Since we are in Geometry learning context, it is important to know the dependence links between the elements, e.g. the connections between several segments. These links are modeled in the precondition DSCs. Fig. 10 presents a focus on the precondition block of the segment production rule. It is composed of a dis-junction of three preconditions. They model the fact that there are three possible contexts for a segment creation. The stroke can be linked to two existing segments by their extremities, or linked to the extremity of only one segment, or not linked to anything (in the document).

The **FirstPrecondition** operator, introduced in [11], establishes an order between the preconditions, e.g from specific to general, and forces the parser to stop the context research at the first valid precondition.

**Fig. 11:** FirstPrecondition limits

Fig.11 illustrates a scene in which a stroke $t$ is recognized as a segment. **FirstPrecondition** forces the parser to consider the stroke as linked to two segments, considering only the first precondition. Without this operator, the three preconditions, which are valid in this case, will be tested as hypotheses and the parser will choose the one with the highest score.

A limit to this operator is that the verification of the first precondition can also be complex. Indeed, in this example, there are six contextually valid hypotheses: $t$ can be linked to the couples (s1, s5), (s1, s6), (s1, s7), (s4, s5), (s4, s6), (s4, s7). Hence, six equivalent branches will be created in the analysis tree multiplying the analysis complexity by six. To tackle this problem, we propose to extend the formalism by creating a new operator **FirstContext**. This operator forces the parser not only to stop the research at the first valid precondition, but also at the **first valid context** within a precondition. That means that the search is stopped when the first reference elements that are coherent with the precondition are found. In the example (Fig. 11), the parser will choose the first couple of segments that satisfies the DSCs of the preconditions, e.g it will choose the couple (s1, s7) without considering the other combinations. This will drastically reduce complexity, without losing information about connections. In use, we have noticed a limit to this new operator, which occurs when the segments are not exactly connected, but have overlapping zones (see the example in Fig. ??). Without the **FirstContext** operator, the system computes the membership degree of the stroke’s extremity in each zone of the segments to choose the best possible interpretation. With **FirstContext** operator, it has to
choose the first valid interpretation, which is not necessarily the best. However, the robustness of this extension lies in the interaction with the user since he has the possibility to implicitly validate the interpretation by continuing his drawings, or to delete the segment and redraw it more precisely. This is a trade-off between interpretation precision and analysis process. We will detail the impact of our contributions in the next section.

V. EXPERIMENTS AND RESULTS

A. Quantitative study

To evaluate the impact of our contributions on the system performance, we established several criteria:

- Iterations: number of reduced productions rules
- Interpretations: number of branches in the analysis tree
- Time: Analysis time
- Triggered: Number of triggered rules

The evaluation is realized on one complex drawing benchmark, illustrated in Fig. 13.

We study the impact of our contributions on three critical steps of the drawing scenario of this figure, illustrated in Fig. 14, Fig. 15, and Fig. 16. We compare the performance of our system with DALI, the framework based on CD-CMG and its associated parser. In the following, the term ZoneOpt refers to the constraints refinement on the zones while FirstContext refers to the formalism extension by the addition of the new operator.

1) First step of the scenario: The scene (Fig. 14) illustrates a drawn stroke that will produce an analysis process. The stroke will be first interpreted as segment [AD]. This segments will trigger the production of a rectangle.

2) Second step of the scenario: The scene (Fig. 15) illustrates a more complex production. The drawn stroke will be first interpreted as segment [EB]. This segments will trigger the production of two triangles, one being isosceles and the other rectangle.

3) Third step of the scenario: In this final step of the scenario (illustrated in Fig. 16), the drawn stroke will be first interpreted as segment [DF]. This segment will trigger the production of rectangle triangle, a trapezes and a parallelogram.

TABLE I: First step analysis result

<table>
<thead>
<tr>
<th>Approach</th>
<th>Iterations</th>
<th>Interpretations</th>
<th>Time</th>
<th>Triggered</th>
</tr>
</thead>
<tbody>
<tr>
<td>DALI</td>
<td>4</td>
<td>1</td>
<td>0.23 s</td>
<td>43</td>
</tr>
<tr>
<td>ZoneOpt</td>
<td>4</td>
<td>1</td>
<td>0.20 s</td>
<td>23</td>
</tr>
<tr>
<td>FirstContext</td>
<td>4</td>
<td>1</td>
<td>0.20 s</td>
<td>23</td>
</tr>
<tr>
<td>ZoneOpt + FirstContext</td>
<td>4</td>
<td>1</td>
<td>0.19 s</td>
<td>23</td>
</tr>
</tbody>
</table>

With the classic Dali methodology, the performance is not acceptable since analysis time takes 6.4 seconds. This is due to the number of triggered rules (1057) and equivalent interpretations (3). By modifying the format of the DSCs, ZoneOpt improves the analysis time (2s) by reducing the number of triggered rules. FirstContext operator forces the parser to consider only one interpretation. In consequence, the number of triggered rules decreases from 1057 to 356, and the analysis time is down to 4.2 seconds, which is still not acceptable in a context of real-time interaction with a user. However, the coupling of Opt1 and Opt2 enables to have an analysis time of 0.62 which is acceptable.

TABLE II: Second step analysis results

<table>
<thead>
<tr>
<th>Approach</th>
<th>Iterations</th>
<th>Interpretations</th>
<th>Time</th>
<th>Triggered</th>
</tr>
</thead>
<tbody>
<tr>
<td>DALI</td>
<td>15</td>
<td>3</td>
<td>6.4 s</td>
<td>1057</td>
</tr>
<tr>
<td>ZoneOpt</td>
<td>15</td>
<td>3</td>
<td>2 s</td>
<td>330</td>
</tr>
<tr>
<td>FirstContext</td>
<td>5</td>
<td>1</td>
<td>4.2 s</td>
<td>356</td>
</tr>
<tr>
<td>ZoneOpt + FirstContext</td>
<td>5</td>
<td>1</td>
<td>0.62 s</td>
<td>113</td>
</tr>
</tbody>
</table>
For this step of the scenario, the coupling of ZoneOpt and FirstContext enables to decrease the analysis time from 30 seconds to only 1.5 seconds. As we can see from TABLE II and TABLE III, the more complex the scene gets, the greater the impact of our contributions on the performance gets. Thus, taking into account the desired real-time user interaction, the proposed optimizations allow the design of a system with acceptable performance.

B. Qualitative study

As shown in Fig. 17, we have implemented a pen-based prototype for the composition of geometric sketches. The real-time analysis of the user’s composition allows to give visual feedbacks, which are signs that the system has correctly interpreted the hand-drawn figures (see angle recognition in green in Fig. 17). We propose also edition functions for the user, such as the ability to modify angles, force segments equality and manage parallelism and orthogonality, plus a protractor and compass tools. These features are important in the context of geometric figure composition, such as drawing a parallelogram with certain constraints on segments length and angles. We also implemented a generic method for typical exercises definition, based on Mentoniezh [9], an expert system for proof demonstration in Geometry. We define the procedure of the exercise, i.e. the elements the student has to draw in order to compose the figure given in the instruction. In Fig. 17 we can see the system interface, where the instruction and the corrective feedback are displayed in the left part, whereas the rest of the screen is the sketching space.

**Fig. 17: Prototype interface**

Even though it is still in its early stage, the prototype has been validated by pedagogical experts.

VI. CONCLUSION AND PERSPECTIVES

In this paper, we propose a pen-based system that interprets in real-time geometric figures in order to give visual and corrective feedback, given an instruction. We optimize and extend the CD-CMG formalism to adapt the DALI methodology to geometry. Our contributions have a consequent impact on the system performance which is now acceptable for real-time user-interaction. Our future work consist in improving even more the analysis time, one possible solution being the modification of the analysis process in terms of triggering rules. We will also work on an author mode, where the teacher can create customized exercises. The goal will be to generate automatically the solver procedure from the teacher’s drawings. The system will also have to generate all the alternative procedures in order to give personalized corrective feedback to each student. The successive versions of our prototype will be tested in pilot middle schools, and we will benefit from the studies of LP3C and LOUSTIC laboratories in usage psychology and ergonomics to design visual and corrective feedbacks that are well suited for the students.

ACKNOWLEDGMENT

"ACTIF" project is funded by the French government and the region of Brittany. The authors would like to thank the academic partners and the LP3C and LOUSTIC laboratories.

REFERENCES

Generalization of Parameter Recovery in Binocular Vision for a Planar Scene

Marcus Valtonen Örnhag and Anders Heyden
Centre for Mathematical Sciences
Lund University, Sweden
{marcus.valtonen_ornhag, anders.heyden}@math.lth.se

Abstract—In this paper we consider a mobile platform with two cameras directed towards the floor. In earlier work this specific problem geometry has been considered under the assumption that the cameras have been mounted at the same height. This paper extends the previous work by removing the height constraint, as it is hard to realize in real-life applications.

We develop a method based on an equivalent problem geometry, and show that much of previous work can be re-used with small modification to account for the height difference. A fast solver for the resulting non-convex optimization problem is devised. Furthermore, we propose a second method for estimating the height difference by constraining the mobile platform to pure translations. This is intended to simulate a calibration sequence, which is not uncommon to impose. Experiments are conducted using synthetic data, and the results demonstrate a robust method for determining the relative parameters comparable to previous work.

Keywords—Relative Pose Estimation; SLAM; Visual Odometry; Binocular Vision; Planar Motion; Homography

I. INTRODUCTION

The past decades Simultaneous Localization and Mapping (SLAM) has been studied by the robotics community and the computer vision community alike. The importance of correctly mapping and navigating through unknown terrain is crucial as we enter the age of autonomous vehicles, which is why SLAM is still an active field of research. Furthermore, as the cost for well-performing image sensors is low compared to other alternatives they are an understandable design choice for many applications.

Taking the known properties of the physical environment into account when constructing a SLAM system can increase performance in all stages—from navigation and localization to consistent map building—and many SLAM systems have been developed in different domains, ranging from outdoor environments, to underwater and airborne systems. Modern SLAM systems can, due to the increasing computational power available on consumer products and carefully developed algorithms, handle all of the stages in real-time. Not only monocular systems are available, but a variety of different setups including stereo and RGB-D cameras are supported by the latest frameworks.

In this paper we will continue to investigate methods suitable for planar motion, a problem which is typical for indoor environments. A common approach to determine the correlation between scene points is to estimate the fundamental matrix [1], [2]. This approach fails for planar motion, as it is known to be ill-conditioned, see e.g. [3], and other algorithms have been devised to take the intended environment into account. One way of doing so is by using inter-image homographies, which was proposed by Liang and Pears [4] and Hajjdiab and Laganière [5] among others.

II. RELATED WORK

Recent work on ego-motion recovery has been conducted by Wadenbäck and Heyden [6] for a monocular system using inter-image homographies for a planar scene. The work was later generalized to use more than one homography in each frame in order to make the estimations more accurate and robust [7]. In [8] the authors demonstrate that the parameter recovery can be used as an initial stage in order to transform the problem into a two-dimensional rigid body motion problem, thus being able to track the motion by point correspondences alone, by first correcting for the fixed parameters.

It was shown in Valtonen Örnhag and Heyden [9] how to recover the parameters from a binocular system by extending the methods developed by Wadenbäck and Heyden [7]; however, the method assumed the camera centers to be positioned at the same height above the ground floor, which is hard to achieve in real-life applications.

In [9] the procedure for recovering all parameters, i.e. the tilt angles, the motion of the mobile platform, as well as the relative translation and orientation between the cameras, can summarized by the following: Step 1: Compute the translation and rotation for the mobile platform using images from the first camera. The tilt angles are obtained for both cameras by treating each camera trajectory as being monocular. Step 2: The relative translation is estimated by minimizing the error of a function relating the homographies from the second camera to the motion of the mobile platform. Step 3: The relative rotation is estimated using the homographies and the estimated fixed angles as well as the relative translation.

III. THEORY

A. Problem Geometry

Consider a mobile platform with two cameras directed towards the floor. By a suitable choice of the world coordinate system the cameras move in the plane $z = a$ and
$z = b$ respectively relative to the ground plane positioned at $z = 0$. Both cameras are assumed to be mounted rigidly onto the platform and no common scene point is assumed to be visible in the cameras simultaneously. In this setting the second camera center is connected to the first by a rigid body motion.

It is convenient to work with the camera centers at $z = 0$ and therefore, we consider the equivalent problem where both camera centers are positioned at $z = 0$, but with different ground planes, as is illustrated in Figure 1.

These two models are equivalent; however, in order to re-use results from previous work one may consider the relative translation $\mathbf{\tau} = (\tau_x, \tau_y, \tau_z)$ decomposed into two components $\mathbf{\tau}_{xy} = (\tau_x, \tau_y, 0)$ and $\mathbf{\tau}_z = (0, 0, \tau_z)$. By doing so, $\mathbf{\tau}_{xy}$ from the physical problem geometry is identical to $\mathbf{\tau}$ in the theoretical model with two ground planes. The $z$-component from the original problem, $\mathbf{\tau}_z$, corresponds to the difference between the image planes in the theoretical model. Without loss of generality, one may assume that the origin is located in the first camera center.

### B. The impact of different image planes

In general, a translation by $\mathbf{t} = (t_x, t_y, 0)^T$ in a plane $z = z_0$ is given by $(x, y, z_0) \mapsto (x-t_x, y-t_y, z_0)$, which in homogeneous coordinates correspond to

$$\mathbf{\hat{X}} \mapsto \left[I \mid -\mathbf{t}\right] \mathbf{\hat{X}},$$

where $\mathbf{X} = (x, y, z_0, 1)$, $\mathbf{\hat{X}} = (x, y, z_0)$ is the corresponding image point and $\mathbf{n} = (0, 0, 1)^T$ is a floor normal. We introduce the notation $\mathbf{T}_b(z_0) = I - \mathbf{n}\mathbf{n}^T/z_0$, which is justified by (1). For convenience, let $\mathbf{T}_b(1) = \mathbf{T}_b$, thereby leading to the relation $\mathbf{T}_b(b) = \mathbf{T}_{b/b}$.

The difference, compared to the simplified case where $a = b = 1$, is that the translations occur in different image planes. This physical property can be incorporated in the translation matrices $\mathbf{T}_b$ and $\mathbf{T}_t$ by a scaling factor equal to the distance to the image plane. The monocular case is not affected by introducing different image planes more than up to scale; however, this issue is always present due to global scale ambiguity. The method proposed in [9] for estimating the relative orientation can be re-used with the modified translation matrix without changing the theoretical aspects or the performance. Therefore, the major concern in the generalized problem is to estimate the position of the different image planes relative to each other.

### C. Camera matrices

It was shown in [6] that the camera matrices for the first camera, assuming two consecutive images $A$ and $B$, can be parametrized as

$$\begin{align*}
\mathbf{P}_A &= \mathbf{R}_{\psi_\theta} [I \mid 0], \\
\mathbf{P}_B &= \mathbf{R}_{\psi_\theta} \mathbf{R}_{\phi} [I \mid -\mathbf{t}],
\end{align*}$$

where $\mathbf{R}_{\psi_\theta}$ is a rotation $\theta$ around the $y$-axis followed by a rotation of $\psi$ around the $z$-axis. The translational component of the mobile platform is denoted $\mathbf{t}$ and the rotational component by $\phi$, which is assumed to rotate about the $z$-axis. The corresponding rotation matrix is denoted by $\mathbf{R}_{\phi}$. In [9] the camera matrices for the second camera were derived. By accounting for the different image planes they are given by

$$\begin{align*}
\mathbf{P}_A' &= \mathbf{R}_{\psi'_\theta} \mathbf{R}_{\phi} \mathbf{T}_b(b)[I \mid 0], \\
\mathbf{P}_B' &= \mathbf{R}_{\psi'_\theta} \mathbf{R}_{\phi} \mathbf{T}_b(b)[I \mid -\mathbf{t}],
\end{align*}$$

where $\psi'$ and $\phi'$ are the tilt angles defined as in the first case, $\tau$ is the relative translation between the camera centers and $\eta$ is the fixed rotation about the $z$-axis relative to the first camera.

### D. Homographies

The homographies between two consecutive images are derived as in Valtonen Örhag and Heyden [9], taking the impact of different image planes into account, giving

$$\begin{align*}
\mathbf{H} &\sim \mathbf{R}_{\psi_\theta} \mathbf{R}_{\phi} \mathbf{T}_b(a) \mathbf{R}_{\psi'_\theta}', \\
\mathbf{H}' &\sim \mathbf{R}_{\psi'_\theta'} \mathbf{R}_{\phi} \mathbf{T}_b(b) \mathbf{T}_b^{-1}(b) \mathbf{R}_{\psi_\theta} \mathbf{R}_{\phi} \mathbf{T}_b(b).
\end{align*}$$

The matrices representing the homographies can be made unique by imposing $\det \mathbf{H} = \det \mathbf{H}' = 1$, which will be assumed throughout the paper.

### IV. PARAMETER RECOVERY

By separating the fixed angles from $\phi$ and the translation $\mathbf{t}$ in (4) the following relation holds

$$\begin{align*}
\mathbf{R}_{\phi} \mathbf{T}_b(b) &= \mathbf{T}_b^{-1}(b) \mathbf{R}_{\psi_\theta} \mathbf{H}' \mathbf{R}_{\psi'_\theta} \mathbf{R}_{\phi} \mathbf{T}_b(b),
\end{align*}$$

### A. Pure translation

Impose the constraint $\mathbf{R}_{\phi} = \mathbf{I}$, i.e. the mobile platform does not rotate. Due to global scale ambiguity one may choose $a = 1$. This leaves the translation vector $\mathbf{t} = (t_x, t_y)$, the fixed parameters and the additional scale parameter $b$ to be estimated. Under these assumptions (5) is simplified to,

$$\begin{align*}
\mathbf{T}_b &= \mathbf{R}_{\psi_\theta} \mathbf{R}_{\psi'_\theta} \mathbf{H} \mathbf{R}_{\psi'_\theta} \\
\mathbf{T}_b(b) &= \mathbf{T}_b^{-1}(b) \mathbf{R}_{\psi_\theta} \mathbf{R}_{\psi'_\theta} \mathbf{H}' \mathbf{R}_{\psi'_\theta} \mathbf{R}_{\phi} \mathbf{T}_b(b).
\end{align*}$$

Note that in this model the translation vector $\mathbf{t}$ can be estimated from the first homography. Following the same method used by Valtonen Örhag and Heyden [9], when treating the case $b = 1$, one may separate the relative translation vector from the relative orientation, which in this case cancels out, due to commutativity of translation matrices

$$\begin{align*}
\mathbf{T}_b(b) &= \mathbf{R}_{\phi} \mathbf{T}_b^{-1}(b) \mathbf{H}' \mathbf{R}_{\psi'_\theta} \mathbf{R}_{\phi}.
\end{align*}$$
Since the eigenvalues of $T_{t/b}$ do not depend on $t$ nor $b$ one may multiply both sides with the transpose from the left, which yields
\[ T^T_t(b)T_t(b) = R^T_{t/\psi}R^T_{\psi'/\psi}H'^TH'H'R_{\psi'/\psi}R_{\eta}. \]  
(8)

By interpreting the right-hand side of (8) as a similarity transformation, it follows that the eigenvalues of the right-hand side are those of $H'^TH'$. In [10] it was shown that the eigenvalues for the general form $T^T_sT_s$ are given by
\[ \lambda_2 = 1, \quad \lambda_{1,3} = 1 + \frac{|s|^2}{2} \pm \frac{|s|}{2} \sqrt{|s|^2 + 4}, \]  
(9)
where all eigenvalues are positive and $\lambda_1 \geq 1$, and $\lambda_3 \leq 1$ fulfilling the relation $\lambda_1/\lambda_3 = \lambda_2^2$. Using the same approach as in [9] and the relation $T_t(b) = T_{t/b}$, the sum of the eigenvalues gives an equation for $b$
\[ 3 + \frac{t^2}{b} = \text{tr} H'^TH'. \]
(10)

For convenience, introduce $\xi = 1/b^2$, then $\xi > 0$. Furthermore, when $b = 1$ it follows that $\xi = 1$, which gives the same equation as when the cameras are mounted at the same height. For a single pair of homographies one gets
\[ r_i\xi - h_i = 0, \]  
(11)
where $r_i = |t_i|^2$ and $h_i = \text{tr} H^T_iH_i - 3$. Due to noise one may consider minimizing
\[ \min_{\xi \geq 0} \sum_{i=1}^N |r_i\xi - h_i|^2, \]
(12)
for $N$ pairs of homographies. Vectorizing the objective function, this can be written as
\[ f(\xi) = \|r\xi - h\|^2 = \xi^2 r^T r - 2 \xi r^T h + h^T h, \]  
(13)
giving an optimal value $\xi^* = (h^T r) / (r^T r)$, which in turn gives an estimate for $b$, namely
\[ b = \sqrt{\frac{r^T r}{h^T r}}. \]  
(14)

The practical implications of this result is that one may recover $b$ when no rotations are present, which implies that the mobile platform can be calibrated along a straight path before allowing more general motion. In practice, it is not uncommon to impose a calibration sequence on a mobile platform; however, in the following section we will devise a method that recovers the parameters for general planar motion.

B. General planar motion

Consider (5) without the constraint $R_x = I$. Again, $t$ and also $\varphi$, can be recovered from the monocular case and the second equation can be re-written as
\[ T_{\tau/b}T^T_{\tau/b}T^{-1}_{\tau/b} = R^T_{\tau/\psi'}R^T_{\psi'/\psi}H'R_{\psi'/\psi}R_{\eta}, \]  
(15)
where we use the relation $T_t(b) = T_{t/b}$ and similarly for $\tau$. Multiplying with the transpose from the left yields
\[ T^T_{t/b-\tau/b}R^T_{\tau/\psi'}T^T_{\tau/b}T_{t/b-\tau/b} = R^T_{\psi'}R^T_{\tau/\psi'}H'^TH'R_{\psi'/\psi}R_{\eta}. \]  
(16)

It is shown in Valtonen Örnhag and Heyden [9] that the left-hand side LHS of (16) can be simplified to
\[ \text{LHS} = \begin{bmatrix} 1 & 0 & \ell_1 \\ 0 & 1 & \ell_2 \\ \ell_1 & \ell_2 & \ell_3 \end{bmatrix}, \]  
(17)
which still holds true, with a modification of the values $\ell_1$, $\ell_2$ and $\ell_3$. It is also shown that the eigenvalues of LHS are given by $\lambda_2 = 1$ and $\lambda_1, \lambda_3$ such that $\lambda_1\lambda_3 = \ell_3 - \ell_1^2 - \ell_2^2 = 1$. Introducing $\xi = 1/b^2$ for convenience, it follows that
\[ \ell_3 = k_1\xi\tau_x + k_2\xi\tau_y + c|\tau|^2 + |t|^2 + 1, \]  
(18)
where
\[ k_1 = 2(t_x \cos \varphi - t_y \sin \varphi - t_z), \]
\[ k_2 = 2(t_x \sin \varphi + t_y \cos \varphi - t_y), \]
\[ c = 2(1 - \cos \varphi). \]  
(19)
Furthermore, the right hand side of (16) has the same eigenvalues as $H'^T H'$, as they are similar. Since the sum of the eigenvalues is the trace of the corresponding matrix, the following relation holds

$$\text{tr } H'^T H' = 2 + \ell_3,$$  

(20)

which is independent of $\eta$. By letting $h = \text{tr } H'^T H - 3$ the relation becomes

$$k_1 \xi_{r x} + k_2 \xi_{r y} + c \xi |\tau|^2 + \xi |t|^2 - h = 0.$$  

(21)

C. Solving for the relative translation $\tau$ and unknown scale $b$

Using multiple pairs of homographies the problem can be formulated as

$$k_1^{(1)} \xi_{r x} + k_2^{(1)} \xi_{r y} + c^{(1)} \xi |\tau|^2 + \xi |t^{(1)}|^2 - h^{(1)} = 0,$$

$$k_1^{(2)} \xi_{r x} + k_2^{(2)} \xi_{r y} + c^{(2)} \xi |\tau|^2 + \xi |t^{(2)}|^2 - h^{(2)} = 0,$$

$$\vdots$$

$$k_1^{(N)} \xi_{r x} + k_2^{(N)} \xi_{r y} + c^{(N)} \xi |\tau|^2 + \xi |t^{(N)}|^2 - h^{(N)} = 0.$$  

(22)

The system (22) is over-determined for $N > 3$, hence minimizing

$$\min_{\tau \in \mathbb{R}^2} \sum_{i=1}^{N} \left| k_1^{(i)} \xi_{r x} + k_2^{(i)} \xi_{r y} + c^{(i)} \xi |\tau|^2 + \xi |t^{(i)}|^2 - h^{(i)} \right|^2,$$  

(23)

reduces the impact of noise. The minimization problem (23) can be re-formulated as

$$\min_{\tau \in \mathbb{R}^2} \min_{c \in \mathbb{R}^N} \| K \xi \tau + c \xi |\tau|^2 + \xi r - h \|^2_2,$$  

(24)

where $K \in \mathbb{R}^{N \times 2}$, $c \in \mathbb{R}^N$, $r \in \mathbb{R}^N$ and $h \in \mathbb{R}^N$. The vector $r$ consists of the elements $r_i = |t_i|^2$, and the other follow the naming convention used in Valtonen Órnhag and Heyden [9]. Introducing $x = (\xi_{r x}, \xi_{r y}, \xi |\tau|^2, \xi |r|^2)^T$ the problem can be reformulated as

$$\min_{x \in \mathbb{R}^4} \| M x - h \|^2_2,$$  

(25)

where $M = [K \mid c \mid r]$. The constraint can be written $x^T A x = 0$ where

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1/2 \\ 0 & 0 & -1/2 & 0 \end{bmatrix}.$$  

(26)

The Lagrangian is given by

$$\mathcal{L}(x; \lambda) = x^T Q x - 2d^T x + \lambda x^T A x,$$  

(27)

where $Q = M^T M$ and $d = M^T h$. A necessary condition for a minimizer is given by $\nabla_x \mathcal{L}(x; \lambda) = 0$, which yields

$$x = (Q + \lambda A)^{-1} d.$$  

(28)

We will denote $Q_\lambda = Q + \lambda A$ in the remaining part of this section. Inserting (28) in the equation obtained by $\nabla_\lambda \mathcal{L}(x; \lambda) = 0$ yields

$$d^T Q_\lambda^{-1} A Q_\lambda^{-1} d = 0,$$  

(29)

which is a rational equation in $\lambda$. Assuming $\det Q_\lambda \neq 0$ this can be turned into a sixth degree polynomial equation

$$d^T \text{adj } Q_\lambda A \text{adj } Q_\lambda d = 0,$$  

(30)

where $\text{adj } A$ denotes the adjoint matrix of $A$. The degree of the polynomial equation is due to the adjoint matrix containing cubic terms. This can in turn be transformed into an eigenvalue problem, and solved robustly as the error is negligible in the case of a general $6 \times 6$ matrix.

V. EXPERIMENTS

The experimental setup follows the one presented in [9], by using synthetic data. Firstly, images simulating those taken by a mobile platform was extracted from a high-resolution image, by transforming image segments into $400 \times 400$ pixels. The original image is of a highly textured floor, chosen to yield many keypoints. Secondly, homographies were estimates from these sequences of images estimated by extracting SIFT keypoints and matching them between subsequent images. The best homography was estimated using RANSAC, where, in order to be considered an inlier, the reprojection error for a point pair was set to five pixels.

The field of view of the simulated cameras are normalized to 90 degrees, and the non-fixed parameters for the mobile platform were estimated from the first camera using the method proposed in [7].

The simulated path of the mobile platform starts by translating along the $y$-direction. This part of the trajectory is intended as a calibration sequence for estimating the parameter $b$, as explained in Section IV-A, and consists of ten images—the remaining 30 images are generated from general planar motion, as shown in Figure 2.

A. Initial calibration vs. General motion

This test case highlights the differences between the two proposed methods, i.e. as an initial calibration sequence consisting of pure translation or allowing general motion. By only considering the first ten images, containing pure translation, the distance to the ground floor was estimates. Using the same setup, but considering only the last 30 images including general planar motion the distance to the ground floor was estimated. In both cases five pairs of homographies were used in each frame. The result is shown in Figure 3.

If one restricts the movements of the mobile platform by imposing an initial calibration sequence the estimated height generally gives a better result than allowing general motion. The methods are both suitable for real-time application, and
Synthetically generated images cropped from a high-resolution image. The sequence consists of 10 images with pure translation initially, which may be used to calibrate the height.

Homographies on the form (4) were generated with $a = 1$, and $b$ randomly chosen, as well as the other parameters. Normal distributed noise was added to the parameters $t$, $\varphi$ and $h$ after generating the homographies. The angle was distorted by $0.01\varepsilon$ (in radians) and the others $\varepsilon$, where $\varepsilon \in N(0, \sigma)$. The error was measured in the weighted norm

$$e = \sqrt{\frac{|\tau^* - \tau|^2}{|\tau|^2} + \frac{(b^* - b)^2}{b^2}},$$

(31)

where the starred variables denote the estimated values, and the values were averaged over 1000 randomized problems for different numbers of homographies and standard deviation. The results are shown in Figure 4.

The average speed, computed on a normal laptop, was 700 $\mu$s for the general case and even faster using the initial calibration sequence.

### B. Error vs. noise

Homographies on the form (4) were generated with $a = 1$, and $b$ randomly chosen, as well as the other parameters. Normal distributed noise was added to the parameters $t$, $\varphi$ and $h$ after generating the homographies. The angle was distorted by $0.01\varepsilon$ (in radians) and the others $\varepsilon$, where $\varepsilon \in N(0, \sigma)$. The error was measured in the weighted norm

$$e = \sqrt{\frac{|\tau^* - \tau|^2}{|\tau|^2} + \frac{(b^* - b)^2}{b^2}},$$

(31)

where the starred variables denote the estimated values, and the values were averaged over 1000 randomized problems for different numbers of homographies and standard deviation. The results are shown in Figure 4.

The average speed, computed on a normal laptop, was 700 $\mu$s for the general case and even faster using the initial calibration sequence.

### B. Error vs. noise

Homographies on the form (4) were generated with $a = 1$, and $b$ randomly chosen, as well as the other parameters. Normal distributed noise was added to the parameters $t$, $\varphi$ and $h$ after generating the homographies. The angle was distorted by $0.01\varepsilon$ (in radians) and the others $\varepsilon$, where $\varepsilon \in N(0, \sigma)$. The error was measured in the weighted norm

$$e = \sqrt{\frac{|\tau^* - \tau|^2}{|\tau|^2} + \frac{(b^* - b)^2}{b^2}},$$

(31)

where the starred variables denote the estimated values, and the values were averaged over 1000 randomized problems for different numbers of homographies and standard deviation. The results are shown in Figure 4.

### VI. Conclusion

This paper has extended the work of Valtonen Örnhag and Heyden [9] to account for two cameras positioned at different heights relative to the ground floor.

A method comparable in speed and robustness to the simplified case when the cameras are located at the same height has been proposed. The method can easily be incorporated in a complete system to determine all fixed and non-fixed parameters of the mobile platform, and makes use of several pairs of homographies at each time step. Furthermore, a method for computing the distance to the ground floor for pure translations has been devised, which can be used as an initialization sequence of the mobile platform.

Experimental results using synthetic data show that both methods give a good estimate for the height of the cameras; however, using an initial calibration sequence yields a better result at the cost of imposing pure translation for the first frames.

### ACKNOWLEDGMENT

This work has been funded by the Swedish Research Council through grant no. 2015-05639 ‘Visual SLAM based on Planar Homographies’.

Figure 2. Synthetically generated images cropped from a high-resolution image. The sequence consists of 10 images with pure translation initially, which may be used to calibrate the height.

Figure 3. Estimated distance to the ground floor for pure translations (top) and general motion (bottom). The blue circles represent the ground truth and the red dots are the estimated parameters. Five pairs of homographies have been used in each frame.

Figure 4. Error vs. noise for different amounts of pairs of homographies $N$. The error was estimated from the mean of 1000 randomly generated problems for.
REFERENCES


Developing Image Processing Algorithms in Julia

Michael Wirth
School of Computer Science
University of Guelph
Guelph, Canada
mwirth@uoguelph.ca

Abstract—The dawn of the 21st century has brought with it a plethora of data, much of which is in the visual spectrum. Associated with this data is the need to develop analysis algorithms to yield relevant information. There are a myriad of different programming languages with which to develop image processing algorithms. This paper explores Julia as a potential language, and introduces a new image processing library.

Index Terms—programming language, image processing, programming environment, Julia

I. INTRODUCTION

Image processing likely had its birth at the Jet Propulsion Laboratory in 1964, processing pictures of the moon taken by Ranger 7 to remove image distortion [5]. This was followed by the 1969 missions of Mariner 6 and 7, processing images of Mars. The surface of Mars is low-contrast, and image processing was used to extract features [1]. The types of image processing included the correction of camera-induced image distortion [5], removal of residual images from previous snapshots affecting current images, and suppression of transmission noise from Mars to Earth [3]. In the early years of what was aptly termed picture processing, Fortran was likely the largest influence due to the numerical nature of the task. Images were also quite small - the Mariner images were 200x200, so processing them within the confines of image enhancement was not exactly arduous.

From the early days of using Fortran, the art of implementing image processing evolved through a series of generic programming languages such as C, C++, and Java, proprietary languages such as MATLAB, and imaging environments such as ImageJ. More recently emphasis has been placed on dynamic languages such as Python due to their rapid prototyping capabilities. The processing of images occurs in such diverse fields as biology, geography, archeology, and agriculture, and as such it can be expected that people involved in analyzing images may be interested in implementing very simple algorithms, for example noise removal, or object counting. These individuals may not necessarily have the programming experience to use languages such as C, or C++, and may want to tailor the processing they do, i.e. not rely solely on the use of standardized algorithms. If a programming language is easy to learn, with a reduced cognitive load, then it can be easily adopted to design and implement algorithms or modify existing code. Programming in C is efficient, but is often lacking from a usability perspective. Python on the other hand is more amenable to the dynamic nature of image processing, but suffers from efficiency constraints when exposed to processing larger images. The programming language Julia provides a segue between these two extremes, offering the speed of C, with the usability/rapid prototyping ability of Python. This paper explores the benefits of using Julia in designing image processing algorithms, and offers a simple image processing library natively written in Julia.

II. LANGUAGE REQUIREMENTS FOR IMAGE PROCESSING

When choosing a programming language for image processing algorithm, it is important to consider the requirements from an image processing perspective. This is particularly relevant for novice programmers, part of the difficulty in implementing algorithms involves the idiosyncracies of a particular language.

1) Array Structures: Image processing is based on the notion of manipulating pixels contained in matrices, and therefore a language used to implement image processing algorithms must be capable of efficiently processing two- and multi-dimensional arrays. A language should be adept at performing element-wise operations on arrays, i.e. not requiring the using of nested loops. An example is processing every element of an image with a single operation, and also have the ability to access blocks of data from an array by means of array slicing.

2) Learnability: A language should be easy to learn for the novice programmer. The basic features should be predictable and easily understood. This is encapsulated in the notion of simplicity. For example a language used for image processing should not require an in-depth knowledge of memory management, nor should the data structures used to hold the images be mind-numbingly complex. With some languages there is a tendency to become fixated with certain languages structures, e.g. datatypes, which ideally should be as transparent as possible. Also related to learnability is the notion of readability, because the easier a language is to use, the more likely it will be adopted by novice programmers from varied disciplines. As mentioned, many people wishing to process images want to perform some simple task, such as image enhancement. A readable language syntax reduces overall program complexity, and affects factors such as program modifiability and maintain-
ability. One example is the use of explicit delimiters such as \texttt{if-endif}, instead of optional block delimiters.

3) \textbf{Efficiency}: How efficiently a language can process images largely depends on how successful it will be. An algorithms efficiency influences its resource footprint, e.g. memory, CPU use, which effects constrained programming environments such as mobile devices and drones (e.g. battery life). The problem with many languages is that coding speed is often sacrificed for program efficiency. A language should not require "tricks", for example vectorization of operations, in order to make a program run more efficiently. The addition of parallelization features is also important, as many image processing algorithms could likely make use of this to improve efficiency.

4) \textbf{Portability and Extensibility}: The ability of code to be portable, reduces the barrier from a prototyping stage, to the final stage of embedding it on a device for real-time applications. This is partially associated with the programming environment. A simple environment with a lack of dependencies makes it easier to port code, as opposed to one which relies on the use of specific third-party libraries. There is also the need for a language to have access to existing libraries of image processing and numerical functions. For example image processing often involves some form of statistical analysis, so providing these simple functions, reduces the programming load. In C applying a median filtering involves implementing a function to calculate the median of a block of data (extract an image sub-block, linearize the data, sort, and derive the median value). Existing code written in other languages should be easily accessible to the language, without needing to re-write the functionality natively.

5) \textbf{Rapid Prototyping}: A simple, readable language also allows for rapid prototyping, i.e. iterative and incremental development. Languages such as Python have become popular because of their ability to rapidly prototype algorithms. This is partially due to their dynamic nature, and also because they have a rich set of built-in data structures, and functions. If an image processing task involves filtering using a Fast-Fourier Transform (FFT), then it is easier for the programming if a suite of such functions exist.

III. \textbf{EXISTING LANGUAGES}

Early image processing algorithms were simple, and pictures were grayscale and relatively small, so programming languages could easily handle the data. Over the years, languages used for image processing progressed through the C-family of languages, proprietary languages such as MATLAB, and rapid prototyping languages such as Python, C and Fortran still hold the edge for computationally intensive calculations, because many dynamically typed languages lack performance. Conversely, high performance languages often lack the ability for rapid prototyping, reducing their usefulness in terms of productivity. The discussion which ensues focuses on pure programming languages rather than visual processing environments such as Gimp, or ImageJ.

1) C, C++: The C programming language likely has the best background for large-scale image processing in an efficient manner (from the perspective of speed). C is quite capable of dealing with large images, however it requires the use of dynamic memory via pointers to create structures capable of storing them. The use of dynamic memory can reduce the languages usability by impeding on rapid prototyping. C also suffers from a lack of both element-wise array operators, and an ability to perform array slicing, and as such relies heavily on the use of nested loops. So processing an entire image by extracting a $5 \times 5$ neighbourhood in C requires a piece of code of the form shown below (where $z=2$ represents the boundary pixels to deal with the $5 \times 5$ neighbourhood):

\begin{verbatim}
for (i=z; i<dx-z; i=i+1)
  for (j=z; j<dy-z; j=j+1)
    for (x=-2; x<=2; x=x+1)
      for (y=-2; y<=2; y=y+1)
        block[x+2][y+2] = image[i+x][j+y]
\end{verbatim}

The cognitive load of C can be quite high, especially as it relates to arrays. To understand arrays requires an understanding of how memory is accessed C, pointers, and ultimately heap memory.

There is also limited built-in support for mathematical functions, and few dedicated image processing libraries. C++ has many of the same benefits/limitations of C, however there are a number of libraries, e.g. OpenCV, which allow for access to primitive image processing functions.

2) Python: Python is a hybrid of a data mining language, and a prototyping language. It is fairly intuitive to learn and has an extensive ecosystem of libraries to facilitate image-related processing, including OpenCV. One of Python’s strengths is fast prototyping, however it suffers from being a slow dynamic language, and the requirement of having to re-write the code in a faster static language such as C or C++ when a “final” product is needed - something commonly known as the two-language problem. It is also less than usable from the perspective of dealing with arrays, which are not native to Python, but rather require a library called numpy. The use of loops in Python is often inefficient, and requires the use of vectorization. Python, which uses parentheses to access array elements (easily confused with function calls), and 0-based indexing, which in itself is not an issue, but coupled with the use of “list”-based iterators in loops can lead to usability issues.

3) MATLAB: A proprietary environment for implementing mathematically oriented algorithms, MATLAB is ideally suited to dealing with the image processing, because it is focused on matrix and linear algebra operations. It offers high readability, and has many built-in functions, but because it is not open-source, requires the purchase of additional toolboxes to deal with various aspects of processing images. However the closed source character of the toolboxes prevents programmers from learning from the code or modifying it for specific purposes. In addition, Matlab is an interpreted language, which negatively affects its performance. Fast code relies on the
The programming language Julia was developed by Stefan Karpinski, Jeff Bezanson, Viral Shah, and Alan Edelman at MIT [6]. It was first released in 2012, and has been in an evolutionary tract since then. The objective was to create a language by encompassing the positive attributes of numerous languages: Python (simplicity and dynamism), R (statistical processing), C (execution speed), Perl (string processing), and Matlab (linear algebra). Julia has many good qualities, foremost of which is that it is an extremely well designed language, with usability one of the guiding principles. It is primarily suppose to act as a scientific language, as a potential replacement for the likes of Python, Fortran, and Matlab.

Julia provides extensive functionality, mostly in the form of data structures, such as arrays and stacks, and a comprehensive library of mathematical functions. In that respect it is less of a traditional language with a low form factor, such as C. As Julia is heavy used in statistical processing of data, it is easy to extend this functionality to processing image data.

A. Language Features

From the perspective of processing images, Julia is a very capable language. One of the major differences between Julia and Python/C is the use of control structure terminators in the guise of the end statement. The for loop is much simpler that its counterpart in C, and Python (where a loop range of 1→n actually implies 1→n-1). It like many of the structures work on the notion of WYSIYG - what-you-see-is-what-you-get. To illustrate its various features with respect to image processing, consider this snippet of Julia code which filters a grayscale image (stored in img) using a 3×3 kernel (i.e. mask).

```julia
1 kernel = [-1 0 1; -4 0 4; -1 0 1]
2 dx, dy = size(img)
3 newImg = zeros(Int, dx, dy)
4
5 for i=2:dx-1, j=2:dy-1
6    # Extract a 3x3 block from the image
7    imgBlk = img[i-1:i+1, j-1:j+1]
8    # Apply the kernel
9    newImg[i, j] = sum(imgBlk .* kernel)
10   end
```

There are effectively seven lines-of-code. The first line defines the kernel to be used as a 3×3 array. Next the dimensions of img are extracted using the built-in function size. On line 3, a new zero-filled image is created to hold the processed array. The loop (lines 5-10) processes each pixel in the image, and array slicing is used to extract the block to be averaged at each pixel. Element-wise multiplication is then used to multiply the elements in the extracted block with the corresponding elements in the kernel.

1) Loops: One of its core benefits is the ability to slice arrays. As seen previously, C’s reliance on loops is inefficient from a rapid prototyping perspective. Here is the same piece of code written in Julia:

```julia
for i=z+1:dx-z, j=z+1:dy-z
    block = img[i-2:i+2, j-2:j+2]
end
```

In Julia, vectorization is not needed for performance increases - Julia generates optimized code, allowing programs to be written in a style that is readable, without sacrificing performance. If a loop is the most understandable way for a novice programmer to solve a problem, there is no need to vectorize the code. One of the more interesting aspects of for loops is that multiple nested for loops can be combined into a single outer loop, which makes for a simpler syntax.

2) Arrays: Arrays incorporate 1-based indexing, and use brackets, [ ]. Some programmers will object to the lack of 0-based indexing, however this is more of an artifact of C than anything else. Both Fortran and Matlab use 1-based indexing, as do most matrices and vectors. It is also easy to define constant arrays, such as kernels, an example of which is shown in Line 1 of the sample code. Many image processing algorithms involve a lot of processing of blocks of information from arrays. This has always been challenging in traditional languages such as C because of the lack of slicing infrastructure. Like Fortran, and Matlab, Julia performs this in a trivial manner. An example of extracting an image sub-block is shown on Line 7 of the code example. Julia allows for element-wise operations on arrays, which reduces the need for additional loop structures, making code more compact (e.g. Line 9, convolution of the image block with the kernel).

3) Types: One of the distinct benefits of Julia as it concerns image processing is data types. Being dynamically typed, variables do not need to have a datatype associated with them, but the type can be indicated. In many traditional languages such as C, the use of an int to store 8-bit image data is overkill. To process an 8-bit image with 256 differing levels of gray does not require a 32-bit container. So having access to a more appropriate datatype such as UInt8 results in both a reduction in the memory used to store an image, and a reduction in the processing time.

4) Image I/O: One of the inconveniences of any programming language used for image processing is reading in specific file formats. In fact trying to read TIFF or JPEG images is often a bugbear in any language. One alternative is to store images as text files, containing the raw 8-bit values (for grayscale images). Julia provides an easy method of reading (and writing) images using the functions readdlm() and writedlm(). This function readdlm() reads the image in based on the the rows and columns of the image inside the
file given by `fname`. The argument `UInt8::Type` specifies the format type to be stored in the array `imgIn`. The function `writelm()` is analogous for writing images.

**B. Interoperability**

From an interoperability perspective, Julia is capable of easily integrating C, Python, and Matlab code, which makes it easy to access existing libraries without the need to port the code. Julia is also capable of parallel operations, which makes perfect sense in the case of numerous image processing applications. Calling functions from C or Fortran shared libraries can be done with no extra boilerplate glue code. For example the following line of code calls a C function called `vectorMean` which takes as input a vector of values, and its length, and returns the mean of those values.

```plaintext
aM = ccall((:vectorMean, "myC.so"),
            Float64, (Ptr{Cint},Cint), x, 5)
```

**C. Efficiency**

In traditional static-typed languages, such as C, compilers need type information in order to generate efficient code. Julia is a dynamically typed language, but approaches the performance of a static-typed language. This is achieved by generating machine code using a just-in-time LLVM-based compiler. To illustrate the efficiency of Julia, we applied a 5×5 mean filter to a grayscale image. A mean filter is simply a way of smoothing an image, i.e. suppressing irregularities in the intensities, and is common used for noise suppression. As such it is an array-processing intensive algorithm.

In this particular example, a mean filter is applied to a 2144×6640 pixel, 8-bit image: effectively 14 megapixels. The algorithm is analyzed using the Julia code shown below, and analogous code written in C, Fortran, and Python.

```plaintext
function meanFilter(img)
    dx,dy = size(img)
    e = div(5,2)
    for i=e+1:dx-e, j=e+1:dy-e
        block = img[i-e:i+e,j-e:j+e]
        mn = mean(block)
        imgN[i,j] = round(UInt8,mn)
    end
    return imgN
end

img = readdlm("pano.txt",Int16::Type)
@time imgN = meanFilter(img)
writedlm("meanPano.txt",imgN)
```

The code has been timed based just on the process of performing the mean filtering, and avoiding extras such as file I/O. In most cases this involves a nested loop to process each pixel in the image. In the case of C, it includes a second set of nested loops to extract the 5×5 neighbourhood. The C code was able to process the image in less than a second (0.878s), with Fortran in second place at 1.344s. At 4.017 seconds, the Julia code could be considered to be sufficiently fast as well. The underachiever in this instance is Python which is invariably slow at 245.2s, even using only one pair of nested loops.

**D. Learnability**

The simplicity of Julia stems partially from the simpler nature of the structure used to store the image (a simple array, with transparent storage). The C function equivalent to `meanFilter()` uses four for loops (in addition to four for image I/O), the Julia function uses one (technically a nested loop). Copying the image is simpler, and does not require a structure which incorporate the dimensions of the array (again, transparent). Extraction of the block is achieved through array slicing, and convolution of the block with the mask is by means of element-wise multiplication. Finally, calculating the average uses the built-in function `mean()`. The size of the array in C requires the use of either a static array (used in this case), or a dynamic array. Both require the knowledge that large arrays cannot be stored on the stack memory in C. The code is simple because Julia provides efficient means of performing I/O on an entire block of integers in an ASCII file. Python requires the use of an ancillary library, and both C, and Fortran require explicit code to read in data from a file.

To put the C code into context, it is given below.

```c
#include <stdio.h>
#include <time.h>
#define dx 2144
#define dy 6640

int main(void)
{
    FILE *ifp, *ofp;
    int i, j, x, y, w=2, sum;
    static int image[dx][dy];
    static int imageN[dx][dy];

    ifp = fopen("pano.txt","r");
    for (i=0; i<dx; i=i+1)
        for (j=0; j<dy; j=j+1)
            fscanf(ifp,"%d", &image[i][j]);
    fclose(ifp);

    for (i=w; i<dx-w; i=i+1)
        for (j=w; j<dy-w; j=j+1)
            imageN[i][j] = image[i][j];

    for (i=0; i<dx; i=i+1)
        for (j=0; j<dy; j=j+1)
            imageN[i][j] = image[i][j];

    for (i=w; i<dx-w; i=i+1)
        for (j=w; j<dy-w; j=j+1)
            sum = 0;
            for (x=-w; x<=w; x=x+1)
                for (y=-w; y<=w; y=y+1)
                    sum = sum + image[i+x][j+y];
```

```c
    return 0;
}
```

Copyright © 2018 by CENPARMI, Concordia University
imageN[i][j] = (int) sum/25.0;
}

ofp = fopen("meanPano.txt","w");
for (i=0; i<dx; i=i+1){
    for (j=0; j<dy; j=j+1)
        fprintf(ofp,"%d ", imageN[i][j]);
    fprintf(ofp,"\n");
}
fclose(ofp);
return 0;
}

E. Caveats

Julia, like any language has some features that may be somewhat controversial. Most people experienced in languages such as C deal with 2D arrays in a row-major, which means arrays are stored as rows. Julia however harks back to Fortran (and MATLAB) and uses column-major arrays. This is a little tricky, but it’s just about memory layout, and shouldn’t interfere too much with processing an array. It is actually a benefit when incorporating functions from languages such as Fortran, and R which are similarly column-major. The only caveat occurs when “flattening” the array, as it is flattened by column, instead of by row. The biggest hindrance to the use of Julia at the moment is the fact that Julia is still an evolving language, and modifications to the structure of the language will facilitate changes to programs written in Julia, as elements are modified, and deprecated. Once V1.0 is released, then hopefully changes will be reduced moving forward.

V. JULIA IMAGE PROCESSING TOOLBOX

We are developing a toolbox for image processing written in Julia. The toolbox is free of dependencies, meaning it does not require any other packages to work (except for visualization). The rationale for developing the toolbox is to provide a suite of functions to perform simple image processing tasks in native Julia. The increased popularity of Julia as a scientific programming language, together with increased manipulation of visual information makes this an ideal environment in which to design an image processing toolbox. The toolbox is currently written in Julia V0.5.2, and will be released when Julia becomes stable. We have used the toolbox in an upper year course on image processing to allow students to focus on applying algorithms.

A. Aims

The aims of the designing the toolbox are threefold.

1) Accessibility: With an increased use of image-based data in many fields, it is important to provide easy access to implementations of the more common image processing and analysis algorithms. Individuals from diverse fields may wish to implement algorithms to perform simple processing, for example basic suppression of noise in photographs. For an algorithm to be used in an effective manner in research, i.e.

in the context of reproducible results, the source code used to process images should be open-source. This allows the users to easily augment algorithms.

2) Ease of use: Many existing image processing toolboxes can be challenging for novice users to use. This may be due to a lack the programming skills in the language being coded, or overly complex algorithms. Providing simple algorithms, allows novice programmers to learn basic algorithms by exploring the code in a very hands-on fashion - modifying code, and making adjustments to parameters. It is easy to take the meanFilter(), and modify it so that it uses something other than a 5×5 neighbourhood, or another statistical function.

3) Pedagogy: Courses on image processing require access to toolboxes for students to build solutions to problems. They also require access to the actual code to understand how image processing algorithms are designed, and implemented. Providing a toolbox in which the source code is organic, and easy to understand will help them solidify image processing concepts. They can also build on existing algorithms, use the toolbox functions for rapid prototyping, or use the algorithm suites, such as image thresholding to perform experimental evaluations.

B. Toolbox overview

The toolbox focuses on simple functionality in the realm of image processing, and basic image analysis with respect to segmentation, and image morphology. It does not endeavour to replicate more complex functionality such as feature detection via SURF or ORB. Instead it provides a broader repertoire of functionality in areas such as thresholding. Most of the existing algorithms work on grayscale (G), binary (B), or colour (C) images. There are a number of toolboxes:

- Image sharpening (G): Unsharp masking filters
- Noise suppression (G): A series of varied filters from mean filtering to non-linear filters, e.g. Nagao-Matsuyama
- Histogram (G): Functions to generate and manipulate histograms, eg. contrast enhancement
- Colour space (C): Conversion from RGB to YIQ, HSV, YCbCr, and CIELab (and back)
- Noise generation (G,C): Functions to generate noise in grayscale images.
- Image Binarization (G): Local and global thresholding algorithms for grayscale images, e.g. Otsu, Sauvola
- Image I/O (G,C): Text image files, and PGM images
- Spatial transformation (G,C): Various geometrical transformation algorithms, e.g. rotation, perspective correction
- Edge processing (G): Edge enhancement and detection algorithms, e.g. Sobel, Canny
- Segmentation (G): General segmentation algorithms, e.g. histogram back projection
- Image Morphology (B,G): A basic repository of functions for morphological analysis
- Art (C): Artistic filters, e.g. vignette, sepia-tone, social media filters

This toolbox also includes application specific algorithms in the form of toolboxes for skin segmentation, and fire/smoke
segmentation. The purpose of the customized algorithm libraries is to provide the programmer with algorithms derived from the literature which they can readily use for algorithmic comparisons. The skin segmentation library provides seven different algorithms all of which use algorithms in different colour spaces to extract skin regions from colour images. This is especially useful in an educational setting where students can use the toolbox to explore the effects of the algorithms, rather than spend time writing code.

C. Example: Template matching

The imageMATCH toolbox contains a function called matchTemplateFFT that performs template matching, i.e. finding a template-image within an image where they differ only by displacement. The function is based on work template matching using FFT, and the cross-power spectrum [7]. Using the built-in Fast Fourier Transform signal processing functions in Julia, fft() and ifft(), it is possible to easily build the function matchTemplateFFT without having to resort to building the FFT functionality. Now it is possible to write a script which uses the toolbox to achieve a match.

The first portion of the code allows access to the appropriate modules. In addition to imageMATCH, this script also uses the two I/O toolboxes, imageIO and PGMImages, which process text image and PGM images respectively. The next piece of code inputs both the reference image (Fig. 1) and the template image (Fig 2). The template matching function matchTemplateFFT is then activated, and the resulting match image output as a text image. The result of the match is shown in Fig. 3, with a box around the identified template region.

VI. CONCLUSION

Image processing is all about crunching numbers, albeit numbers constrained within the guise of an array. Julia may still be too esoteric for widespread adoption, but its speed, statistical abilities and expressiveness make it a potential language for image processing. With speed approaching that of C, it is scaleable, and easy to learn. This latter point is very important, as many individuals wanting to process images may not have extensive programming experience. There are a myriad of reasons to choose Julia as an image processing language. Foremost it provides for fast prototyping without sacrificing speed. Its structure (e.g. types) makes expressing algorithms simpler, and it has a large standard library. It plays well with other languages, and makes it easy to parallelize code. Julia is easy to learn, its code is elegant and clean, and is a dynamic interactive language.

REFERENCES

Abstract—In this paper, we propose a novel method for face recognition under varying lighting conditions. We first transform the face images to the logarithm domain, which makes the dark regions brighter. We then use dual-tree complex wavelet (DTCWT) transform to generate face images that are approximately invariant to illumination changes, and use collaborative representation-based classifier (CRC) to classify the unknown faces to one known class. Our experiments show that our proposed method compares favourably to existing methods under both the noise-free and noisy environments for the Extended Yale Face Database B and the CMU-PIE illumination face database. Our method is easy to implement and it is fast as well because it has linear computational complexity.

Keywords—Face recognition, invariant features, dual-tree complex wavelet transform (DTCWT), collaborative representation-based classifier (CRC).

I. INTRODUCTION

Face recognition is a very popular research topic in such diverse real-life applications as military, commercial, and public security. A huge variety of variations exist in human faces because of illumination, shift, pose, occlusion, and expression. There are three categories of methods to deal with illumination variation in face recognition. The first category of methods extracts illumination invariant features from the faces. The second category of methods creates a 3D shape model to deal with illumination difference in human faces. The third category of methods utilizes image-processing techniques to normalize different illumination conditions. In this paper, we choose the third category of method for invariant face recognition.

In this paper, we propose a novel method for face recognition by using dual-tree complex wavelet transform (DTCWT) [1] to generate illumination invariant face images, and by using collaborative representation-based classifier (CRC) to classify the faces. Our recognition rates are almost always higher than existing methods, except one test case, for the Extended Yale Face Database B and the CMU-PIE illumination face database. Our proposed method does not require any modelling and can be implemented easily. In addition, they can be applied directly to any face image without any lighting assumption or any prior information on 3-D face geometry.

The organization of the rest of the paper is as follows. Section II proposes a new method for robust face recognition. Section III conducts some experiments for the Extended Yale Face Database B and the CMU-PIE illumination database. Finally, Section IV draws the conclusions of the paper.

II. PROPOSED METHOD

In the following, we will briefly discuss the Lambertian reflectance theory, the DTCWT transform, and the CRC classifier. According to the Lambertian reflectance theory [2], the intensity image can be modeled as

$$I(x,y) = R(x,y)L(x,y)$$

(1)

where $R$ is the reflectance and $L$ is the illumination. Because $R$ depends only on the surface material of the subject, it is the intrinsic representation of a face image. Many existing methods attempt to extract the reflectance component for face recognition. However, it is a very difficult problem to solve. The more suitable solution is to convert the intensity image of the face to the logarithm domain, where the multiplication becomes addition:

$$\log(I(x,y)) = \log(R(x,y)) + \log(L(x,y))$$

(2)

In this way, the face recognition problem is easier than before.

Inspired by LOG-DCT [4], in this paper, we propose a new method for illumination invariant face recognition. Due to difference in illumination conditions, the captured face images may be very dark, which makes existing face recognition methods fail. In order to improve classification rates, we propose to elevate the dark region and suppress bright region by means of logarithm transform. We perform DTCWT transform to these logarithm images and set the approximation subband and the two highest frequency DTCWT subbands to zero values. An inverse DTCWT
transform will generate our enhanced face images, which are
approximately invariant to illumination. The reason why we
choose DTCWT is because this transform has approximate
shift invariant property, which is very successful in our
previous works for character recognition, signal denoising,
and shape recognition. Fig. 1 shows the different stages of our
illumination invariant face recognition algorithm.

We give detailed steps of our new face recognition method
as follows:

Step 1. Initialization: J=4.
Step 2. Take the logarithm transform of the intensity image
I(x,y) as equation (2).
Step 3. Perform forward DTCWT transform to log(I(x,y)) for
J scales, denote it as B=DTCWT(log(I), J).
Step 4. Set the approximation subband and the two highest
frequency DTCWT coefficient subbands to zero values.
Step 5. Conduct inverse DTCWT to the output image from
Step 4 in order to obtain face image D.
Step 6. Set E=D^k, where k=0.69 is a constant.
Step 7. Normalize E so that it has zero mean and unit
variance.
Step 8. Use CRC to classify the resulting face image to one
of the known classes.

The contributions of this paper can be described as follows.
In our proposed algorithm, we perform logarithm transform to
make dark regions brighter and we use DTCWT transform to
generate illumination invariant face images. This combination
of logarithm-DTCWT is new to our best knowledge. Our
proposed method does not generate discontinuing region
boundary, and it does not need to smooth the face image as
Gradientfaces [5]. In addition, our new method performs very
well under noisy environment, but the LOG-DCT does not.
Our method is easy to implement and for the majority of test
cases they yield higher recognition rates than existing methods
compared in this paper.

The computational complexity of our new method can be
estimated as follows. Let the input face images have M rows
and N columns. Also, let the number of training images be P
and the number of testing images be Q. The logarithm
transform is of linear complexity. The 2D DTCWT is linear as
well. In addition, Step 6 in our new method has linear
complexity because it calculates the power (k=0.69) for every
pixel. Therefore, the computational complexity of our
proposed method is O(PMN) for training, and O(QMN) for
testing. This means that our new method is linear in time
complexity, which should be fast in real-life applications.

We briefly discuss a few face recognition methods that
features for illumination invariant face recognition. However,
our new method is different from [6] because we not only set
the highest two DTCWT coefficient subbands to zero values
but also set the approximation subband to zero. Furthermore,
we perform inverse DTCWT to obtain illumination invariant
also proposed a new method for face recognition due to the
attractive properties of the DTCWT: approximate shift-
invariance, orientation selectivity, and efficient computation.
Zhang et al. [7] performed one level DTCWT to the face
image and then used SVM as a classifier for face recognition.
In [8] and [9], the authors performed the DTCWT transform
for four decomposition scales, and the high-frequency
subbands are employed to construct the representation of
faces. Nevertheless, they simply utilized DTCWT and did not
study the performance of DTCWT for face representation
under different variations such as shift and illumination.
Principal component analysis (PCA) was used for dimension
reduction, but it was not clear how to select the number of
principal components to retain.

III. EXPERIMENTS

In this section, we conduct a number of experiments for
face recognition by using the Extended Yale Face Database B
[10] and the CMU-PIE illumination face database [11]. The
Extended Yale B database has face images of 38 subjects in
64 diverse lighting conditions: from normal to extremely
badly illuminated. There is only one ideal image for each
person. There are 2414 available images in total and they are
already aligned well. The cropped and normalized face images
of size 192×168 were captured under various laboratory-
controlled lighting conditions. We take one well-lighted face
image as the single reference and take all the rest available
2414-38 = 2376 images as test samples. The faces are divided
into 5 subsets according to angles between the light source
direction and the camera axis:

Table1. The five subsets of the Extended Yale Face Database,
their corresponding angles, and the number of faces in each
subset.

<table>
<thead>
<tr>
<th>Subsets</th>
<th>Angles</th>
<th>Number of Faces</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subset 1</td>
<td>1°≤angle≤12°</td>
<td>7×38</td>
</tr>
<tr>
<td>Subset 2</td>
<td>13°≤angle≤25°</td>
<td>12×38</td>
</tr>
<tr>
<td>Subset 3</td>
<td>26°≤angle≤50°</td>
<td>12×38</td>
</tr>
<tr>
<td>Subset 4</td>
<td>51°≤angle≤77°</td>
<td>14×38</td>
</tr>
<tr>
<td>Subset 5</td>
<td>78°≤angle</td>
<td>19×38</td>
</tr>
</tbody>
</table>

The degree of variation gets higher from Subset 1 to Subset 5.
It is easy to see that our image enhancement method can do an
excellent job in invariant face recognition.

The CMU Pose, Illumination and Expression (PIE)
database contains 41368 images of 68 subjects. Each subject
has images captured under 13 different poses and 43 different
illumination conditions and with four different expressions.
We only select a subset that focuses on illumination variations
on light intensity and direction in frontal view. There exist 68
subjects in each 43 images yielding a total of 2924 images.

For both databases, we choose only one frontally lit face
image in each class for training and the rest face images for
testing. We transform every face image to the logarithm
domain and normalize it to have zero mean and the pixel
values fall in the range of $[0,255]$. We then perform DTCWT transform to these normalized face images for $J=4$ scales and set the DTCWT coefficients to zero for the two highest frequency subbands and the approximation subband. An inverse DTCWT transform will generate the enhanced faces, which are approximately invariant to illumination changes and therefore good for face classification.

We tabulate the correct classification rates of our method and a number of existing methods in Table 2 for both face databases. The correct classification rate is defined as the percentage of faces that are recognized correctly with regard to their true class labels. It can be seen that our proposed method yields higher classification rates except one case in the Extended Yale Face Database B. For subset 1 of the Yale-B face database, our method has a classification rate of 93.8% whereas Large and Small Scale features [12], LOG-DCT, LTV [13] and Local Binary Pattern all obtained perfect classification rate (100%). Xie et al. proposed an illumination normalization technique on large and small scale features for robust face recognition. The average recognition rate of [12] is 91.2% for the Yale-B face database. However, our proposed method achieves 92.3% average classification rate, which is higher than all other methods. This demonstrates the success of our proposed method for robust face recognition for the Yale-B face database. For the CMU-PIE illumination database, our proposed method achieves 100% correct classification rate whereas other methods obtain equal or lower classification rates than our method. In summary, our proposed method in this paper is preferred for illumination invariant face recognition especially under noisy environments.

Table 3 show the correct classification rates (%) of the proposed method in this paper and of LOG-DCT for face images corrupted by Gaussian white noise. The ‘1’ and ‘2’ in the second column means that no noise is added to the reference face images, and noise is added to the reference images, respectively. It can be seen that our proposed method outperforms the LOG-DCT for all test cases except one special case, where for the Extended Yale-B database, the LOG-DCT is better than our proposed method for $\sigma_{n}=5$. However, this is the only case that our proposed method fails to beat LOG-DCT.

IV. CONCLUSION

In this paper, we have proposed a novel method for face recognition by extracting DTCWT faces in logarithm domain. Our method is relatively invariant to illumination changes in the face images. The CRC is used as a classifier in our method. Our proposed method is better than existing methods for nearly all cases in terms of correct recognition rates for the Extended Yale Face Database B. Our method is better than or comparable to all other methods for the CMU-PIE illumination face database. Compared with existing methods for face recognition, our proposed method in this paper is more suitable for recognizing face images with uneven illumination and even for noisy environments. Our method is very simple to implement and they achieve state-of-the-art recognition rates for both the Extended Yale Face Database B and the CMU-PIE illumination face database.

REFERENCES

Table 2. The correct classification rates (%) of the proposed method in this paper, and of the methods: Large and small scale, LOG-DCT, LTV, Local binary pattern, histogram equalization, and no features extraction (None). The best results are highlighted in bold font.

<table>
<thead>
<tr>
<th>Methods</th>
<th>CMU-PIE</th>
<th>EXTENDED YALE-B</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Subset 1</td>
<td>Subset 2</td>
</tr>
<tr>
<td>Proposed</td>
<td>100</td>
<td>93.8</td>
</tr>
<tr>
<td>Large and Small Scale features</td>
<td>99.9</td>
<td>100</td>
</tr>
<tr>
<td>LOG-DCT</td>
<td>100</td>
<td>93.8</td>
</tr>
<tr>
<td>LTV</td>
<td>99.8</td>
<td>100</td>
</tr>
<tr>
<td>Local Binary Pattern</td>
<td>75.4</td>
<td>100</td>
</tr>
<tr>
<td>Histogram Equalization</td>
<td>42.2</td>
<td>99.1</td>
</tr>
<tr>
<td>None</td>
<td>35.1</td>
<td>99.6</td>
</tr>
</tbody>
</table>

Table 3. The correct classification rates (%) of the proposed method in this paper and of LOG-DCT for face images corrupted by Gaussian white noise. The ‘1’ and ‘2’ in the second column means that no noise is added to the reference face images, and noise is added to the reference images, respectively. The best results are highlighted in bold font.

<table>
<thead>
<tr>
<th>Databases</th>
<th>Methods</th>
<th>Noise Standard Deviation (σn)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Proposed 1</td>
<td>87.69</td>
</tr>
<tr>
<td></td>
<td>Proposed 2</td>
<td>86.05</td>
</tr>
<tr>
<td></td>
<td>LOG-DCT 1</td>
<td>88.87</td>
</tr>
<tr>
<td></td>
<td>LOG-DCT 2</td>
<td>82.43</td>
</tr>
<tr>
<td>CMU-PIE</td>
<td>Proposed 1</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>LOG-DCT 1</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>LOG-DCT 2</td>
<td>100</td>
</tr>
</tbody>
</table>

Figure 1. The different stages of our illumination invariant face recognition algorithm: (a) the input face image, (b) the logarithm of the input image, (c) Tree 1 of the DTCWT transform on the logarithm image, (d) Tree 2 of the DTCWT transform on the logarithm image, (e) set the approximation and the two highest frequency scales of tree 1 to zero, (f) set the approximation and the two highest frequency scales of tree 2 to zero, (g) the output illumination invariant face generated by inverse DTCWT.
Dissimilarity-based representation for radiomics applications

Hongliu CAO†‡, Simon BERNARD*, Laurent HEUTTE* and Robert SABOURIN†
*Université de Rouen Normandie, LITIS (EA 4108), BP 12 - 76801 Saint-Étienne du Rouvray, France
†Laboratoire d’Imagerie, de Vision et d’Intelligence Artificielle, Ecole de Technologie Supérieure, Université du Québec, Montreal, Canada

Abstract—Radiomics is a term which refers to the analysis of the large amount of quantitative tumor features extracted from medical images to find useful predictive, diagnostic or prognostic information. Many recent studies have proved that radiomics can offer a lot of useful information that physicians cannot extract from the medical images and can be associated with other information like gene or protein data. However, most of the classification studies in radiomics report the use of feature selection methods without identifying the machine learning challenges behind radiomics. In this paper, we first show that the radiomics problem should be viewed as an high dimensional, low sample size, multi view learning problem, then we compare different solutions proposed in multi view learning for classifying radiomics data. Our experiments, conducted on several real world multi view datasets, show that the intermediate integration methods work significantly better than filter and embedded feature selection methods commonly used in radiomics.

Index Terms—Radiomics, dissimilarity, random forest, feature selection, multi-View learning

I. INTRODUCTION

One of the biggest challenges of cancer treatment is that every tumor is different, known as tumor heterogeneity. It demands for more personalized treatment. The normal process of cancer detection is from certain signs and symptoms to the further investigation by medical imaging and at last confirmed by biopsy. However, with the improvement of medical imaging technology, tumor phenotype characteristics can be visualized in a non-invasive way [1], [2].

Since 2012, by combining the word radiology and the suffix omics, a new term, radiomics, was introduced, which refers to the process of extracting large amount of features from standard-of-care images obtained with CT (computed tomography), PET (positron emission tomography) or MRI (magnetic resonance imaging) to build descriptive, predictive or prognostic models for different cancers [3]. Compared to the current qualitative analysis in radiology, radiomics can provide a quantitative analysis including much more useful information to make optimal treatment decisions and make cancer treatment more effective and less expensive [4]. Many studies focus on the prediction of survival patients or the prediction of the response of patients to the treatment. A lot of classification tasks like classifying between patients with cancer and without cancer have also been done. Radiomics data can also be combined with genomics data and clinical data to improve the accuracy. From a machine learning point of view, radiomics is challenging in three ways:

1) Small sample size: Most of radiomics data sets have no more than 200 patients, many studies have even fewer than 100 patients. And the data sharing is very difficult because of different laws or politics issues.

2) High dimensional feature space: As radiomics aims at extracting large amount of different features from medical images, the feature space is always high dimensional. Even though there is no quantitative definition of ‘large amount’, most of radiomics studies used at least 200 features. For example, in the work of [1], [2], they used over 400 features, and in the work of [5], they used 6746 features.

3) Multiple feature groups: In order to obtain more useful information, multiple feature groups are extracted for radiomics data. These feature groups can come from different sources, different feature extractors or be of different natures, and each of them brings useful and complementary information, e.g. in many radiomics studies, the extracted features come from tumor intensity, tumor shape, texture of the tumor [2]. Exploiting the complementary information that different groups contain is a challenging task for improving the learning performance.

In most of the state-of-the-art works in radiomics, the multiple feature groups are concatenated all together in a very high dimension feature space, which results in an HDLSS (high dimension low sample size) machine learning task. Hence feature selection is a most commonly used method to reduce the dimension. However, if only a small subset of the features is chosen, certainly a lot of useful information will be lost and the heterogeneity of the tumor cannot be well represented. By concatenating all feature groups together into one single view of the patient, the complementary information that different feature groups can offer may be ignored.

In contrast with the current studies that treat radiomics data as a single view machine learning problem, we propose in this paper to consider an HDLSS multi-view learning framework. Multi-View learning is a machine learning framework where data are represented by multiple distinct feature groups, and each feature group is referred to as a particular view [6]. We propose to compare the feature selection solutions commonly
used in radiomics with classical multi-view learning solutions that are meant to improve the performance by exploiting the information from different views of the same input data [6].

The remainder of this paper is organized as follows: the related works in radiomics applications and multi-view learning are discussed in Section II. In section III, two dissimilarity based multi-view learning solutions are introduced. Before turning to the result section (Section V), we describe the data sets chosen in this study and provide the protocol of our experiments in Section IV. The final conclusion and future works are given in Section VI.

II. RELATED WORKS

This section firstly gives a brief overview of radiomics literature from a machine learning point of view. It secondly presents the multi-view learning approaches that could be straightforwardly applied to radiomics.

A. Machine learning methods used in radiomics

In a large majority of radiomics works, the multiple views are concatenated to form a single-view feature vector. As explained in the introduction, it usually leads to HDLSS issues. As a consequence, feature selection methods are systematically used to overcome the difficulty of learning in high dimension.

Feature selection methods are traditionally divided into three main categories: filter, wrapper and embedded methods [7]. Radiomics works have mainly investigated filter selection methods. In general, MRMR (minimum redundancy maximum relevance), RELF (relief) and MIFS (mutual information feature selection) have better results than other filter methods in radiomics [1], [8]. In the work of [8], the authors investigated 24 feature selection methods and three classification methods for histology prediction: RELF showed higher prediction accuracy as compared to other methods in multivariate analysis. RELF was also compared with wrapper methods for classification of progression free survival in the work of [9], and features selected by RELF had better results. For other HDLSS problems, RELF also achieved very good performance [7].

In addition to those filter approaches, a few radiomics works have applied a successful embedded feature selection method, namely SVMRFE (support vector machine recursive feature elimination) [10]. This approach differs from the filter approaches used in radiomics by embedding the feature selection into the learning procedure, so that it can take the resulting classifier performance into account. In the work of [11], [12], the authors showed that SVMRFE had very good performance on radiomics data. For other HDLSS data, SVMRFE also had very good performance in the work of [7].

B. Multi-view learning methods

Multi-view learning methods can be divided into three main categories: early, intermediate and late integration approaches [13]. Early integration refers to the approach discussed in the previous subsection, that is to say concatenating the multiple views and tackling the learning task from the resulting single-view feature space (e.g. with feature selection, and learning afterwards).

Late integration consists in learning a different model separately on each view and aggregating the decisions. The most popular approach in this category is the Co-training method. It is a semi-supervised method that maximizes the mutual agreement on two distinct views, by exploiting the outcomes of two classifiers, one per view, on a subset of unlabeled data [6]. As explained in the introduction section, radiomics datasets are often made up of very few labeled instances and no additional unlabeled instances are available. As a consequence, co-training approaches are not straightforwardly applicable to radiomics learning tasks. Another method for late integration is to use a classical Multiple Classifier System approach (MCS). MCS combines the outputs of different classifiers trained on each view separately to improve total performance. In the work of [14], the authors proposed to use five heterogeneous feature groups which represent different aspects of semantics for identifying health related messages in social media. Then they chose five classifiers for each feature group, and used MCS methods to combine the results together. Simply integrating the results of different classifiers makes MCS fast, efficient and flexible.

Finally, the third category in multi-view learning literature, namely the intermediate integration, is an in-between approach that builds an intermediate representation of the views, for better combining them before learning. A typical example is the dissimilarity-based learning, that projects each view in a space in which the samples are described by their dissimilarities to all the training instances. In that way, each view is separately projected in the same description space. Further details about this principle are given in the next section. In the work of [15], the authors used multi-modalities data for Alzheimer’s Disease patients. To combine four modalities together, they calculated a Random forest similarity matrix for each modality and then fused the four dissimilarity matrices by averaging. Finally multidimensional scaling was used on the joint dissimilarity matrix, followed by a random forest classifier in the embedded dissimilarity space.

C. Discussion

Most of the studies in radiomics have made use of filter selection methods, because they are independent from the classifier, simple to implement and computationally fast. But they may also easily filter some useful information for the classification task, whereas the objective of extracting a large number of features is exactly to bring additional information: if only a small subset of the features is chosen, certainly a lot of useful information is lost and the heterogeneity cannot be well represented. Filter methods also ignore the interaction with the classifier, the search in the feature subset space is separated from the search in the hypothesis space [6].

In contrast to early integration, the two other multi-view learning methods, intermediate and late integration, can deal with feature heterogeneity by introducing one function to model a particular view and jointly optimize all the functions to improve the learning performance [6]. The late integration methods focus more on finding the agreement among views,
whereas the intermediate methods use the same idea as in data fusion to generate a better representation of data by taking advantage of complementary information contained in each view. Intermediate integration seems thus more appropriate for radiomics data.

III. DISSIMILARITY-BASED LEARNING

As discussed above, intermediate integration methods can generate a better representation of data by taking advantage of the complementary information contained in each view. However, the question of how to integrate information coming from different views is a challenge because different views may have different feature types, feature sizes, and are not directly comparable. Projecting each view of the data in some dissimilarity space can offer a smart solution to that issue as views become comparable from one dissimilarity space to another (same feature type, same feature size) and the dimension of the initial HDLSS data is reduced.

**Dissimilarity matrix:** Let \( T = \{ (X_1, y_1), (X_2, y_2), \ldots, (X_N, y_N) \} \) denote a training set made up of \( N \) instances \( X_i \) from a domain \( X \), each one labeled with its true class \( y_i \). A dissimilarity measure \( d \) is a function from \( X^2 \) to \( \mathbb{R}^+ \) that estimates how dissimilar two instances are. For two given instances \( X_i \) and \( X_j \), a high value \( d(X_i, X_j) \) means that the two instances are very "different", while on the opposite, a low \( d(X_i, X_j) \) means they are very similar; in particular, \( d(X_i, X_i) = 0 \).

For classification problems, the dissimilarity between two instances from the same class is expected to be small, while on the contrary the dissimilarity between two instances from two different classes is expected to be high.

Now, let \( D \) denote a \( N \times N \) dissimilarity matrix, built from a given dissimilarity measure \( d \) and from a training set \( T \), as defined in Equation (1):

\[
D = \begin{bmatrix}
d_{11} & d_{12} & d_{13} & \ldots & d_{1N} \\
d_{21} & d_{22} & d_{23} & \ldots & d_{2N} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
d_{N1} & d_{N2} & d_{N3} & \ldots & d_{NN}
\end{bmatrix}
\]

where \( d_{ij} \) denotes \( d(X_i, X_j) \), for all \( X_i, X_j \in T \).

\( D \) is non-negative and respects the reflexivity condition. Such a dissimilarity matrix can be viewed as a new training set, where each training instance \( X_i \) is described by a vector \( (d_{i1}, d_{i2}, \ldots, d_{iN}) \). In the same way, using its dissimilarity to each of the training instances, any new instance \( X \) can be mapped into a \( N \) dimensional dissimilarity space \( DS \).

For HDLSS data, the dimension of this dissimilarity space is necessarily smaller than the dimension of the original feature space.

Typically, a distance measure such as the Euclidean distance can be used to measure dissimilarities. However, such a measure does not capture the class membership, which is an important criterion for classification tasks to tell whether or not two instances are similar. Compared to such a distance function without class information, class-based dissimilarity measures are more powerful, e.g. the Random Forest dissimilarity measure [16].

**Random Forest:** Given a training set \( T \), a Random Forest classifier \( H \) is a classifier made up of \( M \) trees, denoted as in Equation (2):

\[
H(X) = \{ h_k(X), k = 1, \ldots, M \}
\]

where \( h_k(X) \) is a random tree grown using bagging and the random feature selection [17]. For predicting the class of any new instance \( X \) with such a tree, \( X \) goes down the tree structure, from its root till one of its terminal nodes (or leaves). The descending path is described by successive tests on the values of the features of \( X \), one per node. The prediction is given by the terminal node in which \( X \) falls. We refer the reader to [17] for more information about this process.

Hence if two different instances fall in the same terminal node, they are likely to belong to the same class and they are also likely to share similarities between features, since they have followed the same descending path.

**Random Forest Dissimilarity (RFD):** the RFD measure is inferred from a RF classifier \( H \), trained from \( T \). Let firstly define a dissimilarity measure \( d^k \) inferred by the \( k \)-th decision tree \( h_k \): let \( L_k \) denote the set of leaves of \( h_k \), and let \( l_k(X) \) denote a function from \( X \) to \( L_k \) that returns the leaf node of \( h_k \) where a given instance \( X \) falls when one wants to predict its class. The dissimilarity measure \( d^k \), inferred by \( h_k \), is defined as in Equation (3): if two training instances \( X_i \) and \( X_j \) fall in the same leaf of \( h_k \), then the dissimilarity between both instances is set to 0, else to 1.

\[
d^k(X_i, X_j) = \begin{cases}
0, & \text{if } l_k(X_i) = l_k(X_j) \\
1, & \text{otherwise}
\end{cases}
\]

The RFD measure \( d^H \) consists in calculating the \( d^k \) value for each tree \( h_k \) in the forest, and to average the resulting dissimilarity values over the \( M \) trees, as in Equation (4):

\[
d^H(X_i, X_j) = \frac{1}{M} \sum_{k=1}^{M} d^k(X_i, X_j)
\]

**Multi-view learning dissimilarities:** For multi-view learning tasks, the training set \( T \) is composed of \( Q \) views: \( T^{(q)} = \{ (X_1^{(q)}, y_1), \ldots, (X_N^{(q)}, y_N) \} \). Firstly, for each view \( T^{(q)} \), the RFD matrices are computed as in Equation 1, and noted \( \{ D_{H}^{(q)}, q = 1..Q \} \). In multi-view learning, the joint dissimilarity matrix \( D_H \) can typically be computed by averaging over these matrices (cf. Equation (5)).

\[
D_H = \frac{1}{Q} \sum_{q=1}^{Q} D_{H}^{(q)}
\]

For multi-view learning, this joint dissimilarity matrix \( D_H \) can be used in two ways, either by using \( D_H \) as a kernel matrix (denoted RF SVM method) or by using \( D_H \) as a new training set (denoted RFDIS method):

1) **Multi-view Random Forest kernel SVM (RF SVM):**

From the joint RFD matrix \( D_H \) of Equation (5), one
can calculate the joint similarity matrix $S_H$ as $S_H = 1 - D_H$ where 1 is a matrix of ones. For SVM classifier, apart from the most used Gaussian kernel, user defined kernel matrix is also popular. Many studies have been done on user defined kernel matrix. For example, in the work of [18], they proposed to use the problem specific distance measure to construct a substitution Gaussian kernel. Similar to the idea in the work of [19], the joint similarity matrix $S_H$ inferred from the RF classifier $H$ is then used as a kernel matrix in a SVM classifier.

2) **Multi view random forest dissimilarity (RFDIS):**

RFDIS consists in learning a RF classifier $H$ as if $D_H$ was a new training set. It is similar to the method described in [15]. The joint dissimilarity vector is seen as a feature vector, and a random forest classifier is built on these new features.

### IV. EXPERIMENTS

A. **Description of the data sets**

In this study, we use several publicly available HDLSS multi-view datasets. A general information of all datasets can be found in Table I. The first four datasets are radiomics data. There are five views for each of these four datasets: four texture feature groups from axial T1-weighted MR images before and after gadolinium-based CE material administration as well as axial T2-weighted and axial T2-weighted fluid attenuated inversion recovery (FLAIR) images; the fifth view is made up of vasari features. More details about this dataset can be found in the work of [5]. LSVT is a dataset on vocal performance degradation of Parkinson’s disease subjects with four groups of features extracted: physiological observation, signal-to-noise ratio measure, wavelet measure and Mel frequency cepstral coefficients [20]. Metabolomic contains biomarkers (CEA and TIMP), fluorescence concentration (PF) and NMR profiles for early detection of colorectal cancer [21]. Cal7 [22], Cal20 [22] and Mfeat [23] are image classification data using different purpose, six methods are compared: RELF and SVMRFE, the two feature selection methods mostly used in radiomics; RF SVM and RFDIS, the two intermediate integration methods described above; LateRF and LateRFDIS, two late integration methods. LateRF method builds a random forest classifier $H^{(q)}$ for each view of the data $T^{(q)}$, and then combines the results together by majority voting. LateRFDIS method firstly creates a RF dissimilarity space $DS^{(q)}$ for each view $q$, then builds a random forest classifier in each $DS^{(q)}$, and finally combines the results together by majority vote. Here random forest has been chosen because it can deal well with different data types and mixed variables, which can avoid searching for the best classifiers according to data types, data size or data complexity for each view [16].

For the two feature selection methods, it is necessary to fix the number of selected features. However it is very difficult to find the best threshold, most authors just take the top-1 or top-5 features. Following the experiments of [7], the rules we use to fix the number of selected features from the total number $n$ of initial features are the following:

i) if $n < 10$, then select 75% of features

ii) if $10 < n < 75$, then select 40% of features

iii) if $75 < n < 100$, then select 10% of features

iv) if $100 < n < 1000$, then select 3% of features

v) if $n > 1000$, then select 25 features

For both feature selection methods, a random forest classifier is then built with selected features.

For RF SVM, the search range of parameter $C$ for SVM is $[0.01, 0.1, 1, 10, 100, 1000]$. For all random forest classifiers, the number of trees is set to 500 while other parameters are set by default with scikit-learn package for python.

Note also that in [25], the authors found that when dealing with HDLSS data, stratification of the sampling is central for obtaining minimal misclassification. In this work, a stratified repeated random sampling approach was used to achieve a robust estimate of the performance. The stratified random splitting procedure is repeated 10 times, with 50% sampling rate in each subset. In order to compare the methods, the mean and standard deviations of accuracy were evaluated over 10 runs.

### V. RESULTS

A. **Results on non-radiomics data**

For the experiments, the results on the seven non-radiomics HDLSS multi-view datasets are firstly presented. For data splits with 50% training data and 50% test data, the results of mean and standard deviation of accuracy over 10 repetitions are shown in Table II. It can be seen that in general the
intermediate method RFSVM performs the best among all the six methods, and is ranked first for six of the seven datasets. By looking at the average ranking, the two dissimilarity based intermediate integration methods are the best, while the two feature selection methods are the worst.

To see more clearly the difference, a pairwise analysis based on the Sign test is computed on the number of wins, ties and losses as in the work of [26]. The result is shown in Figure 1. The two intermediate and the two late integration methods are compared to RELF in Figure 1 (a) and compared to SVMRFE in Figure 1 (b). Each vertical line indicates the critical value corresponding to a confidence level \( \alpha \). If the number of wins is greater than or equal to a critical value, it means that the corresponding method is significantly better than the baseline method. Figure 1 (a) shows that with \( \alpha = 0.10 \) and 0.05, the four multi-view methods are significantly better than RELF. Figure 1 (b) shows that the two late integration methods are not always significantly better than SVMRFE, but the two intermediate integration methods are significantly better than SVMRFE with \( \alpha = 0.05 \). The results on the seven non-radiomics data confirm our hypothesis that dissimilarity based intermediate integration methods are significantly better than the state of art feature selection methods used in radiomics.

**TABLE II**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>RELF +RF</th>
<th>SVMRFE +RF</th>
<th>RFSVM</th>
<th>RFDIS</th>
<th>LateRF</th>
<th>LateRFDIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSV1</td>
<td>81.11%</td>
<td>84.12%</td>
<td>84.14%</td>
<td>81.45%</td>
<td>80.47%</td>
<td>81.42%</td>
</tr>
<tr>
<td></td>
<td>5.04</td>
<td>3.48</td>
<td>2.93</td>
<td>1.97</td>
<td>2.01</td>
<td>2.66</td>
</tr>
<tr>
<td>Metabolic</td>
<td>60.83%</td>
<td>63.54%</td>
<td>68.75%</td>
<td>67.11%</td>
<td>64.58%</td>
<td>64.37%</td>
</tr>
<tr>
<td></td>
<td>6.37</td>
<td>7.53</td>
<td>5.10</td>
<td>5.12</td>
<td>7.63</td>
<td>6.55</td>
</tr>
<tr>
<td>Cal</td>
<td>90.18%</td>
<td>94.57%</td>
<td>96.36%</td>
<td>95.21%</td>
<td>91.66%</td>
<td>93.98%</td>
</tr>
<tr>
<td></td>
<td>0.86</td>
<td>0.47</td>
<td>0.67</td>
<td>0.46</td>
<td>0.73</td>
<td></td>
</tr>
<tr>
<td>Cal20</td>
<td>75.01%</td>
<td>85.06%</td>
<td>88.39%</td>
<td>89.12%</td>
<td>81.57%</td>
<td>86.15%</td>
</tr>
<tr>
<td></td>
<td>0.94</td>
<td>1.98</td>
<td>0.34</td>
<td>0.69</td>
<td>0.33</td>
<td>0.58</td>
</tr>
<tr>
<td>Mfeat</td>
<td>89.13%</td>
<td>91.13%</td>
<td>97.83%</td>
<td>97.56%</td>
<td>96.86%</td>
<td>96.50%</td>
</tr>
<tr>
<td></td>
<td>2.49</td>
<td>4.12</td>
<td>0.95</td>
<td>0.99</td>
<td>1.11</td>
<td>1.26</td>
</tr>
<tr>
<td>BBC</td>
<td>59.92%</td>
<td>74.19%</td>
<td>95.63%</td>
<td>92.92%</td>
<td>92.12%</td>
<td>88.88%</td>
</tr>
<tr>
<td></td>
<td>1.89</td>
<td>1.70</td>
<td>0.39</td>
<td>0.67</td>
<td>0.62</td>
<td>0.50</td>
</tr>
<tr>
<td>BBCSport</td>
<td>56.33%</td>
<td>78.05%</td>
<td>95.58%</td>
<td>81.75%</td>
<td>88.13%</td>
<td>81.61%</td>
</tr>
<tr>
<td></td>
<td>4.08</td>
<td>2.88</td>
<td>0.81</td>
<td>2.70</td>
<td>1.78</td>
<td>1.96</td>
</tr>
<tr>
<td>Average Rank</td>
<td>5.82</td>
<td>4.23</td>
<td>1.55</td>
<td>2.55</td>
<td>3.23</td>
<td>3.64</td>
</tr>
</tbody>
</table>

**TABLE III**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>RELF +RF</th>
<th>SVMRFE +RF</th>
<th>RFSVM</th>
<th>RFDIS</th>
<th>LateRF</th>
<th>LateRFDIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>mniH1</td>
<td>76.60%</td>
<td>76.28%</td>
<td>80.09%</td>
<td>79.53%</td>
<td>82.79%</td>
<td>80.63%</td>
</tr>
<tr>
<td></td>
<td>5.55</td>
<td>4.39</td>
<td>3.57</td>
<td>3.57</td>
<td>2.37</td>
<td>2.51</td>
</tr>
<tr>
<td>IDHcode</td>
<td>72.94%</td>
<td>73.23%</td>
<td>76.76%</td>
<td>76.47%</td>
<td>76.97%</td>
<td>76.17%</td>
</tr>
<tr>
<td></td>
<td>4.89</td>
<td>5.50</td>
<td>4.52</td>
<td>3.95</td>
<td>2.06</td>
<td>2.06</td>
</tr>
<tr>
<td>lowGrade</td>
<td>60.40%</td>
<td>62.55%</td>
<td>63.95%</td>
<td>63.48%</td>
<td>64.41%</td>
<td>65.11%</td>
</tr>
<tr>
<td></td>
<td>5.79</td>
<td>3.36</td>
<td>4.56</td>
<td>3.76</td>
<td>3.76</td>
<td>5.20</td>
</tr>
<tr>
<td>progression</td>
<td>60.26%</td>
<td>62.36%</td>
<td>65.52%</td>
<td>63.42%</td>
<td>61.31%</td>
<td>58.94%</td>
</tr>
<tr>
<td></td>
<td>4.92</td>
<td>3.73</td>
<td>4.47</td>
<td>4.69</td>
<td>4.25</td>
<td>6.02</td>
</tr>
<tr>
<td>Average Rank</td>
<td>5.75</td>
<td>4.90</td>
<td>2.12</td>
<td>3.25</td>
<td>2.12</td>
<td>3.25</td>
</tr>
</tbody>
</table>

**B. Results on radiomics data**

We now show that our hypothesis, validated on the seven non-radiomics datasets, can be confirmed on four real world radiomics datasets. The results of mean and standard deviation of accuracy over 10 repetitions are presented in Table III. The best performance on the four radiomics datasets are achieved two times by RFSVM, two times by LateRF and once by LateRFDIS. By looking at the average ranking, RFSVM and LateRF are ranked at the first place, while RFDIS and LateRFDIS are ranked at the second place.
non-radiomics data, which shows great potential for solving radiomics problem.

VI. CONCLUSIONS

In this paper, we have tackled the problem of radiomics data classification as an HDLSS multi-view learning task. Contrary to the most commonly used methods in radiomics that concatenate the multiple feature groups into a single one view in an early integration manner and then select the best features for classification, we have shown that intermediate and late integration methods can offer better benefits for taking advantage of the complementary information brought by each view than the early integration methods.

To confirm our hypothesis, we have compared two representative early integration methods (RELF and SVMREFE), two dissimilarity based intermediate integration methods (RFSVM and RFDIS) and two late integration methods (LateRF and LateRFDIS), across seven real-world non-radiomics datasets and four real-world radiomics datasets. Our experiments have shown that the two intermediate integration methods, RFSVM and RFDIS, are significantly better than the state-of-the-art early integration methods with RELF and SVMREFE. We have also shown that even if LateRF and LateRFDIS methods are better than the two early integration methods with feature selection, they are not significantly better than SVMREFE. We can conclude that, for radiomics like data, the dissimilarity-based intermediate integration methods are a better alternative than the commonly used early integration methods.

As part of our future works, we aim at improving the quality of the dissimilarity space for each view by adapting the hyperparameters of the random forest based dissimilarity measure, as at present the same hyperparameters are used for each view. In this work, the two dissimilarity-based intermediate integration methods treat all the views equally, but a weighted combination could have also been used to generate a better joint dissimilarity matrix. Finally, none of the datasets we have tested in this work contain missing values and missing views. But in our future works we have to deal with that issue.

ACKNOWLEDGMENT

This work is part of the DAISI project, co-financed by the European Union with the European Regional Development Fund (ERDF) and by the Normandy Region.

REFERENCES


Oral Session C

Machine Learning and deep networks

Tuesday May 15, 2018, 10:00 AM

Dan Harborne, Chris Willis, Richard Tomsett and Alun Preece
  Integrating Learning and Reasoning Services for Explainable Information Fusion

Mahdi Abolfazli Esfahani, Keyu Wu, Shenghai Yuan and Han Wang
  A New Approach to Train Convolutional Neural Networks for Monocular Visual Odometry

Qicheng Lao, Haoran Sun and Thomas Fevens
  Segmentation-Free Cell Phenotype Classification using Deep Residual Neural Networks

Lin Bai, Lina Yang, Yuanyan Tang, Lin Huo and Taoshen Li
  Hybrid Deep Neural Network for Visual Phrase Detection

Katsuhiko Yoda, Wataru Kanemori, Mitsuru Tomono, Makiko Ito and Teruo Ishihara
  Training Quantized Nets with Adaptive Shared Exponents Based on Statistical Distributions
Abstract—We present a distributed information fusion system able to integrate heterogeneous information processing services based on machine learning and reasoning approaches. We focus on higher (semantic) levels of information fusion, and highlight the requirement for the component services, and the system as a whole, to generate explanations of its outputs. Using a case study approach in the domain of traffic monitoring, we introduce component services based on (i) deep neural network approaches and (ii) heuristic-based reasoning. We examine methods for explanation generation in each case, including both transparency (e.g., saliency maps, reasoning traces) and post-hoc methods (e.g., explanation in terms of similar examples, identification of relevant semantic objects). We consider trade-offs in terms of the classification performance of the services and the kinds of available explanations, and show how service integration offers more robust performance and explainability.

Index Terms—information fusion, explainability, interpretability, machine learning, reasoning, distributed systems

I. INTRODUCTION

Our goal is to facilitate situational understanding by human analysts by providing an open, distributed information fusion architecture that integrates machine learning and reasoning services operating on multimodal data feeds. We broadly subscribe to the view that an integrated learning and reasoning system can be viewed as a dynamic composition of simpler models such that the composed system is able to answer questions that the individual models cannot [1]. In our case, we require that the system not only provide human analysts with classified or inferred assessments of a situation, but also be able to provide explanations for its assessments in terms of a typology of model interpretability [2]. Moreover, we aim to take advantage of the compositional architecture by exploiting semantic relationships between model outputs to improve generated explanations, especially for information fusion processes involving sub-symbolic models where interpretability remains a challenging problem, including deep neural networks, DNNs [3].

We require that the architecture be distributed to allow for information processing (including pattern recognition and inference) to occur anywhere on the network, and also to allow sharing of heterogeneous services from multiple providers. Sharing brings the additional requirement that there must be a degree of control over service access and information flow, e.g., for privacy or security reasons.

The work presented in this paper is framed as a case study, in which a representative situational understanding problem and available multimodal data feeds facilitate experiments in:
1) integrating machine learning and reasoning models via a lightweight service-oriented architecture;
2) exploring trade-offs between (i) model performance for situational assessment and (ii) interpretability for explanation generation; and
3) enabling the imposition of constraints to control information flow.

The paper is organised as follows: Section II summarises prior work in information fusion and explainability; Section III introduces the application used as the basis for our case study; Section IV describes the main services: a machine learning service based on a convolutional neural network (CNN) and a composition of services that comprise a reasoner; Section V describes the explanations generatable for each service; Section VI concludes, pointing to future work.

II. RELATED WORK

Our work aims to support situational understanding via information fusion, with a focus on higher fusion levels and human-in-the-loop processes [4]. While much work in fusion of multimodal data addresses signal processing techniques applied at lower (data) levels (e.g., [5]) our interest in integrating heterogeneous machine learning and reasoning services leads us to consider late fusion at higher (information) levels (e.g., [6]). Moreover, we require services to provide explanations for their outputs and, where outputs are fused, we require the fusion of those explanations also.

Explainability in artificial intelligence systems has been recognised as a problem for several decades, and was extensively studied in the 1980s and 1990s in the context of symbolic reasoning (‘expert’) systems (e.g., [7], [8]). The focus then was on effective means to make the reasoning of knowledge-based systems transparent to end-users. Fundamentally, the approaches sought to frame explanations in terms of reasoning traces (e.g., chains of rule firings or proof trees) and component data (i.e., input data that triggered the rules or grounded the proof). While non-trivial, this work benefited from these systems being symbolic rather than sub-symbolic; their internal elements were largely explicable.

The resurgence of interest in sub-symbolic approaches in recent years, chiefly DNNs, has led renewed interest and
concerns regarding explainability\(^1\). System transparency remains a key issue, but it is also recognised that ‘traces’ in terms of DNN weights are not explicable and, often, post-hoc explanations are more useful [2]. The dominant approach to addressing transparency in DNNs is in image classification systems, to associate an output class with the parts of an input image that had the greatest weight in determining the classification; e.g., saliency mapping identifies regions of similarly-weighted pixels in an input image that contribute positive weight towards a particular output class [9], [10].

Common post-hoc explanation approaches for DNNs include explanation-by-example and text explanation. The former draws on earlier work in case-based reasoning wherein an explanation is framed in terms of a selection of labelled cases computed to be similar to the input case [11], [12], [13]. The latter employs object detection techniques to identify meaningful sub-elements of the input and constructs an explanation based on these elements, in a manner similar to automatic image captioning [14].

III. TESTBED APPLICATION

Our testbed was chosen as an exemplar application where open data and pre-existing services were readily available, and where machine learning and reasoning services could plausibly operate at a range of semantic scales (low to high-level). We selected the problem of monitoring and predicting traffic congestion. In many cities, multiple organisations offer open sources of information. For example, Transport for London (TfL) offers an application programming interface (API) to view imagery and video from their traffic cameras placed around London\(^2\), while Open Street Maps (OSM) offers information about roads, e.g., speed limit\(^3\).

A. Testbed System Architecture and Design

The structure of the system is shown in Figure 1. The figure shows information flows from data sources through processing services to decision-support classifications. Dark grey arrows show the path information takes to produce congestion classifications (ratings) and light grey arrows show information flows that produce explanations for those ratings. There are two service chains producing classifications: one via a CNN (only) and a second via a more complex composition of services that feed in to a reasoning service. We refer to the CNN as the congestion classifier and the latter as the congestion reasoner. Section IV details these services and Section V examines the kinds of explanations that can be generated from each.

B. Data Collection and Labelling

TfL provides still images and video sequences (of a few seconds duration) from over 1,200 cameras, released at a 5-minute refresh rate. This paper focuses principally on the still images; these are 352×288 pixels. We collected images and video sequences from 691 cameras over a period of 23 days, including imagery at all times of the day and night, in different location types, and over a range of traffic and environmental conditions. See Figure 2 for example images illustrating some of the variations in lighting, traffic and road configuration present in the dataset.

A subset of the imagery covering five locations over a 24 hour period was selected for ground truth labelling. For this, a web-based image annotation tool\(^4\) was developed to allow users to label each image as one of: ‘congested’, ‘uncongested’, ‘unknown’ or ‘broken’. Users were guided to use the first two labels for images where they were confident that the traffic was congested or not, to use ‘unknown’ where they were uncertain, and ‘broken’ to label images that were blank or otherwise unreadable (an artifact of the TfL system). Use of this tool by 12 users yielded a dataset consisting of 4,117 images labelled by at least one annotator. Once the broken images were rejected a total of 3,967 usable labelled examples remained. Taking the most prevalent labelling for each image revealed that 57% were uncongested and 43% congested. In order to take account of the non-binary nature of the phenomena of interest, and differing opinions expressed

---

\(^1\)Recent machine learning literature favours the term ‘interpretability’ over ‘explainability’. We use the terms interchangeably here.

\(^2\)http://www.trafficdelays.co.uk/london-traffic-cameras/

\(^3\)http://www.openstreetmap.org/

\(^4\)https://image-annotation.eu-gb.mybluemix.net/
when labelling the imagery, soft class memberships were derived for use in training and validation. These took into account the labels supporting a class, against the class and for the ‘unknown’ labels:

\[ p_i = \frac{v_i + u + 1}{\sum_{j=1}^{N} (v_j + u + 1)} \]

Here, \( v_i \) are the number of annotations assigned to class \( i \) and \( u \) are the number of annotations for the unknown class. This soft class membership avoids issues caused by the hard classification of borderline cases adversely biasing results. For classifier development the images were randomly partitioned into a training set of 80% of images and 10% for each of the validation and test sets. (This approach resulted in acceptably good performance for our purposes with relatively fast training; for a production system we would perform cross-validation.)

The services are integrated using the Node-RED programming environment for Internet of Things applications (nodered.org). System code and data is available at: https://osf.io/wm3t9

IV. MACHINE LEARNING AND REASONING SERVICES

A. CNN Congestion Classifier

The congestion classifier uses a pair of DNNs. The first uses the GoogLeNet [15] Inception network, pre-trained on ImageNet data, for feature extraction. A feature vector for transfer learning is tapped-off before the network’s fully connected layer. The feature vector is passed into a five-layer fully-connected network to transform image features into a classification assessment. This output (congestion rating) is the congestion class conditional probability. Based on previous experience, we used hyperbolic-tangent and softmax activation functions and trained the network using the Adam optimizer until the cross-entropy on the validation set stops improving. Both networks are implemented in TensorFlow [16].

Training Method: The training, validation and test datasets were generated by collecting a number of 200×200 pixel sub-images from each starting image. Firstly, the central sub-image was collected, from which we produced eight overlapping images formed by offsetting the selection region by 25 pixels to the left, right, up and down. This was repeated for left-right flipped variants yielding an augmented dataset of over 57,000 training and 7,000 validation and test images respectively. The sub-images generally omit the TfL annotations at the edges of the originals (Figure 2).

Results: A selection of classified images are shown in Figure 2: an image classified as uncongested on the left, a borderline case in the centre and an image classified a congested on the right. Performance on the test set as a whole (Table I) indicated that the proportion of images classified according to most prevalent label was over 95%, and examination of the miss-classified images revealed that most were on the classification borderline (with probability close to 50%). The recorded precision and recall are 0.98 and 0.96 respectively.

<table>
<thead>
<tr>
<th>Predicted class</th>
<th>True class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Congested</td>
<td>1616</td>
</tr>
<tr>
<td>Not Congested</td>
<td>35</td>
</tr>
<tr>
<td>Total</td>
<td>1651</td>
</tr>
</tbody>
</table>

TABLE I

CONFUSION MATRIX FOR CNN CONGESTION CLASSIFIER (TFL IMAGERY FROM 5 LOCATIONS OVER A 24 HOUR PERIOD)

B. Congestion Reasoner

As shown in Figure 1, the congestion reasoner involves an integration of multiple services and data sources and was intentionally crafted as a lightweight heuristic-based service.

Development Method: Video clips from the TfL API are passed to an optical flow (OF) algorithm\(^5\) which generates information in the form of blobs in motion. Semantically, these detected entities cannot be assigned any meaning and we are unsure as to whether the blobs are cars, people or something else. Therefore, to make this information more usable, key frames of the video are sent to an object detector (a retrained instance of the VGG-16 regional-convolutional neural network, R-CNN, model [17]) able to identify the cars in the frames. We fuse the information regarding the detected cars together with the detected blobs (including their pixel velocities across the video) which results in a fused output of detected cars and their pixel velocities.

\(^5\)Mixture of Gaussian (MOG2) & Lucas-Kanade (github.com/itseez/opencv)
Only using the R-CNN on key frames (specifically every 30 frames) reduces the computational cost of the service chain (given that the OF algorithm is comparatively inexpensive when compared to the R-CNN). This approach also demonstrates how services can be used within a service composition whilst adhering to possible usage constraints (such as limiting the number of calls that can be made to a service).

Once the pixel velocities of the cars for a video are obtained, they can be averaged and this can be compared to the speed limit of the location. From the ratio between the velocity and speed limit, a conclusion can be drawn about the value range that the ratio commonly moves within. From this we can derive a heuristic that determines at what point the ratio indicates low and high levels of traffic flow, allowing us to infer the level of congestion.

Results: For the generation of results for this service chain, ground truth for TfL video was required. To produce this, we utilised the labels assigned during the CNN image data set generation. These images were originally produced by pulling frames from the TfL videos from across a single day. As such, the labels assigned to these frames could be aggregated per video to provide a ground truth label for the source video and these labeled videos (733 in total) could form the data set for testing. The performance of the reasoner on the test set is shown in Table II. The recorded precision and recall are 0.79 and 0.87 respectively.

<table>
<thead>
<tr>
<th>Predicted class</th>
<th>Congested</th>
<th>Not Congested</th>
<th>True class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Congested</td>
<td>283</td>
<td>51</td>
<td>334</td>
</tr>
<tr>
<td>Not Congested</td>
<td>91</td>
<td>308</td>
<td>399</td>
</tr>
<tr>
<td>Total</td>
<td>374</td>
<td>359</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Explanation</th>
<th>Detection technique</th>
<th>Supporting services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transparent</td>
<td>Salient map</td>
<td>•</td>
</tr>
<tr>
<td></td>
<td>Component data</td>
<td>•</td>
</tr>
<tr>
<td></td>
<td>Reasoning trace</td>
<td>•</td>
</tr>
<tr>
<td>Post-hoc</td>
<td>Semantic objects</td>
<td>•</td>
</tr>
<tr>
<td></td>
<td>Training examples</td>
<td>•</td>
</tr>
</tbody>
</table>

TABLE III
EXPLANATION AVAILABILITY AND SUPPORTING SERVICES FOR CLASSIFIER AND REASONER SERVICE CHAINS

V. EXPLANATION GENERATION

In this section, we outline explanation generation techniques applicable to our two congestion detection methods, together with additional services used to support them; Table III summarises these.

A. CNN Congestion Classifier Explanations

Transparent explanation via saliency map: Transparent explanations are those which are generated using the signals that pass through the DNN model and thus aim to directly indicate the reasons for model output. For this work we used the LIME [9] tool to generate a saliency map in the form of highlighted regions of the original input. These region were important in the model’s assessment of the likelihood of the output label (strong evidence supporting the label or strong evidence against it). Figure 3 shows an input image and its corresponding saliency map derived using LIME.

‘Map forms’ of explanation are a common approach to adding explainability to black box models. LIME and similar techniques (such as deep Taylor decomposition [18]) that produce maps have important common characteristics that define how they can be used to provide interpretations. One key characteristic is that, in order for an explanation to be human-interpretable, it will often need to be presented in the modality of, and in relation to the original input. This can lead to challenges with sharing the explanation if the original input data cannot be presented for some reason (e.g., privacy). The next section shows how post-hoc explanations that do not reveal the input data can address this problem.

Post-hoc explanation via salient semantic object identification: Unlike transparent explanations, post-hoc explanations do not directly derive from the signals of a model but still can provide justification for, or further insight into, the model’s output. In the technique described in this section, the post-hoc explanation is generated using a transparent explanation (saliency map) as input rather than using the original raw input data. This provides assurance that the post-hoc explanation is related to the decision process of the model.

The light grey arrows in Figure 1 show the passing of the saliency map from LIME to the R-CNN detector that produces a list of detected salient semantic objects (SSOs)

6Our heuristic function for computing the traffic flow ratio $f$ where $s$ is the location speed limit is: $f = \tanh(b \times \tanh(s \times 0.08))$. If $f \leq 0.4$ we infer congestion. The constants 0.08 and 0.4 were empirically derived from observations of traffic behaviour at a range of locations.
that are present within the highlighted regions of the salient map and which have a close semantic relationship to the target label (‘congestion’). To detect the SSOs, the saliency map is used to mask the original input image leaving only the regions of the image that had the highest impact on the model’s classification decision. The set of semantically-relevant objects is provided by a knowledge base (KB); in our current work this is a custom KB, but open KBs can be used instead (e.g., conceptnet.io) Given a set of semantically-relevant objects, an open situational understanding system could in principle automatically discover available services trained to detect the relevant objects. In our case study, we already had a service available for detecting cars in imagery (the R-CNN), created for use in the reasoner service chain.

This explanation technique generates a list of SSOs which is meaningful in absence of the input image. An example is shown in Figure 4. As mentioned above, this allows the original image to remain private (the explanation service can return the classification and the SSO list as explanation without returning the input image). Moreover, where network bandwidth is very low, transmitting the object list requires a fraction of the bandwidth required to transmit even a thumbnail image.

The SSO list is also easily machine-processable, unlike the saliency map which is intended for direct interpretation by a human user. The SSO list can be used to verify whether the classifier has used features that are semantically relevant to the target class. The system can detect unusual cases (e.g., en empty SSO list) and automatically flag a classification as requiring the attention of a human, which can help direct user attention to cases where a human confirmation of a classification is necessary.

Post-hoc explanation via similar training set examples: A common post-hoc explanation technique involves retrieving similar previous cases, usually from the training set [2]. For this explanation to be available, the trained classifier must first be used to provide a rating for the original training set images (or a subset of them). When an explanation is required for a live classification, the system can look for similar classification ratings within the ratings of the training data and present examples alongside the current input image. This indicates what the model considers to be similar presence levels of the output class.

In our case study, for an image with congestion class conditional probability \( p \), we display one example with a class conditional probability close to \( \max(0,p - \epsilon) \) and one with a probability close to \( \min(p + \epsilon,1) \). We considered that this would provide a well-rounded impression of the model’s internal representation. Figure 5 shows an example. Currently, \( \epsilon = 0.1 \); however, this can be configured as desired and further research could be conducted to explore the preferences of users.

![Fig. 5. Explanation by training set examples: (left) training set example with congestion rating 0.1 less than input, (middle) input image, (right) training set example with congestion rating 0.1 more than input](image)

B. Congestion Reasoner Explanations

Transparent explanation via component data: As the congestion reasoner uses a composition of services, the system is able to offer a transparent explanation in the form of the components of information that were used to make the final classification. Here, the speed limit, the traffic flow and the weighted ratio rating between the two are given to the decision agent in order to provide interpretability of the final conclusion generated by the service.

Transparent explanation via reasoning trace: As common in rule-based systems, the ability to easily explain inferences by providing a trace of the system’s reasoning is possible. In our system, a rule is used that states if the ratio between the average car pixel velocity and the speed limit is lower than a threshold, the road is classified as congested. This rule can be presented to the decision maker to provide an explanation of how the service chain used the component data to infer the final classification.

C. Discussion

We compared the classifications produced by the congestion reasoner across the test set (733 videos) to the classifications produced by the CNN congestion classifier for imagery at matching location and times. We also compared both these sets of classifications to the respective counts of detected SSOs from the CNN classification. Table IV shows the correlation between (i) the congestion rating (congestion class conditional probability) provided by the CNN congestion classifier, (ii) the count of SSOs detected, and (iii) the congestion rating provided by the reasoner (i.e., \( 1 - \text{traffic flow ratio} \)). All correlations are significant at \( p < 10^{-7} \). Notably, there is a strong correlation between the number of SSOs detected and the congestion rating of the CNN. This shows that the SSO list method does appear to offer a useful explanation for the CNN’s classification.

These correlations give us confidence that the system shown in Figure 1 provides an integrated approach to traffic congestion detection and explanation. The CNN-based classifier
can be considered as the primary congestion detection method due to its higher accuracy (Table I vs Table II). From this, we can generate statements in the form of ‘This location appears congested/not congested’ dependant on the classifier’s output. Taking the congestion reasoner’s output as a secondary method, we can add to this statement a clause that is either complementary (‘This location appears congested and there appears to be a low level of traffic flow’) or in disagreement (‘This location appears congested but there appears to be a high level of traffic flow’). Moreover, the presence/absence of SSOs can help strengthen/weaken the confidence in the system’s final classification and can indicate whether a human user needs to inspect the range of explanations offered.

VI. Conclusion and Future Work

The main contribution of this paper is a systems approach to distributed information fusion integrating heterogeneous services, based on machine learning and reasoning approaches, with an ability to generate explanations of various kinds. Traffic congestion monitoring was chosen as an exemplar application featuring readily-available data of multiple modalities, plausible integration of machine reasoning and learning-based services, and semantic entities at lower and higher levels of abstraction. Through this case study, we have demonstrated how services based on (i) DNN approaches and (ii) heuristic-based reasoning can be combined in a common decision-support task. We examined the distinct kinds of explanation that can be generated for each type of service, including transparency (saliency maps for (i), reasoning traces for (ii)), and post-hoc methods (explanation in terms of similar examples or identification of relevant semantic objects for (i)). We explored trade-offs in terms of the classification performance of the services (in our case study, (i) was more accurate then (ii)) and the kinds of available explanations ((ii) was more transparent than (i)), and showed how service integration offers more robust performance and explainability by demonstrating that correlations between the service outputs can be exploited to generate richer explanations when services ‘agree’, and also when they ‘disagree’.

Moving forward, we plan to expand the services used within the system to enable prediction of future traffic states, together with accompanying explanations for temporal predictions. We also aim to expand the range of data sources to include weather, textual traffic reports and events feeds, for use both in prediction and explanation.

<table>
<thead>
<tr>
<th></th>
<th>SSO count</th>
<th>1 - flow ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Congestion rating</td>
<td>0.61</td>
<td>0.41</td>
</tr>
<tr>
<td>SSO count</td>
<td>—</td>
<td>0.20</td>
</tr>
</tbody>
</table>

TABLE IV
SPEARMAN CORRELATION BETWEEN (I) CNN CONGESTION RATING, (II) COUNT OF DETECTED SSOs AND (III) 1 - TRAFFIC FLOW RATIO; ALL CORRELATIONS SIGNIFICANT AT $p < 10^{-7}$

ACKNOWLEDGMENT

This research was sponsored by the U.S. Army Research Laboratory and the UK Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the UK Ministry of Defence or the UK Government. The U.S. and UK Governments are authorised to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon. We thank Supriyo Chakraborty and the ICPRAI reviewers for helpful comments on drafts of this paper.

REFERENCES

A New Approach to Train Convolutional Neural Networks for Monocular Visual Odometry

Mahdi Abolfazli Esfahani, Keyu Wu, Shenghai Yuan, Han Wang, Senior Member, IEEE
Department of Electrical and Electronics Engineering
Nanyang Technological University
Singapore
\{mahdi001, wukeyu, syuan003, hw\}@ntu.edu.sg

Abstract—The most important part of any autonomous vehicle system is the localization module that gives robot knowledge about its position and orientation relative to its initial state. This knowledge assists robot to move to the location of its desired goal and complete its task. The relative position and orientation information is also the input of other high-level modules to create the map of the scene. Visual Odometry (VO), measures the change of position and orientation as frame changes. Extracting and tracking distinctive sparse or dense features in consecutive frames is the most significant part of VO modules and failing in this process results in failure in VO. Deep Learning, nowadays, help to learn excellent features for the problem of Visual Odometry and estimating camera frame-to-frame movement. Deep Learning based VO methods try to learn an end-to-end network and solve VO as a regression problem and directly output translation and rotation values of camera movement without looking and discussing the label of training data in the training procedure. In this paper, a new approach to train Convolutional Neural Networks (CNNs) for the problem of VO is introduced which first changes the problem to a classification problem and then converts it to a regression one. This approach helps CNN to solve regression problem in a local domain learned in the classification part and do regression in a better manner to learn distinctive features.

Index Terms—Visual Odometry, Deep Learning, Convolutional Neural Network (CNN), Simultaneous Localization and Mapping (SLAM)

I. INTRODUCTION

Localization, mapping and planning are the most important modules of any autonomous robot. Each of these modules require different types of input and output various types of parameters. Visual Odometry (VO) as the base of all these modules output information about frame-to-frame camera movement, and lots of researches has been done in this field in the past few years [1], [2]. Input of this module can be individually or cooperatively stereo or monocular images from visual sensor, depth information from sensing sensors, Inertial Measurement Unit (IMU) sensor measurements etc. VO methods are different according to the type of input, for instance methods based on stereo camera or depth sensors mostly try to use information from structure of the scene while methods based on monocular images are based on the image intensity change.

In one category, VO methods can be classified according to their visual sensor to Stereo and Monocular VO methods, and in another way they can be classified into conventional and learning based VO methods. In the Stereo methods, depth map can be extracted using geometric methods and using this high-value information and converting it into point cloud it is possible to estimate camera movement accurately [3]. [4] also shows that minimizing pixel intensity errors and the error between projected 3D points increases the accuracy. Monocular VO only have information about intensity of each pixel. Methods based on single camera, mostly try to extract distinctive sparse or dense features from two consecutive frames, track them and estimate the camera movement according to the change of such features. Another difference between stereo and monocular VO methods is that monocular VO lacks in estimating the scale of movement and such methods can not estimate scale accurately. Recent methods suggest to use

![Flowchart of Conventional and Learning based Methods](Image)
Inertial Measurement Unit (IMU) as another source of input to estimate scale of camera movement [1], [2].

Figure 1 shows conventional and learning based methods in one diagram. Conventional methods are based on extracting features which can be structure of the scene in stereo images and corner or edges in monocular ones. Hence, the performance of such methods is high in places with existence of high number of feature points and they may lose tracking in challenging low-texture environments. Conventional methods also require an accurate camera calibration to give a robust, reliable and stable output. The main requirement of VO methods is reliable feature points that can be detected and matched with the change of environment and illumination. Convolutional Neural Networks (CNNs) these days help to learn different robust features for various number of applications [5], [6]. Focus of researchers in the area of robotics is also moved to this topic in the past two years when the first work published in this area in 2015 [22]. The only limitation of such machine learning techniques is the requirement of training data which should be similar to the testing environment to get an accurate result. It is because of the learning procedure, which learns feature points according to the input training images appearance. Hence, if environment changes too much, the training model needs to fine-tune by providing some training data.

Recent introduced deep learning based VO methods train CNN as a regressor end-to-end without looking at the label of data. This paper introduces a novel approach to train CNN for VO purpose. In this approach, VO which is a regression problem converts into a classification one which is an easier problem to solve with CNN. After solving this classification problem, the learned model fine-tunes to do the regression. In this manner, network learns to do the regression in a local area rather than doing it globally. To come up with the proposed method, next section gives a review about the previous VO deep learning based methods. Then, structure of our network to solve VO which is based on preprocessed optical flow input images is introduced and experiments and results comes to conclude the overall performance of the introduced method.

II. RELATED WORKS

VO methods can be classified to geometry and learning based methods. In the first category, to estimate camera movement, feature information is extracted and described while in the second category, learning based VO, a model that can fit input information is learned. Figure 1 shows different behavior of Conventional and Learning based methods.

Methods in the first category [8], [9] apply multi-view geometry [7] and extract, track and match features in successive frames to estimate camera movement. Illumination change, noise, moving obstacles etc. cause drift in feature based VO methods. To solve drift problem, OKVIS [2] and VINS [10] save features and descriptors of key-frames to match in next frames. IMU and Loop closure also help to solve scale problem [2], [10]–[12]. Methods based on extracting features, perform poor in texture-less areas. Direct methods, as another kind of geometric methods, estimate camera movement based on all pixels [13]–[16]. Since such methods use whole pixels in their estimation, they can perform well in low-texture environments.

Learning based methods, use machine learning techniques to learn a model that gets images as input and outputs camera movement parameters. This model trains using training data which in VO is captured images and their corresponding label (six parameters showing displacement of camera movement and orientation change). Recent methods [17]–[19] achieved best results using different techniques such as K Nearest Neighbor (KNN), Support Vector Regression (SVR), Convolutional Neural Networks (CNN) etc. Convolutional Neural Networks (CNNs) have shown their powerful performance in the past few years for both regression and classification problems. CNN features are also getting popular in the problem of camera relocalization [20], [21]. In the area of VO, Konda and Memisevic [22] introduced the first work in learning based Stereo VO. They considered VO as a discretized problem and estimated camera direction and velocity change. Optical Flow images which describe the flow and movement of each pixel used by Costante et al. [23] as input and a regression network trained using such Optical Flow images. Moreover, Recurrent Neural Network (RNN) which can learn sequence of features used by Wang et al. [24] by passing raw RGB images as input. Clark et al. [25] also used RNN and raw RGB images but they also considered IMU information to solve scale and drift problems.

III. PROPOSED METHOD

A. Overview

This section introduces a novel approach to train and learn an accurate CNN model for the regression problem of VO. The input of VO CNN is Optical Flow of two consecutive input frames and the outputs are three values showing camera movement displacement and three other values as displacement of camera orientation for two consecutive frames.

To train such a deep network for VO, in our approach, the problem first changes to the classification problem and then translates to a regression one which helps network to learn better in a local domain. In the next section, structure of Optical Flow CNN is given and afterward the training procedure is introduced.

B. CNN Structure

Since Optical Flow images demonstrate motion in two consecutive frames, it is feasible to have a network with low number of layers with such pre-processed inputs. Borex algorithm [27] minimizes following equation to achieve Optical Flow between two frames. In this equation, \( I \) and \( \Phi \) are rectangular image sequence and concave function respectively. 
\[ w := (u, v, 1)^T, \alpha \text{ and } \gamma \text{ are weights.} \]
\[ V(u, v) = V_d + \alpha V_s \]
\[ V_d(u, v) = \int_\Omega \Phi(|I(x+w) - I(x)|^2 + \gamma |\nabla I(x+w) - \nabla I(x)|^2) dx \]
The network used to train Optical Flow based VO is a part of Costante et al. [23] proposed network. In this network, called P-CNN and shown in Figure 3, input is 8 times down sampled of entire optical flow image, and down sampled local parts after dividing OF into four sub-images for 4 times, Figure 2. In Figure 2, blue lines are the input of the Convolutional Neural Network module which is shown on Figure 3. In this manner, both local and global information of OF image extracts and learns to estimate VO.

In the network structure, after two convolution layers, features of each of these convolution layers concatenate together and forward to the fully-connected layer for regression or classification. Results [23] show that different features are learned by two convolution layers and concatenating those features together improves overall accuracy of VO estimation. This is due to the way features are learned in which the first convolution layer learns simple features while the second one learns more complex features using outcome of the first layer. The final output of this network is going to be six parameters showing displacement of camera movement and orientation change.

C. Training Procedure

The main application of CNN is for classification problems where certain number of classes with high number of input for each class exist. Having bounded output space and high number of input per class helps the network to learn better. Regression problems like VO, require CNN to map input space to the large floating point output space of camera translation and rotation parameters. Previous works on learning based VO directly learn a CNN based regression system by giving raw or pre-processed images as input and getting regressed output parameters showing camera movement. Due to the diverse output space, the regression model cannot efficiently learn features and the best weights.

This paper presents a two-phase approach to train and achieve an efficient regression model. In this method, first, a classifier CNN trains using label of the classes achieved by clustering label of data points using Linkage clustering. Linkage clustering [37] is a type of hierarchical clustering method that clusters similar data, according to a distance metric, into same classes. This method hierarchically clusters data according to the similarity and dissimilarity of data points.

An important policy to train CNN as classifier is that the number of data points in different classes should be near to each other. To come up with this issue, in this paper, we visualize number of observations in each class after applying linkage, and similar classes with low number of observations merge to reach a flat histogram of number of data observation per class; an example comes in experiments and results section.

After clustering data points to a certain number of classes with the policy of having flat histogram of data points over different classes, CNN trains as a classifier. Trained classifier fine-tunes as a regression system by replacing softmax with Euclidean loss to regress feature space to certain floating point output labels. The advantage of this approach is that the model first learns the domain that input data is related to, and after that by training the regression model each of the inputs will regress to the local output domain. It is like looking to the input data locally in a smaller output domain.

IV. EXPERIMENTS AND RESULTS

In this section the proposed method analyzes on the KITTI benchmark [38] which is used for different types of Computer Vision problems. Provided images in this dataset are captured by Pointgrey Flea2 stereo cameras. Provided dataset is 10 frames per second undistorted images with the resolution of 1240×386. 10 sequences of this dataset are provided with ground truth which is used in the experiments. Same as [23], this paper considers first 7 sequences as training set and the rest 3 for testing.

Figure 4 and 5 demonstrate the distribution of training
data in 3D world coordinate for displacement of camera movement and camera orientation Euler angles respectively. It visualizes that labels are more diverge in the $z$ direction of Euler parameters and applying Linkage clustering method also proves it and clusters data points into classes related to their $z$ direction of Euler parameters.

Figure 6 shows the histogram of number of data points in each class after applying linkage to cluster data points into seven classes. It is obvious that the histogram is not flat and some classes like class 2, 6 and 7 have low number of observations. Looking at the data and their label in 3D coordinate, it seems that it is possible to merge classes 2, 6 and 7 with other classes to make the histogram flat. Merging observations in these classes with others results in figure 7 which has a flat histogram of data points over classes. Figure 4 and 5 also show final result of clustering data points, and points with same color are in the same class. It is obvious that data is clustered according to the $z$ value of Euler angles which contains large range of values and is the most divergent parameter over all.

After clustering data points and computing Optical Flow according to the method mentioned in section III, CNN trains as a classifier. Afterward, regression network trains by fine-tuning trained network for classification purpose. Hence, during fine-tuning, network uses prior-knowledge of classifying data in the most diverge direction and so network learns to regress data in a local boundary and problem becomes more simple for CNN to learn. It is also important to note that Euclidian loss is used during training network for the regression purpose.

Figure 8-11 show the average of rotation and translation
error across test sequences of KITTI dataset path and speed. These figures show that, using proposed method to train convolutional neural networks outputs a network with less value of error on the test data compare to training P-CNN as regressor from beginning. It also shows that the average error is reduced for one percent in average for both translation and rotation. Hence, clustering label of training data and training a classifier and afterward fine-tuning the learned model as regressor in the problem of Visual Odometry leads to a better result on the test dataset. This improvement is because of learning a local area using a classifier network. Fine-tuning such network as regressor leads to learn output floating points in local areas. Figures 12-14 also compares trajectories computed from trained P-CNN network using proposed method and training it as regressor from beginning on the test data of KITTI dataset.

V. CONCLUSION

In this paper, a new approach to train CNNs for regression problems specially VO is introduced. In this method, regression problem first formulates as a classification problem according to the label of training data and after training classifier network, regression problem dissolve by fine-tuning classification network. Results show that re-formulating problem as a classification one helps regression problems to first learn a classifier in the direction of the most diverge parameter. Fine-tuning such network helps the regressor to learn weights in a local boundary of the divergent parameter learned in the classification problem. Results show that this approach helps learning procedure of the regression problems and improves their accuracy. In the future, to improve the accuracy of monocular VO, effect of IMU information in addition to camera information investigates to solve drift problem in the monocular VO using the proposed approach.

REFERENCES


Segmentation-Free Cell Phenotype Classification using Deep Residual Neural Networks

1st Qicheng Lao  
Department of Computer Science and Software Engineering  
Concordia University  
Montréal, Canada  
qi_lao@encs.concordia.ca

2nd Haoran Sun  
Department of Computer Science and Software Engineering  
Concordia University  
Montréal, Canada  
s_haora@encs.concordia.ca

3rd Thomas Fevens  
Department of Computer Science and Software Engineering  
Concordia University  
Montréal, Canada  
fevens@encs.concordia.ca

Abstract—Cell phenotype classification can be used to characterize complex cell states associated with chemical compound treatment, therefore has great potential in drug discovery. Previous work on image-based cell phenotype classification required a routine yet cumbersome step of single cell segmentation before the classification task. In this paper, we present a segmentation-free method for image-based cell phenotype classification using deep residual neural networks (ResNets). The cell images are samples treated with annotated compounds that can be mainly grouped into three clusters, giving three classes to be classified. Instead of single-cell phenotype classification, we use the raw images without segmentation for our training and evaluation directly. Compared to previous reference work, we significantly simplify the data preprocessing steps for our training and evaluation directly. Our trained ResNets achieve a 98.2% accuracy rate based on five-fold cross-validation.

Index Terms—deep learning, residual neural network, cell phenotype classification, high-content screening, cell profiling assay

I. INTRODUCTION

Cell phenotypes are complexes of morphological features that are present in cell microscopy images. Multiplex cytological profiling assay provides a way to capture a wide range of cell phenotypes by painting the cells in multiple channels with as many fluorescent labels as possible [1]. This image-based cell profiling has great potential in applications such as compound characterization and future drug discovery. However, with the huge amount of image data acquired and high dimensional features to represent each image, it still remains a big challenge for computer vision algorithms to process the massive image dataset and perform cell phenotype classification with high accuracy but still within an appropriate computation time.

With recent emerging algorithms and technologies in deep learning, object recognition and image classification have made impressive breakthroughs for various applications. Convolutional Neural Network (CNN) is one of the deep learning approaches specialized in image-based classification tasks, which has been widely demonstrated to outperform traditional state of the art machine learning algorithms. CNN is a special type of multi-layer neural network that contains from several up to hundreds of convolutional layers inside the network. It was first introduced by LeCun as LeNet for handwritten digit recognition, and the purpose is to recognize visual patterns directly from images with minimal preprocessing [2]. Indeed, one of the biggest advantages of CNN, compared to other traditional approaches, is that for CNN, all features used for classification are automatically learned by the system itself. Thus there is no need to extract predefined hand-crafted features, which is a quite difficult and time-consuming step for most practical problems.

While LeNet started the era of convolutional neural networks, AlexNet [3] is the first popular CNN model which was used on a large-scale dataset of natural images for general purpose classification tasks. Compared to five-layer LeNet, AlexNet has eight layers, thus is slightly deeper, and besides that, it also uses new techniques such as Dropout, Rectified Linear Unit (ReLU) for the nonlinearity functions, and data augmentation techniques to improve the performance. After AlexNet, the CNN models tend to be designed deeper in order to achieve even better performance, among which VGGNet and GoogleLeNet are examples.

Despite their superior performance, deep CNN models are not easy to train due to gradient vanishing and other optimization problems. To solve the accuracy degradation problem with deep models, residual neural network (ResNet) has been developed by He et al. [4], where the network contains residual units parallel to normal convolutional layers. And more recently, new designs of the residual unit have been proposed to improve the generalization thus giving improved accuracy [5].

A. Previous Related Work

The first CNN application for cell phenotype classification in high content screening was reported in 2016 by Dürr and Sick [6], where the authors segmented individual cells...
from the raw images and then used an AlexNet based CNN model to classify single cells into different phenotypes (see Figure 1(a)). Note that while the authors only focused on single-cell phenotype classification, an applicable high-content screening system would still require a final voting algorithm in order to determine the whole-image cell phenotype. The performance of CNN has also been compared to various traditional methods, and it has been shown that not only CNN could save time and costs because of its feature self-learning ability, but also achieve the best performance with overall accuracy 93.4% for four classes classification (three compound clusters plus a mock class). It is also mentioned in the paper that for three classes classification (three compound clusters only), CNN gives a performance of 97.3%.

To the best of our knowledge, it is also the only CNN application to this problem so far. However, the authors only worked on single-cell phenotype classification but did not consider CNN performance on whole-image cell phenotype classification.

B. Our Motivation and Contribution

The previous work by Dürr and Sick [6] requires a crucial step of single cell segmentation before the classification task, which can take lots of time and resources. In addition, although the previous method gives quite good performance, we argue that there are still at least three potential problems:

- First, single cell phenotype may sometimes be mislabeled. This is because the cells are highly diverse in response to compound treatment, therefore, compound treatment may induce changes in certain cells but not necessary all the cells in the same plate. For example, a cell plate treated with fenbendazole for a certain period of time does not necessarily mean that all of the single cells segmented from that plate will render fenbendazole-like phenotype. Considering the high variability in biological systems, it is possible that certain cells may not respond or be resistant to fenbendazole, thus the actual phenotype of those cells remains the same as the mock class, rather than fenbendazole-like phenotype.

- Second, some compound induced phenotypes may only be reflected in the interactions of cells, e.g. cell aggregation, in which case single-cell phenotype based method will fail to work.

- Third, the treatment of compounds that are lethal to cells can result in massive cell death, making single cell segmentation impossible.

With the above concerns in mind, we propose here a segmentation-free method for image-based cell phenotype classification. As is shown in Figure 1(b), we use the raw images with multiple cells as our samples directly for the training and evaluation. Compared to previous work, we make the following two contributions:

1) Less data preprocessing is required. Instead of using segmented images of individual cell as data samples, our approach uses the raw image with multiple cells directly for the training and evaluation. As a result, the time-consuming segmentation step can be saved.

2) For the first time, we give CNN performance on whole-image cell phenotype classification with high accuracy rate of 93.8% for four classes classification and 98.2% for three classes classification.

II. METHODS

In this section, we first introduce the dataset we use for cell phenotype classification, and then the details of data pre-processing steps are shown together with data augmentation. Finally, we describe the architectures of the two CNN models (AlexNet and ResNet) used in this paper.

A. BBBC022v1 Dataset

We use image set BBBC022v1 [1], available from the Broad Bioimage Benchmark Collection [7]. The image set provides images of cultured U2OS cells (Human Bone Osteosarcoma Epithelial Cell Line) treated with 1600 known bioactive chemical compounds for a certain period of time. The images were acquired in five channels, with each channel representing one or two cell components respectively: Hoechst 33342 (Nucleus), Concanavalin A (Endoplasmic Reticulum),
SYTO 14 (Nucleoli), WGA + Phalloidin (Golgi and Actin) and MitoTracker Deep Red (Mitochondria). Different chemical compound treatment can induce a specific pattern changes in cell morphology, which can be captured in multiple channels as cell phenotypes, therefore, this image set provides a basis for testing cell phenotype classification methods with respect to their ability to discriminate a wide range of complex cell phenotypes. More details about the cell profiling assay and compound induced cell phenotypes can be found here [1]. As will be explained in Section II-C below, only a part of the dataset is used for this paper. Figure 2 shows typical five-channel image samples for different compound clusters that will be described in Section II-B.

B. Data Label

Compounds can be mainly clustered into three classes based on their mechanism-of-action and previous study on image-based cell profile similarity using hierarchical clustering [1]. We use the same clustering strategy for our data labels in this paper. More specifically, Cluster A contains three compounds, fenbendazole, oxibendazole and paclitaxel, which represents tubulin modulators; Cluster B contains five neuronal receptor modulators, including fluphenazine, fluphenazine dihydrochloride, metoclopramide, metoclopramide monohydrochloride and procaine hydrochloride; and Cluster C contains five compounds that are structurally related cardenolide glycosides, which are lanatoside C, peruvoside, digitoxin, nerifolin and digoxin. Table I is the summary of compound clusters.

C. Data Preprocessing and Augmentation

To select cell images that can be used for this paper, we search through the whole BBBC022v1 dataset. For each five-channel image sample, we first check its associated compound

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Compounds</th>
<th>Sample size</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Fenbendazole, Oxibendazole, Paclitaxel</td>
<td>108</td>
</tr>
<tr>
<td>B</td>
<td>Fluphenazine, Fluphenazine Dihydrochloride, Metoclopramide, Metoclopramide Monohydrochloride, Procaine Hydrochloride</td>
<td>180</td>
</tr>
<tr>
<td>C</td>
<td>Lanatoside C, Peruvoside, Digitoxin, Nerifolin, Digoxin</td>
<td>216</td>
</tr>
<tr>
<td>Mock</td>
<td>No compound treatment</td>
<td>1215</td>
</tr>
</tbody>
</table>

Table I: Chemical Compound Clusters
name in the metadata file that is provided together with the dataset, if and only if the compound is found to be included in Cluster A, Cluster B or Cluster C, we retrieve the corresponding five channel-separated images from the dataset. We then resize the images with original resolution \((520\times696)\) to \((100\times100)\). After this data clean, we are able to have 2520 cell images in total, corresponding to 504 five-channel image samples, with 108 in Cluster A, 180 in Cluster B and 216 in Cluster C. The retrieved dataset is randomly split into five folds for further cross-validation experiments. During each run of cross-validation, one fold is chosen for testing (20%), and the rest of them (80%) are further split into two parts: training (60%) and validation (20%) set to train and select the model for evaluation on testing set. For four classes classification, we include mock samples in addition to the other three clusters during the selection, and repeat the same above data preprocessing steps to obtain the image data for training, validation and testing, where the images are resized to a slightly higher resolution \((200\times200)\) to keep more subtle details, as the classification based on lower resolution \((100\times100)\) does not give good performance during our initial trial. There are in total 1215 five-channel image samples selected for the mock class.

Unlike the approach used by Dürr and Sick [6], we do not perform single cell segmentation in our data preprocessing, rather, we consider the whole raw images, without segmentation, as our samples directly. This improvement saves a great deal of time and work for both training and evaluation.

It is well known that deep networks typically require a large amount of training samples to achieve satisfactory performance. Due to the limited number of image samples we have for each of our three compound clusters, real-time augmentation is also performed on our training dataset during the training phase. We augment our training dataset by 1) random rotation from \(-90^\circ\) to \(90^\circ\), 2) random translation within \(\pm20\) pixels both horizontally and vertically, 3) random flip both horizontally and vertically. The randomly augmented image samples are used to train our convolutional neural networks without overfitting the models.

D. AlexNet

In order to classify the image samples, we start with a simple AlexNet model, which consists of five convolutional layers and three fully connected layers. Each convolutional layer is followed by a ReLU activation layer, and a \(2\times2\) max-pooling layer is added to the end of the first, second and fifth convolutional layers. Two batch normalization layers are added to the first and second max-pooling layers respectively. We change kernel depth to \(32, 64, 128, 128,\) and \(64\) respectively for the five convolutional layers, while the kernel size is kept the same as the original architecture [3]. The convolutional layers are then followed by three fully connected layers with \(256, 256\) and \(3\) nodes and finally activated by a softmax layer as the output. For four classes classification (three compound clusters plus a mock class), 4 nodes are used instead for the last fully connected layer.

E. ResNet

The introduction of residual connections into convolutional neural networks makes ResNets currently the state of the art for CNN models [4]. The same as in our previous work [8], we also train an 18-layer ResNet model (ResNet-18) integrated with the latest design of residual unit, which is proposed to make the model easier to train and also has better performance [5]. The basic residual unit consists of six sequential components: Batch Normalization, ReLU, Convolution, Batch Normalization, ReLU and Convolution. The identity shortcut is used when the input and output dimensions are the same, otherwise, we consider the projection shortcut to match the dimension using a convolutional layer. Figure 3(a) and 3(b) show the two types of residual units used in this paper, with identity shortcut and projection shortcut respectively. The overall architecture is shown in Figure 3(c).

III. Experiments and Results

Both AlexNet and ResNet are implemented using Keras, a deep learning library written in python with either TensorFlow or Theano as a backend [9]. We use Theano as the backend in this paper. Our AlexNet has about 10.7 million trainable parameters while ResNet has about 11.1 million. To optimize the weights, we use stochastic gradient descent for both AlexNet and ResNet, with a batch size of 100 to compute the gradients using back propagation. The initial learning rate is set to 0.001, decay by 1e-6 over each update, and Nesterov momentum is set to 0.9. To speed up the training, we also monitor on validation accuracies and use the early stopping strategy. All experiments are done on 12 Intel Core i7-6850K
processors with a NVIDIA GeForce GTX 1080/PCIe/SSE2 GPU with CUDA 8.0 installed in a Ubuntu 16.04 LTS.

We first start with the three classes classification problem (three compound clusters only) to get an initial idea about the performance of our proposed segmentation-free method. Figure 4 shows a representative of the ResNet training curves, with accuracy (left) and loss (right) on both training and validation sets. The accuracy is simply the proportion of samples that are classified correctly, and the loss function used is categorical cross-entropy. As we can see from the figure, the validation accuracy converges after around 40 epochs.

In order to compare the performance of AlexNet and ResNet, five-fold cross-validation is performed in the experiments. The overall accuracy is computed by averaging the accuracies evaluated on the testing sets in five runs. Table II shows the accuracies from five-fold cross-validation experiment on both AlexNet and ResNet-18. ResNet-18 has an overall accuracy of 98.2%, which outperforms AlexNet at 91.3%. In addition, our proposed segmentation-free method using ResNet gives a comparable result on whole-image cell phenotype classification with three compound clusters, compared to what has been reported by Dürr and Sick [6], where a CNN model achieved a performance of 97.3% for single-cell phenotype classification, also with three compound clusters.

We also investigate the misclassified samples by our trained ResNet-18, and summarize them as confusion matrices in Figure 5. In total, 9 out of 504 samples are misclassified. It is noticed that almost all of the mis-classifications are related to Cluster A, which is within our expectation since we have much fewer samples in Cluster A than Cluster B and Cluster C. It’s possible that our trained model has not learned enough features to represent phenotypes that are induced by Cluster A compounds.

Next, we further evaluate our segmentation-free method on the four classes classification problem (three compound clusters plus a mock class), where we also compare performance, efficiency and applicability of our methods with previous method [6] in details. Table III shows the performance comparison and Table IV shows the efficiency and applicability comparison. Overall, our method with ResNet-18 gives a comparable performance to previous method [6] while significantly improves the efficiency and applicability.

![Fig. 4. Representative training curves of ResNet-18 for 100 epoches (Three classes classification). Left: training and validation accuracy; Right: training and validation loss.](image)

![Fig. 5. The confusion matrices of ResNet-18 model from five-fold cross-validation. (Three classes classification)](image)

**TABLE II**

<table>
<thead>
<tr>
<th></th>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
<th>#5</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dürr and Sick [6]*</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>97.2%</td>
</tr>
<tr>
<td>Ours with AlexNet</td>
<td>92.1%</td>
<td>94.1%</td>
<td>89.1%</td>
<td>91.1%</td>
<td>90.0%</td>
<td>91.3%</td>
</tr>
<tr>
<td>Ours with ResNet-18</td>
<td>99.0%</td>
<td>99.0%</td>
<td>100%</td>
<td>98.0%</td>
<td>95.0%</td>
<td>98.2%</td>
</tr>
</tbody>
</table>

*AlexNet based model.
TABLE III
COMPARISON OF PERFORMANCE AMONG DIFFERENT METHODS BASED ON FIVE-FOLD CROSS-VALIDATION (FOUR CLASSES CLASSIFICATION).

<table>
<thead>
<tr>
<th></th>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
<th>#5</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dür and Sick [6]a</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>93.4%</td>
</tr>
<tr>
<td>Ours with AlexNet</td>
<td>86.6%</td>
<td>82.3%</td>
<td>87.2%</td>
<td>93.3%</td>
<td>82.2%</td>
<td>86.3%</td>
</tr>
<tr>
<td>Ours with ResNet-18</td>
<td>95.6%</td>
<td>94.8%</td>
<td>92.2%</td>
<td>94.5%</td>
<td>92.1%</td>
<td>93.8%</td>
</tr>
</tbody>
</table>

aAlexNet based model.

TABLE IV
COMPARISON OF PERFORMANCE, EFFICIENCY AND APPLICABILITY AMONG DIFFERENT METHODS (FOUR CLASSES CLASSIFICATION).

<table>
<thead>
<tr>
<th>Methods</th>
<th>Parameter number</th>
<th>Accuracy</th>
<th>Preprocessing and training timea</th>
<th>Segmentation required</th>
<th>Voting algorithm required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dür and Sick [6]b</td>
<td>~1.3 million</td>
<td>93.4%</td>
<td>segmentation time + 3.75h</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ours with AlexNet</td>
<td>~10.7 million</td>
<td>86.3%</td>
<td>2.83h</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Ours with ResNet-18</td>
<td>~11.1 million</td>
<td>93.8%</td>
<td>0.83h</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

aTime calculated assuming 100 epoches in the training phase.
bAlexNet based model.

The total modeling time required for the previous method [6] is the time for data preprocessing (mostly for segmentation) plus 3.75 hours for the training, assuming the epoch number is set to 100 (Although the authors [6] used 500 epoches.). As is stated in Table IV, not only our methods save time in segmentation, but also in the training, as we have much less whole-image training samples as compared to single-cell images. Our method with ResNet-18 is able to give an accuracy of 93.8% with 0.83 hours training time.

In terms of applicability, our methods does not require an overhead step in segmentation in order for the data to be fit with the trained classifier, and no further voting step or other similar algorithm is needed during the prediction, which is required in previous methods [6] to merge the results from multiple single-cell phenotypes.

IV. CONCLUSION

In this paper, we present a segmentation-free method for whole-image cell phenotype classification using both AlexNet and ResNet. Our results show that the latter has better performance. Compared to previous work [6] that uses single cell segmentation before training and evaluation, our pipeline is much more efficient since less data preprocessing is required. Moreover, our implemented ResNet-18 model also gives a comparable performance with accuracy rate of 93.8% on four classes classification (three compound clusters plus a mock class) and 98.2% on three classes classification (three compound clusters only) on BBBC022v1 dataset.

The three compound clusters used in this paper and previous work [1, 6] are known clusters that can lead to different phenotypes. However, BBBC022v1 dataset still contains a large number of chemical compounds that are not part of the above three clusters.

Future work can be focus on identifying new compound clusters in an unsupervised learning manner and then re-validation with supervised learning methods including deep residual neural networks. Moreover, to further increase the performance, more complex learning models can be explored.

REFERENCES

Hybrid Deep Neural Network for Visual Phrase Detection

Lin Bai†, Lina Yang†‡, Yuanyan Tang§, Lin Huo§, Taoshen Li∗†
∗School of Computer, Electronics and Information
Guangxi University, Nanning, China
Email: {bailin, lnyang, tshi, lhuo}@gxu.edu.cn, yytang@umac.mo
†Guangxi Colleges and Universities Key Laboratory of Parallel and Distributed Computing, Nanning, China
§Science and Technology, University of Macau, China
‡Corresponding author
†‡China-ASEAN Research Institute of Guangxi University, Nanning, China

Abstract—Detecting visual phrase in cluttered scenes accurately are challenging problems in computer vision. In this paper, we introduce a hybrid deep learning model to detect and recognize the visual phrase involved in an image. A key contribution of our work is modeling the object-level spatial arrangements of images to aid the learning of high-level relational visual features by using the proposed Factored Conditional Restricted Boltzmann Machine (FCRBM). In this work, we use the deep convolution neural network to learn the object-level representation of an image, which can precisely describe the scene and objects of the query image. These object-level features are further fed into the FCRBM to learn the high-level relational features between objects and scenes. Instead of the traditional deep learning model without or with bias-based condition, the three-way multiplicative interaction structure of FCRBM makes sure that the spatial context can precisely facilitate the learning of the relational features. The top classification RBM achieve the mapping from the relational features to the visual phrase structure label. Compared with the state-of-the-arts, our model achieves competitive visual phrase detection on two known datasets.

Index Terms—Hybrid deep learning, Visual phrase detection, Image processing, Spatial context analysis

I. INTRODUCTION

Automatically detecting and recognizing the visual phrase of an image is a very challenging task in computer vision, and it not only has to detect the most interesting objects, but also needs to describe how these objects relate to each other. It has many applications, for instance, image retrieval, image caption generation, and navigation for the blind. Many previous methods treat it as a general image classification task and use the whole image to represent this visual composite [1]. However, these methods can not discover the significant components of an interaction, such as human, the objects, and the operation that are closely related to the interaction.

Some pioneering methods address the challenges of visual recognition by modeling the mutual context between human poses and objects [2], [3]. However, these approaches are prone to problems caused by inaccurate human pose detection and false object prediction. To alleviate these problems, some approaches try to rely on hand-curated annotations of human poses and objects. It is too high cost and labor-intensive to achieve in applications.

Thanks to the rapid development of deep learning in computer vision, we propose a hybrid deep learning model to address both the tasks of object detection and human-object interaction recognition automatically. Our model utilizes an image I and the spatial context D as input, and is trained to maximize the log likelihood \( p(V|I, D) \) of producing a target visual phrase \( V \). To alleviate the shortcomings of deep learning, requiring a lot of high-quality supervised training data, we systematically consider the image features, the component objects (including human), and the image spatial context.

The major inspiration of our work comes from recent breakthrough researches reported by several machine learning research groups in image-to-text generation [4], [5]. In particular, the researches by [6], they all use Recurrent Neural Networks (RNNs) to capture the image caption by maximizing the log probability of image caption, \( p(S|O) \), given the ground-truth object detection results \( O \) and the model parameters. However, they show results only on the narrow domains with limited objects. Also, they lack the ability of predicting the interaction activity that human and objects involved in.

In this work, we try to take a step towards the goal of predicting the visual phrase of an image, which could detect the most interesting objects, but also precisely describe the interaction activity involved in the query image. We propose to follow the above-mention elegant recipe, by designing a hybrid deep neural network, to achieve the maximized log likelihood \( p(V|I, D) \). Our proposed model has several important advantages over existing image caption generation work. First, it takes deep convolution neural networks (CNNs) to learn the high-level image features, and achieve the object detection and recognition. Over the last few years, it has been convincingly proven that CNNs can learn an global invariant representation of an image by embedding it to a fixed-length vector, such features can be used for a variety of vision tasks [7]. Second, our model proposes a undirected graph to abstractly explore the spatial context of an image, which produce a rich representation of the spatial co-occurrence arrangement. Third, we use a special deep learning model, Factored Conditional Restricted Boltzmann Machine (FCRBM), proposed by us in the work of [1], to learn the visual phrase of the query image.
by progressively integrating object features and the image spatial context.

In the rest of the paper, we first introduce the related work of our model in Section 2. Next we present the hybrid deep learning model, as well as model inference and learning in Section 3. Finally, we show experimental results in Section 4, followed by conclusions.

II. RELATED WORK

The problem of visual composite recognition has been studied for a few year. However, most of works focus on one human and one object interaction activity. Desai et al. demonstrate that discriminative model can learn the spatial arrangements of various objects by abstractly formalizing the spatial context [8]. In the work of [8], Ramanan et al. build a complex system to recognize human-object interaction by taking the advantages of Desai researches. Li’s group demonstrate that image context, such as human poses, spatial layout between human and object, can improve human action prediction [3]. The main inspiration of spatial context is the work by [9]. They encode the spatial context between objects by a set of discrete spatial relationships, and then learn a model for the human-object interaction in each of these relationship bins. We differ from them as we comprehensively consider image spatial context by a novel set of discrete spatial relationships, and embed it into a fixed-length vector. These properties make our model to be more widely applicable.

Many recent works attempt to treat the problem of visual phrase recognition as image-caption retrieval task, and formulate this issue as a ranking learning problem [10]. Firstly, they learn the image features as well as the corresponding sentence features, then train an embedding model that maps both of the image features and the sentence features into a image-sentence feature space. In this feature space, the model can show the most relevant sentence by calculating the distance between sentences and images. Recently, Li and Karpathy’ groups respectively demonstrate that object-level image features can produce better image caption results than global-level image features [11]. Hence, we naturally learn the object-level image features to facilitate the visual phrase recognition by deep CNNs.

Using deep learning technologies to aid image understanding is receiving increasing attention recently. Girshick, Simonyan, and Zisserman show that the deep CNNs can achieve the state-of-the-art performance in many image classification and image feature learning tasks [2]. In the works of [2], [11], they show that it can learn the high-level semantic representation from the still image by combining the CNNs and the Recurrent Neural network (RNN). Some researches learn a probability density from the feature spaces of image and text by deep learning model. These methods can generate richer and more flexible image caption than the traditional generative model. The reason might be that the probability of multi-feature space can serve as the affinity metric for image description prediction. We take the advantages of these methods, and propose a special deep neural network, Factored Conditional Restricted Boltzmann Machine (FCRBM), to learn the high-level relational features between objects (including human). Instead of co-embedding of image and text into a feature space, our model precisely focus on the object-level representation and the spatial context. It has been convincingly shown that using these specific image context features can significantly improve the high-level image reasoning [12].

In this work, we combine deep CNNs for object-level image representation learning with FCRBM nets for high-level relational visual features extracting, to create a hybrid deep network that predict the visual phrases of images. Compared with the other approaches with global level image features [8], our deep CNNs can learn an increasingly complex and invariant object features in the deep hierarchical network, which directly reason the component elements of visual phrase. Instead of limiting to the generalized and global image context, our model precisely take the spatial context as the directly conditional inputs, which makes it possible for the deep networks to directly maximize the log likelihood of the correct visual phrase, \( p(V|I,D) \), given the object-level image features and the corresponding spatial context representation.

III. A HYBRID DEEP NEURAL NETWORK

In this paper, we propose a hybrid deep neural network to detect the visual phrase of images. Figure 1 show the diagram of our hybrid deep model, which is a cascade of deep CNN groups and a FCRBM network. Recent researches in image caption generation have demonstrated that, given the variety of object-level image features, it is possible to improve the image captioning results by directly maximizing the log probability of the image caption in a deep method. Thus, following this elegant recipe, we convert the problem of visual phrase prediction into the task of log likelihood maximization, that directly maximize the log likelihood of the correct visual phrase given the object-level image features and the spatial context between objects, \( p(V|I,D) \), by using the following formulation:

$$
\theta^* = \arg \max_\theta \sum_{(V,I)} \log p(V|I,D; \theta) \tag{1}
$$

where \( \theta \) are the parameters of our hybrid deep model, \( I \) are the object-level features of an image, \( D \) represents the object-level spatial context of the image \( I \). Thus, the well-trained model \( \theta^* \) can predict the correct structure label of visual phrase \( V^* \). At the training phrase, \( V \) and \( (I,D) \) are a training example pair. We optimize the log likelihood as represented in Equation [1] by the proposed special deep neural network, FCRBM. Compared with the other deep networks converting the context information as biases, our FCRBM can directly take the spatial context \( D \) as condition input to facilitate the visual phrase prediction.

For the high-level object representation of images, we use deep convolutional neural networks (deep CNNs). The CNNs have been widely and successfully used to a variety of image recognition tasks, and proved to be the state-of-the-art methods for image object-level features learning.
matrix objects. Hence, the corresponding adjacency matrix is a vector that represents the spatial relationship between a pair of each edge forward propagating a 227 bounding box region. The object-level features are learned by accurately extract a 4096-dimensional feature vector from each where CNN boxes) and the whole image to learn the object-level image representation, we use the top 20 detected object locations (bounding boxes) and the whole image to learn the object-level image representation by using following Equations:

\[
 r_i = W_i [CNNC(O_i)] + b_i
\]

where $CNNC$ is a region-based CNN model, which can accurately extract a 4096-dimensional feature vector from each bounding box region. The object-level features are learned by forward propagating a $227 \times 227$ RGB bounding-box image block through five convolutional and sub-sample layers, as well as two fully-connected layers. The parameters $W_i$ is a transformation matrix, having $k \times 4096$ dimensions, which transforms the object-level feature vectors into a fixed-length invariant representation. The $k$ is the size of the fixed-length multimodal embedding space. Thus, every image would be represented as a set of $k$-dimensional vectors: $\{v_i | i = 1 \cdots 20\}$.

B. Spatial Context learning with Graph-based Model

In order to describe the spatial context, we introduce a graph-based model, $G = (V, E)$, which explicitly describe an image as a collection of $M$ objects (Including human), and the spatial context between objects. The $V$ is the vertex set and each vertex $v_i \in V$ represents an object. $E$ is the edge set and each edge $e_{ij} \in E$ is an ordered pair with a fixed-length score vector that represents the spatial relationship between a pair of objects. Hence, the corresponding adjacency matrix is a $M \times M$ matrix $\tilde{A} = (\tilde{e}_{ij})_{M \times M}$, where $\tilde{e}_{ij}$ represents the score vector of the arc that show the spatial relationship between object $v_i$ and $v_j$. $\hat{e}_{ij}$ bins the spatial relationship between $v_i$ and $v_j$ into one of canonical discrete relations including above, overlapping, below, next-to, top-left, lower-left, top-right, lower-right and far. Obviously, $\hat{e}_{ij}$ is a binary vector of length 9 with a couple of 1 that represents these corresponding relations is satisfied this pair of objects $v_i$ and $v_j$. Figure 2 shows an example of capturing the spatial context of an image by using our graph-based model.

C. Visual Phrase Prediction with FCRBM

To achieve the log likelihood, $p(V|I,D;\theta)$, we introduce a novel deep learning model, Factored Conditional Restricted Boltzmann Machine (FCRBM), which casts the problem as a visual composite structured prediction task. Rather than retrieving a binary label for image block, our model can precisely address the interaction relationship between human and objects by modeling the mapping from the spatial relationship between objects to the interaction activity.

The FCRBM utilizes factored three-way interactions to allow real-valued style features to control the input and hidden pairwise interactions. The FCRBM defines a joint probability distribution, $P(I,H|D)$, over the visible layer, $I$, and the hidden layer, $H$, conditional on the spatial context features, $D$. In order to maximize $P(I,H|D)$. We train the hidden layer by using the object-level image features learning from the deep CNNs and the spatial context features from the proposed graph-based model in the CD-K algorithm. The hidden layer connects to the visual phrase prediction layer, a classification RBM, which indicate the probability distribution over a variety of

Fig. 1. Diagram of our hybrid deep neural network. It uses the deep CNNs for object-level image features and learns the features of the spatial context by using the proposed graph-based model. And then the FCRBM reasons the visual phrase by learning the relationship between objects with the help of the spatial context features directly.

Fig. 2. A visualization of graph-based model. It considers 9 different spatial layout to describe object-level spatial arrangement by using an ordered graph model, and then utilizes a CNNs to learn the invariant and high-level spatial features.
visual phrase structure; that is, which visual phrase the image belong to. The FCRBM structure is shown in Figure 1.

The FCRBM energy function captures all possible correlations among the image feature units, the spatial context units, and the relational feature units, which is defined as:

\[
E(v, h | z) = -\sum_{i, j, k} z_i v_j h_k w_{ij}^v w_{jk}^h - \sum_k w_k^h \sum_j w_j^v v_j 
\]

where \(f\) indexes a set of factors, the number is \(F\). \(w_{ij}^v\) models the connection between factors and conditional units, \(w_{jk}^h\) models the connection between input units and factors, \(w_k^h\) models the connection between factors and hidden units, \(v_j\) represents the state of the input unit \(j\); \(h_k\) is the state of the hidden unit \(k\); \(z_i\) represents the state of the conditional unit \(i\). The \(\sum w_k^h h_k\) and \(\sum w_j^v v_j\) are the bias terms used to model the base rate of activity of the hidden and input units.

The classification RBM that models the joint distribution between the visual phrase structural label (the output), the relational features (The input layer), and the RBM-hidden layer, as \(p(y, x, h) \propto e^{-E(y, x, h)}\), where \(E(y, x, h) = -h^T W_x - h^T U_y - b^T x - c^T h - d^T y\). Hence, given the relational features, the conditional probability of the output \(y\) can be represented as:

\[
p(y | x) = \frac{e^{d c}}{\sum e^{d c}} \prod_x (1 + e^{d c} + \sum_k e^{d c} w_{jk} y_k) 
\]

where \(c\) indexes the \(c\)-th visual phrase class. We train the FCRBM under supervised learning method with CD-K algorithm, which is introduced by Hinton et al. [14]. It minimizes the negative log likelihood log \(p(y | x)\) of the target visual phrase class \(t\), given the high-level relational features between objects \(x\).

IV. EXPERIMENTS

To evaluate the performance of our hybrid deep neural network, the experiments are conducted on two known datasets: (1) UIUC phrase dataset [8]: (2) a six-class sports dataset [3]. We show extensive experimental results about visual phrase detection and recognition.

A. Data Source and Representation

UIUC phrase dataset is a special subset of the PASCAL dataset. It contains 2769 images labeled with 8 PASCAL categories. In addition, every image has hand-curated bounding box annotation for the involved visual phrase. There are 17 visual phrase classes in this dataset, which is 120 images per visual phrase. 12 of these visual phrase describe interaction activities between two objects (e.g. person riding bicycle), the other 5 categories depict one object performing an action. Since the goal of our model is to detect and recognize the interaction phrase between more than one object, the experiment focuses on the 12 visual phrases.

Six-class sports dataset collects thousands of images that contains bounding box annotations for a manually generated list of 6 sport visual phrase classes. In this dataset, a lot of visual phrases contains more than two objects. Hence, instead of limiting to the interaction between one human and object, our model can consider all of involved objects, such as considering cricket bat, ball, stump, player in “cricket batting” and “cricket bowling”; considering tennis racket, ball, and player in “tennis forehand” and “tennis serving”. There are more than 50 images in each visual phrase category. We use the same setting as in [3]: 40 images for training and 10 for testing.

B. Visual Phrase Detection

We focus our experimental results for visual phrase detection. We compare our hybrid deep learning model with three state-of-the-art models: Choi model [15], Mutual Model [3], and Group of Objects [16] on the above mentioned datasets, respectively. We evaluate the results by averaging statistic items, including per class Average Precision (AP) and overall AP, for all these four methods. We learn the three state-of-the-art methods by using the publicly available code on both of the dataset.

Tables 1 and 2 show the results of the visual phrase detection and recognition by our model and the other three models on these two datasets, respectively. From these two tables, it is noted that our model outperforms all the state-of-the-art methods in most visual phrases across the two datasets. Moreover, on the six-sport dataset, compared with the overall AP of 0.894, the best results from the three models, Our model gets an overall AP of 0.863. Especially, our model achieve the best performance for all of the six visual phrase detection, even showing a 5 percent average improvement over the Choi model where the ground-truth interaction activity labels are used. We believe the reason may be that our model explores every spatial arrangement detail between different objects, which can facilitate the interaction activity involved in the visual phrase.

For the tasks of UIUC phrase dataset, our model also achieve the best AP for the most visual phrase in 12 visual phrase classes. Even though Choi model and Mutual model achieve the good performance as well as our model in a couple of tasks, such as “Dog lying on sofa”, our model still obtains the best overall AP. Compared with the other methods without or with limit bias-based image context, We believe that the outstanding detection results attribute to the special structure of FCRBM, which directly takes the spatial context features as conditional information to significantly facilitate the relational features learning.

Furthermore, instead of limiting to the interaction activity between one human and one object, our model can deal with the situations where the number of human and object can be arbitrary. For example, people interacts with a tennis racket and a tennis ball in “playing tennis”. So that our model would have more wider application prospect.

V. CONCLUSION

In this paper, we introduce a novel hybrid deep learning model for visual phrase detection. Our model utilizes deep
Our model | Choi model | Mutual model | Group of Objects
---|---|---|---
Person next to bicycle | 76.5 | 62.3 | 66.5 | 56.3
Horse and rider jumping | 92.5 | 87.3 | 86.0 | 75.3
Person drinking from bottle | 41.6 | 36.5 | 41.5 | 30.3
Person sitting on sofa | 30.8 | 28.0 | 26.5 | 24.8
Person riding horse | 80.6 | 77.5 | 78.5 | 77.3
Person riding bicycle | 75.5 | 70.5 | 68.6 | 66.9
Dog lying on sofa | 25.5 | 24.8 | 24.7 | 25.5
Bicycle next to car | 59.6 | 52.3 | 43.2 | 49.6
Person next to horse | 53.6 | 47.5 | 41.8 | 43.8

**Table I**

Per-class AP scores for visual phrase detection on UIUC visual phrase dataset. We compare our method with three state-of-the-art models.

---

Table II

<table>
<thead>
<tr>
<th>Visual phrase</th>
<th>Our model (AP%)</th>
<th>Choi model (AP%)</th>
<th>Mutual model (AP%)</th>
<th>Group of Objects (AP%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cricket batting</td>
<td>92.5</td>
<td>92.0</td>
<td>88.5</td>
<td>84.3</td>
</tr>
<tr>
<td>Cricket bowling</td>
<td>81.5</td>
<td>67.3</td>
<td>79.0</td>
<td>70.3</td>
</tr>
<tr>
<td>Croquet shot</td>
<td>93.2</td>
<td>92.8</td>
<td>92.5</td>
<td>85.3</td>
</tr>
<tr>
<td>Tennis forehand</td>
<td>84.8</td>
<td>78.0</td>
<td>80.5</td>
<td>76.8</td>
</tr>
<tr>
<td>Tennis serving</td>
<td>98.6</td>
<td>79.5</td>
<td>84.5</td>
<td>74.3</td>
</tr>
<tr>
<td>Volleyball smash</td>
<td>95.5</td>
<td>89.5</td>
<td>92.6</td>
<td>84.9</td>
</tr>
</tbody>
</table>

The report worked in this paper was partially supported by projects of Natural Science Foundation of Guangxi 2016GXNSFBA380163 and 2017GXNSFBA198226, one project of Middle Age and Youth Teachers Basic Ability Promotion of Guangxi 2017KY0033, and projects of Higher Education Undergraduate Teaching Reform Project of Guangxi 2017JGA124 and 2017JGB108.

**References**


Abstract—Recently, Deep Neural Networks (DNNs) have dramatically improved performance in computer vision, speech recognition and natural language processing. However, training of the DNNs requires a large amount of computational resources. As a result, the power consumption of the training servers continues to increase in proportion to the expansion of the DNN size. Therefore, we need a method of reducing the power consumption required for DNN training. In this paper, we propose a method of training DNNs in which we adopt a reduced bitwidth representation to improve energy efficiency of arithmetic operations and to reduce the required memory size. Our quantization method extracts the characteristics of the DNN parameters as statistical distributions in parallel with arithmetic operations of the DNN training. We tap into the collected statistical distributions to determine the shared exponents. We applied our method to the training of LeNet, VGG8, AlexNet and long short term memory (LSTM) and found that the accuracy of our method is equivalent to that of the 32-bit floating-point method.

Index Terms—shared exponents, quantization, floating-point, fixed-point, deep learning, CNN, LSTM

I. INTRODUCTION

The advancement in deep learning has substantially pushed the boundaries in computer vision, speech recognition, and natural language processing. This progress depends in large part on graphics processing units (GPUs) which require a vast amount of computational resources. However, with the increase in the application areas of deep learning and the neural network size [1], designing new deep learning algorithms or developing new application areas for deep learning is often limited by computational capability of GPUs. One of these limitations comes from the large power consumption of GPUs because the amount of computational resources is limited by the predefined power budget of boards or servers. Therefore, in order to extend the range of application of deep learning, we need energy-efficient hardware to accelerate DNN training.

We propose a quantization method of training DNNs called deep-learning integer (DL-INT), in which we adopt a reduced bitwidth representation to improve the energy efficiency of the arithmetic operations and to reduce the required memory size. Our method can be applied to a dedicated hardware for deep learning to reduce power consumption and the required memory size. We will first discuss some training methods in which low precision arithmetic operations are applied. We will then present a brief summary of our method and give the details. Finally, we then discuss the experimental results of comparing our method to a 32-bit counterpart.

II. RELATED WORK AND SUMMARIES

The quantization methods of DNNs can be categorized into two types: quantization for only inference and quantization for both training and inference. The methods of the former type convert the pre-trained models into quantized DNNs [2]–[6] and some of methods retrain the quantized DNNs to improve the test accuracy. However, as mentioned above, limitations of computational resources hinder the development of new algorithms. Therefore, a quantization method including a training phase is more useful.

The methods to train quantized DNNs from scratch can be divided into two categories: those with a high-precision copy of the weights and those without. Some of the methods of the former category use binary networks [7]–[16]. However, their accuracy is still not sufficient. Some other methods with which data are quantized into 16 bits, 8 bits, or less have also been proposed [17]–[24]. Although these methods can reduce the cost of arithmetic operations due to quantization, most have a high-precision copy of weights for updating. As a result, they cannot reduce the required memory size to train large size DNNs.

Courbariaux et al. proposed a quantization method for training DNNs from scratch without a high-precision copy of weights [25]. They adopted the overflow rate to adjust the shared exponent of fixed-point data. Since their method does not require high-precision weights, it improves energy efficiency of arithmetic operations and reduces the required
memory size. However, they applied their method only to a Maxout network with small-size datasets, such as MNIST [26], CIFAR-10 [27], and SVHN [28], and did not conduct a comprehensive evaluation of their method. Therefore, there is a need for a training method with sufficient evaluation that would improve energy efficiency of arithmetic operations and reduce the required memory size.

A. The Brief Summary of the Proposed Method:

- We adopted the shared exponents and the fixed-point representation with reduced bitwidth. In our method, each tensor of each layer such as weight and bias has a single shared exponent in contrast to floating-point representation in which all of the data have individual exponents.
- Our method collects statistical distributions of each tensor of each layer. We updated shared exponents from the collected statistical distributions in every mini-batch training.
- We applied our method to convolution layers and fully connected layers which occupy more than 90% of all operations on DNNs [29]. All the tensors of convolution layers and fully connected layers were quantized. That is, we quantized tensors of weight (W) operation, activation (A), gradient (G) operation, and updating (U).
- We can reduce the memory size necessary for all data and parameters since our method exploits the reduced bitwidth and shared exponents and quantizes not only W, A, and G but also U.
- Our method can reduce power consumption during the DNN training by adopting fixed-point operations. Typically, integer units consume less energy compared to floating-point units [30].

B. The Advantages of Our Method:

- We can reduce the memory size necessary for all data and parameters since our method exploits the reduced bitwidth and shared exponents and quantizes not only W, A, and G but also U.
- Our method can reduce power consumption during the DNN training by adopting fixed-point operations. Typically, integer units consume less energy compared to floating-point units [30].

III. TRAINING QUANTIZED NETS IN LOW PRECISION

In this section, we describe the procedure of our quantization method. First, we show the details of floating and fixed-point formats, and we estimate the cost of the memory and multipliers. Secondly, we present the outline of our method. Lastly, we show the details of our quantization procedure.

A. Floating and Fixed-Point formats

Floating-point formats are often used to represent real values. They consist of a sign, exponent, and mantissa. The 32-bit floating-point format (FP32) has an 8-bit exponent and a 23-bit mantissa, and the 16-bit floating-point format (FP16) has a 5-bit exponent and a 10-bit mantissa.

Fixed-point formats consist of a sign, shared exponent, and mantissa. For example, a 16-bit fixed-point format has a 8-bit shared exponent and 15-bit mantissa.

B. Cost of Multiplier and Memory

DNN training requires a large amount of memory resources. If we use a 16-bit fixed-point format instead of a 32-bit floating-point format, it is clear that the memory size can be reduced by approximately 1/2, as illustrated in Fig. 1.
The shared exponent should be appropriately updated to achieve high training accuracy. We use statistical distributions of the operation results, such as convolution and dot product, to update the shared exponents. Therefore, we need a sufficient amount of samples.

D. Details of Our Quantized Operation

In this section, we describe the step 1, 2, 3 of Alg. 1. Our DL-INT method uses the fixed-point format and has an accumulator with 16-bits + 16-bits + N guard bits. The guard bits are, for example, 8-bits. This accumulator prevents the saturation and rounding of the arithmetic operation results.

Alg. 2 shows the details of DL-INT computation. $x$ and $w$ are input tensors and $y$ is an output tensor. Each tensor has its own shared exponent, $e_x, e_w, e_y$. These are examples of the shared exponent illustrated in Fig. 1. The array $s$ in Alg. 2 is the statistical distributions of the accumulator.

First, we operate the convolution or dot product, and we put the result into $acc$ in Alg. 2. Secondly, we search the first bit position from the most significant bit without a sign bit. We count the bit’s position. That is, we increment the $i$th index of statistical distributions shown in Alg. 2. Lastly, we reduce $acc$ into $y$ using function $SatRnd()$ that we introduce as follows.

\[
Sat(x, b, e) = \text{Max}(\text{Min}(x, 2^{b+e-1} - 2^e), -2^{b+e-1}),
\]

\[
ep = e_x + e_w - e_y,
\]

\[
SatRnd(x, e_x, e_w, e_y) = Sat(SocRnd(\frac{x}{2^{ep}}), 15, e_y),
\]

where $b$ is the bitwidth of the fixed-point number, $e$ is the shared exponent value, and $2^e$ is the quantization step size. $Sat()$ is saturation with a positive maximum and negative minimum value of the fixed-point number. $SocRnd()$ is stochastic rounding to round the fractional part of shifted $acc$. We use $SatRnd()$ after searching index $i$ of array $s$ as shown in Alg. 2.

E. Method of Updating Shared Exponents

In this section, we describe the step 4 of Alg. 1. We present the algorithm for updating the shared exponent in Alg. 3 We use this algorithm in each mini-batch as shown in Alg. 1. The array $s$ is the statistical distribution which has been collected in Alg. 2.

First, we describe the case where the width of the effective statistical distributions is larger than 15 digits that is the word length of a 16-bit fixed-point format excluding a sign bit. We illustrate an example of the statistical distribution of array $s$ in Fig. 3.

To avoid the malfunction of the outliers, we exclude the outliers from the distribution. Hence, we introduce the maximum threshold. We calculate this value from the total elements of each tensor $\times 0.01\%$. We search the largest value without the outliers using this threshold, and choose the next shared exponent. $acc$’s shared exponent is $e_x + e_w$. $s[e_x + e_w]$ is $2^{38}$. Therefore, the real value of $acc$ is mantissa of $acc + 2^{38}$. For example, the real exponent value of the element of the tensor whose selected index is 37 is $2^{38}$. In Alg. 3, we calculate the next shared exponent by using array $s$. We illustrate this method by using the Fig. 2.
Therefore, the next is the word length of a and output, and the precision of the arithmetic operation is neural network has arithmetic operations with tensors as input and output, and the precision of the arithmetic operation is FP32. Our method is simulated by quantizing components of the tensor that is the input of the arithmetic operations to the precision of a fixed-point number. The tensors are quantized by the bitwidth and shared exponent value of a fixed-point number as shown below:

\[
Quant(x, b, e) = Sat(StocRnd(\frac{x}{2^e}) \times 2^e, b, e),
\]

where \(b, e, Sat() \) and \(StocRnd()\) is same as described in section III-D. In \(Quant()\), the procedure of quantizing the floating-point number is shown as dividing by the quantization step size, rounding stochastically, and multiplying by the quantization step size. \(Quant()\) also collect the statistical distribution from the floating-point number in parallel with the quantization. After the quantization, the arithmetic operations with FP32 are implemented with CUDA [32] to accelerate matrix operations.

The pseudo code of our implementation is shown in Alg. 4. Note that a subscript of tensor \(q\) means quantized tensor, and \(f()\) is an arithmetic operation such as convolution and dot product with FP32. We evaluated FP16 for comparison by type conversion to FP16 instead of \(Quant()\), and the precision of an arithmetic operation is FP32.

\begin{algorithm}
\caption{Strategy to update the shared exponent.}
\begin{algorithmic}
\Require shared exponent of input data \(e_x\)
\Require shared exponent of weight \(e_w\)
\Require statistical distribution \(s\)
\Require the width of effective statistical data \(w\)
\Require max threshold \(t_{\text{max}}\)
\Ensure next shared exponent of output data \(e_y\)
\If {width \geq (16 - 1)}
\State \(sum = 0\)
\For {\(i = 38\ldots0\)}
\State \(sum = sum + s[i]\)
\If {sum \geq t_{\text{max}}}
\State next \(e_y = i + (e_x + e_w) - (15 - 1)\), break
\EndIf
\EndFor
\Else
\For {\(i = 38\ldots0\)}
\If {\(s[i] \neq 0\)}
\State next \(e_y = i + ((15 - \text{width})/2) - (e_x + e_w)\), break
\EndIf
\EndFor
\EndIf
\end{algorithmic}
\end{algorithm}

The \(P\) in Fig. 2 shows the selected \(i\) in Alg. 3. That is, the real exponent of the element that has index \(P\) is \(P + e_x + e_w\). Therefore, the next \(e_y\) is \(P + (e_x + e_w) - (15 - 1)\), where 15 is the word length of a 16-bit fixed-point format excluding a sign bit. 15 - 1 is the distance of the most significant bit and the least significant bit.

Secondly, we have to mention the case where the width of the effective statistical distributions is narrower than 15 digits. First, we search \(P\), and chose the center of the statistical distributions.

\section{Pre-training for the Initial Shared Exponent Value}

Our method uses previous shared exponents and statistical distributions. Thus, we need to choose initial shared exponent (ISE) values from the other way. If we use the random values as the initial shared exponent values, the training of DNNs does not converge in many cases. We carry out the mini-batch pre-training by using a floating-point format. The number of mini-batch pre-training are, for example, 100 iterations. After the pre-training, we determine the initial shared exponents according to the following equation.

\[
ISE = \text{ceil}(\log(max(max(t),(-1 \times \text{min}(t)))), (4)
\]

where \(t\) is tensors of the pre-training results.

\section{IV. EXPERIMENT}

\subsection{A. Implementation of DL-INT}

We implemented software for deep learning to evaluate the DL-INT proposed in section III. Our implementation of the neural network has tensors in each layer, and the data type of each component in the tensor is FP32. Each layer of the neural network has arithmetic operations with tensors as input and output, and the precision of the arithmetic operation is
The quantized tensor in the layer is illustrated in Fig. 4 as a computational graph. At the forward propagation, input data, weight, bias, and output data are quantized as illustrated in Fig. 4 (a). At the backward propagation, output gradient, input gradient, weight gradient, and bias gradient are quantized as illustrated in Fig. 4 (b). At parameter update, we used stochastic gradient descent as an optimization algorithm. Weight and bias were updated with a quantized tensor as shown below:

$$w_{(n,m)} = w_{(n,m)}^q - \eta \cdot g_{w_{(n,m)}}^q,$$

$$b = b^q - \eta \cdot g_{b}^q,$$

where $\eta$ is a learning rate. $g$ is the gradient of each tensor. Thus, W, A, G, and U are quantized in convolution and fully-connected layers. The parameter of the evaluated neural networks are listed in Table II. The details of each parameter can be found in Caffe [33].

**Parameter of Evaluated Neural Networks.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Network</th>
<th>LeNet</th>
<th>VGG8</th>
<th>AlexNet</th>
<th>LSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch size</td>
<td>64</td>
<td>52</td>
<td>256</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Base learning rate</td>
<td>0.01</td>
<td>0.001</td>
<td>0.01</td>
<td>0.01</td>
<td>1.0</td>
</tr>
<tr>
<td>Learning rate policy</td>
<td>inv</td>
<td>step</td>
<td>step</td>
<td>step</td>
<td></td>
</tr>
<tr>
<td>Step size</td>
<td>10000</td>
<td>10000</td>
<td>10000</td>
<td>1^*</td>
<td></td>
</tr>
<tr>
<td>Gamma</td>
<td>0.0001</td>
<td>0.1</td>
<td>0.1</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Power</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Momentum</td>
<td>0.9</td>
<td>0.9</td>
<td>0.9</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Weight decay</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
<td></td>
</tr>
<tr>
<td>Regularization type</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>-</td>
<td>35</td>
</tr>
</tbody>
</table>

*Unit of step size is epoch at LSTM. Learning rate is fixed at less than 6 epoch.

**TABLE III**

<table>
<thead>
<tr>
<th>Network</th>
<th>Datasets</th>
<th>Top-1 accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>FP32</td>
</tr>
<tr>
<td>LeNet</td>
<td>MNIST</td>
<td>98.81%</td>
</tr>
<tr>
<td>VGG8</td>
<td>ImageNet</td>
<td>38.34%</td>
</tr>
<tr>
<td>AlexNet</td>
<td>ImageNet</td>
<td>36.47%</td>
</tr>
<tr>
<td>LSTM</td>
<td>PTB</td>
<td>92.13%</td>
</tr>
</tbody>
</table>

*Penn Tree Bank. **Criterion of LSTM is perplexity.

**Fig. 5. Test accuracy of each neural network.**

**B. Evaluation of DL-INT**

We compared the accuracy of FP32, FP16, and 16-bit DL-INT (DL-INT16) as shown in Table III. The evaluated neural networks are LeNet [26], VGG8 [34], AlexNet [35] and LSTM [36]–[38]. The criteria of LeNet, VGG8, and AlexNet are top-1 or top-5 accuracy. The criterion of LSTM is perplexity.

About the accuracy of LeNet, DL-INT16 is equivalent to FP32. The perplexity of LSTM with DL-INT16 indicate DL-INT16 is applicable to recurrent neural network. The accuracy of VGG8 with DL-INT16 is slightly better than FP32. Therefore, DL-INT is applicable to the classification of large datasets with many classes, such as ImageNet [39]. The accuracy of AlexNet with DL-INT16 is equivalent to FP32. Therefore, DL-INT16 can be applied to deep neural networks that achieve a high accuracy. Fig. 5 illustrates transitions of the test accuracy of each neural network.
V. CONCLUSION AND FUTURE WORK

In this paper, we presented DL-INT which is the quantization method to improve energy efficiency of arithmetic operations and to reduce the required memory size. In our method, we employed a 16-bit fixed-point format with shared exponents based on the statistical distributions instead of a 32-bit floating-point format. We proposed a method of collecting statistical distributions of all exponents of each tensor in the accumulators and updating shared exponents using them.

We tested experimentally our quantization method by implementing a software of deep learning and evaluated the training of LeNet, VGG8, AlexNet and LSTM. We obtained an equivalent accuracy compared to FP32 by using DL-INT16. We confirmed that our method improves energy efficiency of the arithmetic operations and reduces the required memory size.

Further experiments would be required to show the effectiveness of DL-INT for large networks such as VGG16 [40] and ResNet50 [41]. As future work, we will study 12-bit and 8-bit version of DL-INT, and implement an energy efficiency DNN accelerator.

REFERENCES

[38] Chainer/examples/ptb https://github.com/chainer/chainer/tree/master/examples/ptb
Oral Session D

Hierarchical Image Representation and Vision

Tuesday May 15, 2018, 10:00 AM

Elham Etemad and Qigang Gao

*Hybrid Image Representation Method based on Bag of Edge Tokens from Octaves of Edge Elements*

Michael H.F. Wilkinson and Simon Gazagnes

*Distributed Component Forests: Hierarchical Image Representations Suitable for Tera-Scale Images*

Minh-Tan Pham, Sébastien Lefèvre, Erchan Aptoula and Lorenzo Bruzzone

*Recent Developments from Attribute Profiles for Remote Sensing Image Classification*

Edward Cayllahua, Yukiko Kenmochi, Jean Cousty, Arnaldo de Albuquerque Araujo and Guillermo Cámara Chávez

*Algorithms for hierarchical segmentation based on the Felzenszwalb-Huttenlocher dissimilarity*

Tianatahina Jimmy Francky Randrianasoa, Camille Kurtz, Pierre Gançarski, Eric Desjardin and Nicolas Passat

*Intrinsic Quality Analysis of Binary Partition Trees*
Hybrid Image Representation Method based on Bag of Edge Tokens from Octaves of Edge Elements

Elham Etemad  
Faculty of Computer Science  
Dalhousie University  
Halifax, Canada  
eefami@dal.ca

Qigang Gao  
Faculty of Computer Science  
Dalhousie University  
Halifax, Canada  
qggao@cs.dal.ca

Abstract—Despite the fascinating performance gained by the Convolutional Neural Networks in the essential task of image representation, these methods require a huge amount of training data and massive computation power which may not be available for all applications. These applications usually rely on handcrafted image representation methods. We have combined a global characteristic of the image, i.e. shape, with the Bag of Words methodology mostly utilized for local image representation, and have proposed a hybrid image representation. Despite other local image representation methods, we have considered the fact that human visual perception relies on the object appearance described by its visual descriptive properties, such as color, texture, and shape. In particular, shape structures play a significant role in the human visual system for object recognition. On the other hand, despite other methods considering perceptual structures, we have defined computer generated definitions for the edge tokens. We accordingly propose an image representation method that extracts perceptual structure-based descriptors of an object’s shape from a hierarchy of edge maps of smoothed images, while it scales the image to octaves to find edge elements robust against object size. These edge elements are described using a local descriptor and are utilized to define edge tokens in the image. We have tested our proposed method in the multi-label and multi-class image classification problem. The presented results prove the superiority of the proposed method over other local image representation methods.

Index Terms—perceptual descriptor, Bag of Edge Tokens, hybrid image representation, scale octaves hierarchy

I. INTRODUCTION

Image representation is an essential task in the field of computer vision. Developing more efficient image representation methods can result in improvements for different computer vision tasks such as image classification, content based image retrieval, object recognition, and vision-based Augmented Reality (AR) applications.

Various strategies have been proposed for this task including the following major categories: Global, Local and Deep representations, and hybrids of their combinations. Recently, the deep image representation methods using Convolutional Neural Networks (CNN) have gained a performance close to human [1]–[3]. However, these methods require a huge amount of training data, and a powerful computing device, which are not always available for applications.

Many AR applications are examples of scenarios with limited training data, and computing power. As a result, these applications rely on handcrafted features from global or local image representation methods [4]–[6]. Contrary to Deep learning based methods, these handcrafted features require less amount of training samples, and are more straightforward for applying prior knowledge into them. Besides, their calculation generally requires lower computation power.

Global image representation methods utilize some general image properties, including color, texture, or shape to find a representation of the image [7]–[9]. Since the human visual system mostly relies on shape features, many methods have considered extracting perceptual shape features from the image [9]–[11]. These methods have proposed methods to find generic edge tokens in the image, however, these tokens are defined and selected manually.

The local image representation methods usually find a set of keypoints in the image, and describe the area around them to find local descriptors. All local descriptors obtained from the training data are utilized to create a Bag of Words (BoW). Finally, each image is encoded according to this BoW [12].

Many keypoint detection methods have been proposed to find the more descriptive areas in the image. SIFT [13], SURF [14], ORB [15], and FAST [16] are among the most well-known keypoint detection methods. ORB and FAST assume that the corner points in the image are representative candidates for the content of the image, and rely on Harris Corner Measure [17] for selecting keypoints. On the other hand, SIFT and SURF select local maximums in different scales of the image as keypoints. None of these methods preserve the perceptual structures of object shapes while detecting keypoints, disregarding the significant role of these features in the human visual system for recognizing objects.

For describing the area around each keypoint, different descriptors have been proposed. Descriptors such as SIFT [13], and SURF [14] utilize the intensity gradient around the keypoint to describe that area. Other methods such as BRIEF [18], BRISK [19], ORB [15], and FREAK [20] provide a binary feature vector for describing each keypoint. These methods select points around the keypoint and find the description of the area by comparing their values.

A clustering algorithm such as K-means, or a dictionary learning method can be used on the keypoints’ descriptions of the training set to create the BoW. While various dictionary
learning methods have been proposed, the experiments conducted by [21], [22] prove that the selected dictionary learning method has less significance on the performance of the image representation compared to the selected encoding method.

Two major trends exist for encoding an image based on a dictionary, i.e. hard coding, and soft coding. The hard coding techniques associate each local descriptor to a single word in the BoW, while the soft coding methods associate a descriptor to several words in the BoW. Among many methods proposed for encoding images, Vector Quantization [23] utilizes a hard coding strategy while Sparse Coding [24], Locality Constraint Sparse Coding [25], Group Sparse Coding [26], and Label Constraint Sparse Coding [27] consider soft coding.

To improve the quality of the local image representation methods, we have considered the fact that human visual perception mostly relies on objects’ shapes, and as a result the areas on the objects’ boundaries are more descriptive. On the other hand, to improve the quality of methods based on perceptual features, we have proposed a method to define the edge tokens automatically. Our proposed method extracts perceptual structure-based edge elements from the image edge map, describe the area around them, and cluster those elements to find edge tokens. Each image will be encoded using these edge tokens obtained from the training set. These tokens are utilized by the proposed method to find an encoding for each image.

Our proposed method has been tested on the multi-class multi-label image classification problem and its performance comparison is elaborated in the experimental results section. Our evaluation shows that the proposed method has improved the results in this challenging task by around 2%, while its time complexity is in the same range as the existing methods.

In Section II of this paper, we introduce the details of our proposed method. Our experimental setting, and evaluation results, followed by discussion on the obtained performance, are presented in Section III. Finally, this paper is concluded in Section IV and some areas of future work are introduced.

II. PROPOSED METHOD

The proposed image representation method focuses on the fact that human visual perception mostly relies on the shapes of objects. Since an object’s shape is defined by its boundaries, just considering the corner points as descriptive areas is not...
sufficient to carry structural information. The edge map of the image is a representative for its shape. In our proposed method, we consider octaves of an image to have robustness against object scaling. For each octave we extract perceptual structure-based shape descriptors (edge elements) from a hierarchy of edge maps using various smoothing parameters. We extract edge elements from each of these edge maps and describe their surrounding areas using a local descriptor. Descriptors in the training set are utilized to find a set of edge tokens using K-means clustering algorithm. Each image will be encoded according to these edge tokens.

A. Edge Elements

In the proposed method, we have considered several octaves of images for finding edge elements. Each octave has been created by re-sizing the input image to a specified scale of \( 2^k \). Image scaling results in extracting perceptual elements from various sizes of an object. As a result, more descriptive and general edge tokens are generated, and the image representation is more robust against object scaling.

The image from each octave is processed in a hierarchy of edge maps, each of which is obtained from a smoothed image with a different value for \( \sigma \) and produces different edge elements. The higher the value for \( \sigma \), the smoother the image, and the less noisy and coarser the edge map. This hierarchy creates edge elements with longer and less noisy structures, as well as shorter and noisier structures. This wide range of edge elements can describe objects with very well-defined boundaries and objects with noisy textures.

The Canny Edge Detection method [28] is applied to each smoothed image in each octave to detect its edge map. Given the smoothed image, Canny calculates the intensity gradient of the image, and cancels false detected edges by applying a non-maximum suppression. By applying double thresholding, Canny finds potential edges and tracks those edges to reject the detected elements which are not connected to a strong edge.

Since in our proposed method we are interested in finding longer and less noisy edges whenever possible, we first extract edge elements from the most smoothed image, the highest \( \sigma \) value, using the Canny edge detection method. In the next levels, we reduce the smoothing parameters to produce shorter and noisier edge maps.

This decision creates two difficulties to be handled. First, since a single edge in the image may be extracted as a long straight line in one smoothing level, and as a noisy curved line in another, duplicated edges will be created from different smoothing levels. Second, sometimes a very small contrast between the foreground object and the background makes it impossible to extract edge maps from the original image.

To mitigate the first problem, we must keep the less noisy edge element and ignore all of its duplications. Since we utilize various Gaussian Filter parameters, these duplicated edges are not exactly in the same location in different edge maps, but they are very close to each other. To solve this concern, in each level we have dilated the edge map obtained from the previous level(s) with a \( 3 \times 3 \) kernel to make the detected edges thicker. Dilation is a morphological operation on the image which slides a kernel \( K \) on the binary image \( P \), i.e. the image’s edge map, according to \( P \circ K = \bigcup_{P \in K} K_P \).

Using the cumulative edge map of all previous levels, \( P \), and the detected edge map of the current level, \( D \), the edge map obtained from the current level without their duplications are obtained using the logical operation of \( N = P \cdot D \).

At the end of each smoothing level, the cumulative edge map, \( P \), is updated by adding the newly detected edges to the map using the logical operation of \( P = P + N \). The general diagram of this procedure is represented in Fig. 1.

To address the second problem, we have sharpened the image in the level where we apply a very small \( \sigma \), near to zero, for smoothing the image. The sharpening filter of (1) enhances the image’s contrast and increases the likelihood of detecting edges around the foreground object in the image.

\[
\begin{bmatrix}
-1 & -1 & -1 \\
-1 & 9 & -1 \\
-1 & -1 & -1 \\
\end{bmatrix}
\]  

(1)

After detecting edge maps in each octave and in each smoothing level, we utilize the Hough transform [29] to extract the edge elements from the obtained edge map. The Hough transform finds all instances of a specified shape in the image. Since in our proposed method the local descriptor provides the appropriate description for the area around each edge element, considering complex shapes only increases the complexity of the proposed method without any additional value. As a result, we have utilized the Hough transform for finding the existing lines in the edge map.

B. Edge Tokens and Image Encoding

Edge tokens are defined as classes of common edge elements, or perceptual structure-based shapes obtained from
TABLE II: Per-class performance and the performance for each individual classes of PASCAL VOC 2007 test set. SIFT descriptor has been utilized for all methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Aero</th>
<th>Bike</th>
<th>Bird</th>
<th>Boat</th>
<th>Bottle</th>
<th>Car</th>
<th>Cat</th>
<th>Chair</th>
<th>Cow</th>
<th>Dog</th>
<th>Horse</th>
<th>Mbike</th>
<th>Person</th>
<th>Plant</th>
<th>Sheep</th>
<th>Sofa</th>
<th>Train</th>
<th>TV</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFT</td>
<td>94.5</td>
<td>92.0</td>
<td>90.7</td>
<td>94.5</td>
<td>89.4</td>
<td>93.0</td>
<td>76.6</td>
<td>89.4</td>
<td>82.7</td>
<td>95.2</td>
<td>83.0</td>
<td>85.0</td>
<td>91.7</td>
<td>85.0</td>
<td>91.7</td>
<td>92.2</td>
<td>60.7</td>
<td>89.4</td>
</tr>
<tr>
<td>SURF</td>
<td>95.3</td>
<td>92.4</td>
<td>89.7</td>
<td>94.6</td>
<td>90.0</td>
<td>94.4</td>
<td>78.8</td>
<td>89.3</td>
<td>83.6</td>
<td>94.9</td>
<td>92.1</td>
<td>85.8</td>
<td>91.3</td>
<td>92.9</td>
<td>58.8</td>
<td>90.0</td>
<td>96.7</td>
<td>88.5</td>
</tr>
<tr>
<td>ORB</td>
<td>94.3</td>
<td>91.1</td>
<td>89.4</td>
<td>94.4</td>
<td>89.4</td>
<td>93.7</td>
<td>76.6</td>
<td>88.0</td>
<td>81.2</td>
<td>95.3</td>
<td>91.1</td>
<td>84.7</td>
<td>90.4</td>
<td>92.7</td>
<td>56.2</td>
<td>90.1</td>
<td>97.4</td>
<td>87.5</td>
</tr>
<tr>
<td>FAST</td>
<td>94.6</td>
<td>93.0</td>
<td>90.1</td>
<td>95.5</td>
<td>90.9</td>
<td>94.7</td>
<td>81.4</td>
<td>90.0</td>
<td>84.6</td>
<td>95.9</td>
<td>92.4</td>
<td>86.7</td>
<td>93.0</td>
<td>93.4</td>
<td>64.8</td>
<td>90.6</td>
<td>97.1</td>
<td>88.7</td>
</tr>
<tr>
<td>EDGE</td>
<td>96.1</td>
<td>94.4</td>
<td>91.5</td>
<td>96.3</td>
<td>93.1</td>
<td>95.4</td>
<td>81.8</td>
<td>91.4</td>
<td>86.4</td>
<td>95.6</td>
<td>93.6</td>
<td>88.5</td>
<td>93.8</td>
<td>94.7</td>
<td>65.7</td>
<td>93.2</td>
<td>97.4</td>
<td>90.1</td>
</tr>
</tbody>
</table>

Hough transform, that can describe each image. In the proposed method, we have utilized a local image descriptor such as SIFT [13] to describe the area around each edge element obtained from the previous steps. We have calculated this description by assuming a point in the centre of each edge element, \((L_x, L_y)\), and describing the area whose length, \(A\), is equal to the length of edge element, around that point which is rotated according to the edge element’s angle, \(\theta\), with respect to the image coordinate system. These characteristics are calculated according to (2) for an edge element whose starting and ending points are \((S_x, S_y)\), and \((E_x, E_y)\).

\[
\forall w \in \{x, y\}, \quad L_w = S_w + \frac{E_w - S_w}{2} \\
A = \sqrt{\left(\frac{E_x - S_x}{2}\right)^2 + \left(\frac{E_y - S_y}{2}\right)^2} \\
\theta = \arccos\left(\frac{E_y - S_y}{\left(\frac{E_x - S_x}{2}\right) \times A}\right) \quad (2)
\]

All descriptors from the training set are collected and fed into the K-means clustering algorithm for finding edge tokens describing the current dataset. Each image in the dataset can be encoded using these edge tokens. To encode each image, all edge elements in that image are extracted and mapped to the edge token that has the shortest \(L2-norm\) distance.

III. EXPERIMENTAL RESULTS

We have tested our proposed method on the multi-label and multi-class image classification datasets of PASCAL VOC 2007 test set (4952 images) [30], and VOC 2012 validation set (5823 images) [31]. In our experiments we utilized SIFT [13], and SURF [14] to describe the areas around the edge elements and created 500 edge tokens. We compared our proposed image representation method with other existing methods, such as SIFT [13], SURF [14], ORB [15], and FAST [16], using SIFT [13] and SURF [14] as their descriptors.

We have trained a Multi-Layer Perceptron network with two hidden layers of size 200, on 2501 and 5717 training images of VOC 2007 and 2012, with ADAM solver until it reached a maximum of 5000 iterations, or until its calculated loss in two consecutive epochs did not improve by 0.001. Our octaves resize factor was selected from \(K \in \{1, 0, -1, -2\}\) and the standard deviations for Gaussian filters were \(\sigma = 3, 1, 0.01\).

The performance metrics that are adapted in this work are categorized into two classes of overall performance, and per-class performance. For each of these categories the Accuracy, Precision, Recall, and F-Measure are presented. The equations for calculating the precision metric for these performance classes \((O_p\) and \(P_p\) for overall and per-class precision) are shown in (3) and the same terminology is applied to the other metrics. In these equations, \(K\) is the number of classes in the dataset, \(N_k^c\) is the number of correctly predicted instances, and \(N_k^p\) is the total number of predicted samples.

\[
O_p = \frac{\sum_{k=1}^{K} N_k^c}{\sum_{k=1}^{K} N_k^p} \quad P_p = \frac{1}{K} \sum_{k=1}^{K} \frac{N_k^c}{N_k^p} \quad (3)
\]

The overall performance metrics are represented in Table I where in each column the method with the highest performance is bolded. Comparing these results shows the proposed method provides higher F-Measure for all different cases, where in some cases the accuracy, precision, or recall of the FAST method is slightly higher. These results prove that our proposed method provides a better trade-off between precision and recall compared to the other methods.

The per-class and each individual class performance are represented in Table II and Table III. These results represent...
the superiority of our proposed method against other methods in almost all metrics for two different datasets. Only for the recall on PASCAL VOC 2007 the FAST keypoint detection method shows slightly higher performance, around 0.2%.

The interesting fact that is noticeable among the results in Table II and and Table III is the superiority of the proposed method performance on classes with well-defined shapes, such as human-made objects like aeroplane, boat, bus, and car.

The only classes where some other methods show higher performance are animals, or plants whose shapes have many varieties and are not well-defined. Among these classes, our proposed method performance is very close to the best method except for the classes of sheep and plants.

Figure 2 shows the keypoints detected by different existing methods, and a single level of the proposed method. The first row of this figure represents the weakness of methods such as SIFT and ORB in detecting keypoints where the lighting condition is not appropriate. These methods were not able to extract any keypoint in this image. SURF and FAST suggested keypoints that did not belong to the foreground object. On the other hand, keypoints associated to the edge elements of the proposed method are located on the object’s boundary.

Considering the second row of Figure 2, where the foreground is an airplane on a very small scale, we have noticed that SIFT failed to detect keypoints on the foreground object which resulted in it not being able to describe the image appropriately. On the other hand SURF was confused and detected the column in the image as foreground. ORB, FAST, and EDGE detected keypoints on the airplane, while FAST produced more keypoints in the background.

The third row of Figure 2 shows keypoints extracted from an image with a noisy background and noisy foregrounds. It is noticeable that FAST failed in detecting representative keypoints for foregrounds. SURF and SIFT suffered from the same problem with slightly better performances. ORB produced the closest output to the proposed method, although it also missed one of the foregrounds while detecting keypoints.

IV. CONCLUSION

Image representation has a fundamental rule in the success of computer vision applications. Even though deep learning methods for image representation have shown fascinating performance, there are still applications relying on handcrafted features because of their lack of training data, or computational power. In this work we propose a hybrid image representation method which produces higher performance comparing to existing methods. In the future, we are going to utilize the edge tokens obtained from the proposed method for image matching and object tracking. The combination of these edge tokens with CNN-based methods and applying that method to existing methods. In the future, we are going to utilize the edge tokens of the proposed method for image matching and object tracking. The combination of these edge tokens with CNN-based methods and applying that method to an AR application is another path for the future.

REFERENCES


Fig. 2: Qualitative comparison of the existing keypoint detection methods with the points associated to edge elements of the proposed method (EDGE) on images with low lighting condition, small object, and noisy background with noisy objects.


Distributed Component Forests: Hierarchical Image Representations Suitable for Tera-Scale Images.

Michael H.F. Wilkinson
University of Groningen
Groningen, The Netherlands
m.h.f.wilkinson@rug.nl

Simon Gazagnes
University of Groningen
Groningen, The Netherlands
s.r.n.gazagnes@rug.nl

Abstract—The standard representations known as component trees, used in morphological connected attribute filtering and multi-scale analysis, are unsuitable for cases in which either the image itself, or the tree do not fit in the memory of a single compute node. Recently, a new structure has been developed which consists of a collection of modified component trees, one for each image tile. It has to date only been applied to fairly simple image filtering based on area. In this paper we explore other applications of these distributed component forests, in particular to multi-scale analysis such as pattern spectra, and morphological attribute profiles and multi-scale leveling segmentations.

Index Terms—Mathematical morphology, multi-scale representation, distributed-memory computation

I. INTRODUCTION

As image sizes get larger, so too do hierarchical representations of these images. One class of hierarchical image representation is from connected filtering, referred to as min trees and max trees [1] component trees [2], or opening and closing trees [3]. These tree structures form a compact representation of all the connected components of all threshold sets in an image. They find use in various applications, such as attribute filtering [1], [4]–[6], computation of pattern spectra [7], or morphological profiles [8]–[10], and visualization [11].

The computational complexity of algorithms to construct these tree structures is also modest: either $O(GN)$, with $G$ the number of grey levels in the regular 8-16 bit per pixel case or $(N \log N)$ in the 32-bit or more integer or floating point case [12], [13]. Furthermore, shared-memory parallel algorithms for computation of these trees [14], [15], and subsequent postprocessing have been developed [10], [16]. On fairly modest compute servers they can handle images up to a few gigapixel at most.

This is despite the attractive property of these tree structures, that they form a very memory-efficient multi-scale representation of an image, which can represent all scales present in the image, unlike classical image pyramids, or wavelets, which discretize scale in some way. However, their storage cost is proportional only to the image size, not the number of desired scales, such as in e.g. a Gaussian scale space. A typical component tree can grow to about 20-40 times the size of the image in storage. In practice, in parallel computation we pre-allocate this maximum, both because it can simplify the algorithm, and because it avoids the need for (locking) memory allocation during tree construction [14], [15].

Given that many imaging modalities routinely acquire images in the order of tens or hundreds of gigapixels, and tera-scale images occur in both remote sensing and astronomy, there is a need for a representation capable of handling these hierarchies in a distributed manner. Very recently, such a method has been developed in the form of distributed component forests (DCFs) [17]. The algorithm presented only allowed application of morphological area openings [18], [19]. Here, we explore how to implement multi-scale analysis methods such as pattern spectra and morphological profiles using distributed component forests.

The paper is organized as follows: first we discuss connected attribute filters. We then turn to component trees and how they can be used to implement these methods efficiently. After this, we discuss parallelization strategies in shared and distributed memory. Finally, we discuss two multi-scale analysis tools, i.e., morphological profiles and pattern spectra, and how they can be implemented in this framework.

II. CONNECTED FILTERS AND MULTI-SCALE TOOLS

Connected filters [20] are a type of morphological filter based on manipulation of flat zones, which are connected zones of constant grey level or colour of maximal extent. In the binary and grey-scale cases, a simple, but powerful approach is that of attribute filters [4]. In the binary case, an attribute opening computes some increasing property, or attribute of all connected foreground components, and removes all those for which the attribute falls below some threshold $\lambda$. The simplest case is the area opening [18], [19]. In the grey-scale case, we can in principle compute all threshold sets, apply the binary attribute filter to each threshold set, and stack the results up to obtain a grey-scale result. In the case of the area opening, the result resembles filing off the top of every bright peak in the images until its area is at least $\lambda$ in area. In practice faster methods exist, which will be discussed in Section III.

These filters can be extended to non-increasing attributes, like elongation, and also vector attributes [6], which allow enhancement or detection of a range of structures.
III. COMPONENT TREES AND FORESTS

The trivial implementation of attribute filtering of thresholding the image explicitly and performing connected component analysis at each threshold is very inefficient. However, it is readily seen that the connected foreground components of each threshold set except the lowest are nested in precisely one connected foreground component at a lower level. Therefore, they can be organized into a tree structure, as seen in Figure 1.

We can distinguish two types of component trees: max-trees, as depicted in Figure 1, in which each node represents a foreground component at some threshold set, and min-trees in which the nodes represent background components at each threshold set, and where the nesting relationship is reversed. Min-trees are often computed as max-trees of the inverted image. For the low dynamic range images in which we are interested here, the most efficient algorithm to compute these trees is by flooding from the lowest grey levels in a depth-first manner, using the algorithm of [1]. For a recent review of algorithms see [21].

There are various ways in which we can represent component trees. Here we use the representation from [14]. In this representation we allocate an array of nodes the size of the image, which is the worst-case number of nodes for any image. In practice, the number of component tree nodes is smaller, so multiple pixels will in general belong to the same node. All pixels in the node have parent pointers that point to a single pixel that represents the entire node. This is called the canonical element, or level root. The level root in turn points to a pixel in the parent node in the component tree. Level roots are readily distinguished from all others because they point to a node representing a pixel of a different grey level than their own. The overall root of the tree has a pointer with a special value: ⊥. Any manipulation of the component tree can be performed by appropriate manipulation of the level roots in this representation.

A. Parallel Computation

To perform parallel computation for low-dynamic-range images, we can use the algorithm from [14]. The image is first divided into strips or tiles, one to each processor. After this, local component trees are computed, using a modification of the algorithm of [1], where the labeling of nodes is done in such a way that it allows merger of nodes using union-find [22]. After this, neighbouring tiles or strips are merged hierarchically, and a single component tree of the image is constructed. During the final merge, carried out by the thread with rank zero, the entire component tree and image must be accessible to that thread. Once the final merge has been performed, each thread can proceed to filter the image independently [14], or perform more complex operations like computation of differential attribute profiles [10], or pattern spectra [16].

B. Distributed Component Forests

In the distributed case, we cannot directly use the above strategy, if the size of the component tree is such that it will not fit into the memory of the processor of rank zero. Quite apart from that limitation, communicating entire component trees becomes prohibitive. Therefore a new approach is needed, which was first presented in [17].

As before, we split the image up into tiles, and built local component trees for each tile. We now have a hierarchical representation of \( h \)-connected components of each tile. Pixels \( p \) and \( q \) at grey level \( \geq h \) are \( h \)-connected if there exists a path \( \text{within the tile} \) from \( p \) to \( q \) through pixels of grey level \( \geq h \). We now need to correct these by modifying the connectivity along the boundary, such that the pixels \( p \) and \( q \) are considered \( h \)-connected if there exists a path \( \text{within the entire image} \) from \( p \) to \( q \) through pixels of grey level \( \geq h \). Each node in the modified max-trees then represent \( h \)-connected components intersected with the domain of the tile. Furthermore, the attribute assigned to each node must equal that of the corresponding \( h \)-connected component of the entire image.

To do this, boundary trees consisting of the subsets of nodes that touch the boundary of a tile are communicated to the neighbours. If the number of grey levels \( G \) is modest, the boundary tree will be much smaller than the component tree.
of the entire tile. An upper bound bound of the size is simply $G\#(\delta B)/2$, where $\#(\delta B)$ indicates the cardinality of the boundary $\delta B$ of the tile $B$. For a tile of $20,000^2$ and $G = 256$, this works out as $1.024 \cdot 10^7$ nodes maximally, compared to a worst case component tree size of $\#(B) = 4.0 \cdot 10^8$. Assuming square tiles, the boundary tree size scales as the square root of the tile size, whereas the component tree itself scales linearly with tile size. Boundary trees from two neighbours are merged and combined into a new boundary tree that includes only the level roots of the two merged trees. This means the combined tree stores a smaller number of nodes than the worst-case upper limit which contains the total number of nodes in the two merged trees. The merging process is then repeated with the new combined trees until all merges have been done. Figure 1 shows an example of the merging step for a simple case composed of two local component trees. Then, the changes in the boundary trees need to be propagated to the local component trees in such a way that filtering individual trees in the forest yields exactly the same result as filtering the entire component tree. As the previous approach detailed in [17] was found incomplete in certain cases, we implemented a different procedure that corrects and updates each boundary tree as we go backward through the merging steps. The idea is, starting from the last step of the merging phase, to update successively each boundary tree that was merged until we reach the local boundary trees. The update of two merged trees is done through their combined tree, for each node $x$ to be updated, we look into the updated boundary tree and check if its parent is in the same boundary tree. If yes, the node $x$ is reconnected to this parent, if not, the parent node is added to the boundary tree of $x$. Once the two trees have been updated, they are sent back to the process that performed the merging and will, in turn, be used to update the former boundary trees. To perform this procedure, each process performing a merge need to store the boundary trees that will then be updated. Each process then stores $2 \times n_{co} + 1$ boundary trees, where $n_{co}$ is the number of merges and + 1 stands for the local boundary tree in each tile.

The algorithm presented in [17] only allowed application of area openings [19]. Here, we explore how to implement multi-scale analysis methods such as pattern spectra and morphological profiles using distributed component forests.

### C. Granulometries and Pattern Spectra

Apart from simply filtering an image, multi-scale analysis using component trees is an efficient way of extracting information from images. After all, given that the component tree contains information of all connected components at all grey

---

#### Algorithm 1 Successive merging and correction of boundary trees

```plaintext
procedure mergeBTree (rank, n_co : integer; 
  mtree : Max-Tree; 
  var btree : Boundary-Tree[0..2n_co])
  btree[0] ← create_boundary(mtree);
  nearest ← 1;
  if n_co > 0 then
    for i ← 0 to 2n_co step 2 do
      btree[i + 1] ← getBTree(rank + nearest_rank);
      btree[i + 2] ← mergeBTrees(btree[i], btree[i + 1]);
      nearest_rank ← 2nearest_rank
  end for
  if rank ≠ 0 then
    sendBTree(rank - nearest_rank, btree[i + 2]);
  end if
end procedure

procedure updateBTree (rank, n_co : integer; 
  var btree : Boundary-Tree[0..2n_co])
  nearest_rank ← 2n_co;
  if rank ≠ 0 then
    btree[2n_co] ← recvUpdatedBTree(rank - nearest_rank);
  end if
  for i ← 2n_co downto 1 step 2 do
    btree[i - 2] ← correctBTree(btree[i - 2], btree[i]);
    btree[i - 1] ← correctBTree(btree[i - 1], btree[i]);
    sendCorrected(rank + nearest_rank, btree[i - 1]);
    treeBTree(btree[i]);
    nearest_rank ← nearest_rank/2;
  end for
end procedure
```
levels, and their attributes, any filtered version based on those attributes can be computed from it, along with any statistics on the sizes and shapes of components in the image.

A classical morphological tool for multi-scale analysis are pattern spectra [23], which are based on granulometries. Granulometries are totally ordered sets of openings \( \{ \gamma_r \} \), with \( r \) from some index set \( I \), such that
\[
\gamma_r(f) = \gamma_{\max(r,a)}(f)
\]
(1)
A pattern spectrum is obtained by computing how much of the image content is removed by each consecutive filter in a granulometry. Let \( \gamma_0(f) \) be the original image, a discrete pattern spectrum \( S(f) \) can then be calculated as
\[
S(f)_r = \sum_A (\gamma_{r-1}(f) - \gamma_r(f)) \quad r = 1,2,..\,N
\]
(2)
where \( S(f)_r \) is the \( r \)-th bin of the pattern spectrum, and the sum is taken over the image domain \( A \). In principle, this requires \( N \) openings to be computed, followed by \( N \) differences, and summations over the images.

Computing pattern spectra based on attribute filters can be done far more efficiently, simply by analysing the component tree. We need to visit each node, represented by its level root, in the component tree, compute which bin in the pattern spectrum it belongs to, and add the product of the area of the node and the grey-level difference to its parent to that bin. In the shared-memory parallel case, each process does this with a private copy of the pattern spectrum, only for those level roots in its tile, and these are added together in the final stages [16]. In the distributed case, things are slightly more complicated, because each node in the complete component tree may be represented by multiple level roots of different modified component trees in the DCF. This means that the above process will lead to double counting of component tree nodes that span more than one tile.

The solution is fairly simple: we need to add only that part of the component tree node that intersects the local tile to the private copy of the pattern spectrum. No extra computation is needed, as we compute this area anyway while building local component trees. Before starting the merge process, we simply store the computed area in an extra field \( \text{privateArea} \). During the merge process, this field is unchanged.

At the end of the merge, only original level roots of the component tree of the tile have a non-zero \( \text{privateArea} \). After merge, they may no longer be level roots, but they must be accounted for. Therefore, we scan all nodes within the area of the tile \( T_p \) of process \( p \). For every node \( v \) that has a non-zero \( \text{privateArea} \), we find the level root of \( v \) and determine the appropriate scale with in the pattern spectrum the node belongs to. After this, the product of the \( \text{privateArea} \) field and the difference in grey level with the parent of the level root is added to the appropriate bin. We then check whether this node has been reconnected to an other parent with higher grey level during the update phase. To do so, we use on extra variable \( \text{gval}_\text{par} \) that stores the grey value of the local parent at the end of the initial creation of the local component tree. For each node, we compare the grey value of its parent to the grey value of its initial parent given by \( \text{gval}_\text{par} \). If they are different, we iterate up the tree towards the root, determining the correct bin from the area of the current node, and the grey level difference with its parent. This grey level difference is multiplied by the \( \text{privateArea} \) field of the node we started from, and the result is added to the appropriate bin. When all processes have computed their pattern spectra, we sum the private pattern spectra as before. The algorithm to compute each pattern spectrum for tile \( T_p \) is shown in Algorithm 2.

Algorithm 2 Computation of the area pattern spectrum for each tile, for process \( p \). Function \( \text{Par} \) returns the level root of the parent of the current node.

\begin{verbatim}
procedure MaxTreeAreaSpectrum (Tp : Tile; 
node : Max-Tree; 
var spectrum : integer[numscales]; 
lambda : integer[numscales])

for all v ∈ Tp do 
if node[v].privateArea ≠ 0 then 
scale ← findScale(node[LevelRoot(v)].area); 
privateArea ← node[v].privateArea 
parent ← Par(LevelRoot(v)); 
spectrum[scale] ← spectrum[scale]+ 
(gval[v] - gval[parent]) * privateArea; 

u ← parent; 
while not IsRoot(node[u].parent) 
and node[u].gval ≠ node[v].gval_par do 
scale ← findScale(node[u].area); 
parent ← Par(u); 
spectrum[scale] ← spectrum[scale]+ 
(gval[u] - gval[parent]) * privateArea; 

u ← parent;
end while;
end if;
end for;
end procedure
\end{verbatim}

D. Differential Area Profiles

The Differential Area Profiles (DAP) is a multi-scale representation of an image. DAPs are related to pattern spectra in that they can be considered as a vector image in which each pixel of the original image is replaced by a vector representing a local pattern spectrum at that point. They can be computed as a series of top-hat filtered images, using a series of area openings of increasing area threshold \( \lambda \). This contains all the bright features in the image. To extract the dark features, a series of bottom-hat filters based on area closings is used, and the two patterns spectra are concatenated in a single vector. An example is shown in Figure 3.

In many cases, it is not the DAP we are interested in, but a more accessible, 2D representation of the salient features.
in the DAP, derived from the so-called multi-scale leveling segmentation [9]. For each pixel, we determine the scale $S$ (in terms of area) which has the highest amplitude to represent that pixel. We also record the amplitude $C$ (for contrast) at that scale ($S$), and the original luminance $L$, forming the so-called CSL segmentation of that image, shown in Figure 3(c), with the three bands mapped to R, G, and B channels respectively.

Clearly, as we can compute area openings from a distributed component forest (DCF), it is trivial to use them to compute a series of area openings, and consequently difference the results to obtain the required stack of area top-hat images. Alternatively we can use the same component-tree filtering algorithm used in parallel computation of the CSL segmentation on each of the modified component trees in the DCF. The only modification needed is that we must only store data from the current tile, not the nodes added in the merging process. As each node contains a field indicating which process it belongs to, this is straightforward.

IV. RESULTS

We implemented the distributed memory algorithm based on the previous work of [17] and using the recursive update approach detailed in Section III-B. We then added the extra code needed to compute pattern spectra and CSL segmentation based on DAPs respectively. Timings were performed on the Zeus compute server on a 1.2 Gpixel remote sensing image of Haiti. We measured wall-clock time for 1, 2, 4, ..., 64 processes. In both the pattern spectrum and the CSL representation we used 32 scales. Figure 4 shows the results.

As expected computing the more complex CSL segmentation is a bit more costly than computing a single area opening. Somewhat unexpectedly, computing pattern spectra proved a bit more expensive, probably due to possible multiple visits to nodes not in the current tile by Algorithm 2. Speed-up does follow a similar pattern. In all cases, speed-up is good up to 32 processes, after which the slope of the curve is lowered. This may be due to the machine used, as a similar pattern was observed in [15].
V. CONCLUSIONS

We have shown that the use of distributed component forests for connected morphological multi-scale analysis requires little modification of existing component tree processing methods. This implies that DCFs can be adapted to a range of component-tree-based methods for very large images. Some may of course be more challenging than others. For example, if we want to do clustering-based filtering of component trees [24] considerable inter-process communication may well be needed in the filtering stage as well. Likewise, connected filtering using tree-based shape spaces [25], which build a component tree on the component tree of an image, might also be harder to implement.

In future work we will extend the method to 3-D, which will incur higher overheads, due to the comparatively larger boundary surface between the 3-D tiles. Furthermore, extension to at most 16-bit per pixel images is expected to be possible, given that the merging algorithm used is still efficient in the shared memory case [15], and boundary-tree sizes are still manageable. Moving to floating point or even 32-bit per pixel will be much more difficult, due to the huge potential increase in maximum boundary-tree size. This may require a very different approach. We will also investigate how the multiple visits to nodes in computing the pattern spectrum per tile can be avoided, to increase performance.

A similar approach could be extended to the alpha tree [26], for filtering and segmentation of colour and hyper-spectral images, which has applications in information mining in very large images [27]. The self-dual Tree-of-Shapes [28] might also allow a distributed forest variant.

More tests are needed on larger images, and on bigger machines to assess the full potential of these methods, but already they offer the only working method for distributed computation of several connected filters and related operators. The code is available upon request.

ACKNOWLEDGMENT

The authors would like to thank Jan Kazemier for his initial version of the distributed-memory code.

REFERENCES


Recent Developments from Attribute Profiles for Remote Sensing Image Classification

Minh-Tan Pham  
IRISA - Université Bretagne Sud  
UMR 6074, F-56000, Vannes, France  
minh-tan.pham@irisa.fr

Erchan Aptoula  
Institute of Information Technologies  
Gebze Technical University, 41400 Kocaeli, Turkey  
eaptoula@gtu.edu.tr

Sébastien Lefèvre  
IRISA - Université Bretagne Sud  
UMR 6074, F-56000, Vannes, France  
sebastien.lefevre@irisa.fr

Lorenzo Bruzzone  
Dept. of Inf. Eng. Computer Science  
University of Trento, I-3812 Trento, Italy  
lorenzo.bruzzone@unitn.it

Abstract—Morphological attribute profiles (APs) are among the most effective methods to model the spatial and contextual information for the analysis of remote sensing images, especially for classification task. Since their first introduction to this field in early 2010’s, many research studies have been contributed not only to exploit and adapt their use to different applications, but also to extend and improve their performance for better dealing with more complex data. In this paper, we revisit and discuss different developments and extensions from APs which have drawn significant attention from researchers in the past few years. These studies are analyzed and gathered based on the concept of multi-stage AP construction. In our experiments, a comparative study on classification results of two remote sensing data is provided in order to show their significant improvements compared to the originally proposed APs.

Index Terms—mathematical morphology, attribute profiles, multilevel image description, image classification, remote sensing

I. INTRODUCTION

Image classification is one of the most crucial tasks in remote sensing imagery which serves for several applications in land use and land cover mapping and monitoring. With the emergence of high resolution remote sensing technology, the exploitation of the spatial information together with the spectral characteristics becomes more and more significant to characterize and discriminate different thematic classes present from the image content. Within such spatial-spectral context, morphological profiles (MPs) [1] were extensively exploited during the 2000’s [2]–[5] thanks to their multilevel analysis of spatial information by applying a sequence of opening and closing by reconstruction operators with increasing-size structuring elements (SEs). However, their high computation complexity prevent them to deal with large-size images. Besides, SEs can only model the size and scale of regions without their gray-level characteristics, thus not considering contextual features such as texture and contrast.

To overcome the MPs’ shortcomings, morphological attribute profiles (APs) [6] were proposed in early 2010’s as their generalization and consist in applying a sequence of attribute filters (AFs) which are more powerful than operators by reconstruction. These AFs can decompose the image according to different types of attribute (i.e. any geometric and statistical features of regions), not only restricted to the scale and size of SEs employed by MPs. Besides, the construction of APs can be efficiently implemented based on the hierarchical representation of image via tree structures (i.e. originally via min- and max-tree [6]), hence better dealing with large-size remote sensing images. Scalability is further ensured with parallel implementations [7].

In the past few years, a great number of research studies have been devoted to exploit and extend the use of APs applied to remote sensing image analysis, especially for classification task. These studies have been designed to improve the classification performance by focusing on the AP construction framework or adapting their use to different types of input data. In this paper, we conduct a survey on recent research studies that have been proposed and developed from the concept and application of APs. By decomposing the AP generation scheme into different stages, we regroup these studies into each specific stage in order to better provide an overview of their contribution to the general AP framework. We note that a recent survey [8] also exists but its contribution has focused only on the spatial-spectral approaches using different spectral feature extraction techniques and spatial processing by the standard APs [6]. Our survey involves more complete and detailed investigations of different developments and extensions from APs to improve their performance and optimize their construction framework.

We first recall the background of APs and highlights the key components from AP construction framework (Sec. II), before revisiting and discussing different developments from APs which have provided considerable contributions in the past few years (Sec. III). An experimental study (Sec. IV) provides a comparative evaluation of some extensions compared to the original APs by conducting supervised classification experiments on two remote sensing image data. We finally conclude the paper and indicate future research directions (Sec. V).
image $X$ 

 `tree from $X$` 

 `tree with attribute $A$` 

 `pruned tree` 

 Fig. 1. The AP generation framework which involves four main stages: tree construction, attribute computation, tree-based attribute filtering (pruning) and image reconstruction from filtered (pruned) tree.

II. PRINCIPLE OF APs

APs are multilevel image description tools obtained by successively applying a set of morphological attribute filters (AFs) [6]. Unlike usual image filtering operators which are directly performed on pixel level, AFs work on connected component (CC) level based on the concept of image connectivity. In other words, an AF is a filtering operator applied on CCs with regard to a specific attribute characterizing the size, shape, or other properties of objects and regions present in the image. That is why AFs are more general than operators by reconstruction (which are limited to the size and shape of SEs), and why APs are more general than MPs [1], [6].

The generation of the standard APs [6] from an input image can be summarized as a four-step framework (see Fig. 1):

1) construct the hierarchical tree to represent the image. In [6], the authors proposed to form one min-tree and one max-tree to encode the image;
2) compute some relevant attributes describing the geometrical and statistical features from each tree’s node which corresponds to one connected component;
3) filter the tree by keeping/removing nodes according to their attribute values compared to predefined thresholds;
4) reconstruct the image from the filtered tree.

Steps 3) and 4) can be done for different attributes (with different threshold values) to finally produce a set of filtered images (by stacking them) forming the final APs.

More formally, given a grayscale image $X : E \rightarrow \mathbb{Z}, E \subseteq \mathbb{Z}^2$, the standard generation of APs on $X$ is achieved by applying a sequence of AFs based on a min-tree (attribute thickening operators $\{\phi_k\}_{k=1}^{K}$) and a max-tree (i.e. attribute thinning operators $\{\gamma_k\}_{k=1}^{K}$) as follows:

$$\text{AP}(X) = \left\{ \phi_K(X), \phi_{K-1}(X), \ldots, \phi_1(X), X, \gamma_1(X), \ldots, \gamma_{K-1}(X), \gamma_K(X) \right\}, \quad (1)$$

where $\phi_k(X)$ is the filtered image obtained by applying the attribute thickening $\phi$ with regard to the threshold $k$. Similar explanation is made for $\gamma_k(X)$. As observed, the resulted AP($X$) is a stack of $(2K + 1)$ images including the original image, $K$ filtered images from the thickening profiles and the other $K$ from the thinning profiles.

III. RECENT ADVANCES FROM APs

As described above, the construction of APs involves four main stages which are in fact the key components that have been focused for improvements by different literature studies within the past few years. In addition, since APs basically work on panchromatic images, some pre-processing or specific adaptation procedures are required when dealing with other input data (multi-channel images, radar data, etc.). Then, spatially post-processing the output profiles to increase their description capacity for classification has also drawn attention of researchers in several research studies.

We now revisit the recently proposed developments that have provided significant contributions to adapt and improve the AP framework for remote sensing image classification (Sec. III-A to Sec. III-E). Here, our investigation will focus on three main key features:

- the adaptation of APs to other input data, in particularly to multi-channel images (Sec. III-A);
- the construction of APs using various tree representation structures (Sec. III-B);
- the AP post-processing using different feature enhancement techniques (III-E).

For other related extensions (Sec. III-C and Sec. III-D), we provide standard concepts and refer readers to the related references for further details.

A. Input data

Since APs were originally proposed to deal with panchromatic images [6], their adaptation to other kinds of remote sensing data becomes quite significant. In particular, the application of APs to the classification of multi-channel images (multispectral and hyperspectral) has become one of the hottest research topics in this field. The idea is to perform a spatial-spectral approach for classification by combining rich spectral information from these data with efficient spatial modeling capacity of APs.
The standard extension of APs on hyperspectral images was proposed in [9] by first applying the principal component analysis (PCA) on the image and then extracting APs from some first principal components. The advantage of PCA is that this low-complexity technique can compress most spectral information from the hyperspectral image into only some first principal components. Hence, applying APs on these components may perform a basic spectral-spatial feature extraction of the data. Other alternatives have been proposed to replace the PCA with the independent component analysis (ICA) [10], the kernel PCA (KPCA) [11] or other supervised methods such as the discriminant analysis feature extraction (DAFE) [12], the non-parametric weighted feature extraction (NWFE) [13], Sparse Hilbert Schmidt Independence Criterion and surrogate kernel (HSIC) [14], etc. These methods can capture more spectral relations among hyperspectral bands and hence provide better spectral information than the PCA.

Recently, the vector strategy [15] has been investigated to effectively adapt APs on multispectral and hyperspectral image. The motivation of that work is to replace the marginal strategy, i.e. independently applying APs on each image band (or each component yielded by the aforementioned feature extraction methods) and stacking them to form the extended APs, with the vector strategy which can simultaneously process all available bands based on predefined vector-ordering relations. As a result, tree construction can be done once per multivariate image and the proposed vector APs (VAPs) become promising to deal with such hyperspectral data.

While the application of APs to optical remote sensing data has been strongly focused on, their exploitation to other remote sensing data is quite limited. One may witness some tentative work on polarimetric SAR images [16], multispectral image derived features such as NDVI [17] or edge information [7] as well as on LiDAR data [17], [18]. This is still an opened topic for on-going and future research in remote sensing imagery field.

B. Tree formation

Tree formation is the first principal stage of the AP construction framework (Fig. 1). As described in Sec. II, the standard APs [6] were computed based on one max-tree and one min-tree (i.e. both are component trees). Other work has been proposed to exploit the inclusion tree (i.e. tree of shapes) [19] in order to form the self-dual APs (SDAPs). The advantages of using such a tree of shapes are twofold. First, its self-dual property enables the attribute filtering operators to simultaneously access and model both dark and bright regions from the image. And secondly, by using only one tree of shapes to replace both min-tree and max-tree [6], the feature dimension of SDAPs is reduced to half of that of APs. Consequently, SDAPs have been proved to be more efficient than APs in many research studies [13], [19], [20].

Since the above component and inclusion trees both rely on an ordering relation of the image pixels, their construction from multivariate images (e.g. multi- and hyperspectral data) is not straightforward. That is why the authors in [21] have recently investigated and proposed to use the partition trees such as α-tree and ω-tree to compute the α-APs, ω-APs, respectively. These profiles have been proved to provide fair performance compared to the standard APs. Moreover, they offer the possibility to work on multivariate images only using a single tree. Furthermore, it is also possible to rely on training samples to perform metric learning so as to provide the basic elements required for a partitioning tree [22].

C. Node attributes and threshold selection

The selection of tree node attributes as well as their thresholds for filtering on tree plays also an important role. Node attributes are usually related to the geometrical (such as size, shape) and statistical features (pixel distribution, texture, etc.) of the CC corresponding to the node. In the literature, four attributes have been used in most studies related to remote sensing image classification: area, standard deviation, moment of inertia, diagonal length of bounding box.

After deciding which attributes to calculate from nodes, the setting of their threshold values has been also concerned. Early work [6], [10], [19] usually set attribute thresholds manually based on experiments on some specific image data. However, since those values might be not applicable to other data, automatic threshold selection has drawn attention from many researchers. Some interesting studies have been proposed to automatically compute attribute thresholds using fixed formulas [23], [24], supervised approaches [25], [26] as well as granulometric characteristic functions [27], [28]. Readers are referred to the mentioned papers for further details about these attribute selection strategies.

D. Tree filtering

Once the tree is formed and the attributes together with their thresholds are selected, the next stage is to evaluate each node in order to filter (i.e. prune) the tree. Basically, there are two filtering rules including the pruning strategy (min, max, Viterbi decision rules) and the non-pruning strategy (direct, subtractive rules) [6]. Studies on the effect of different filtering rules have been done by [20], [29].

E. Post-processing of output profiles

The output AP features, i.e. sequence of filtered images in Eq. (1), can be directly fed into supervised classifiers such as SVM or Random Forest for classification task. Such direct application has provided better performance compared to MPs [1] in terms of classification accuracy as well as computational cost. However, since APs still involve quite redundant information within their high-dimension features, the post-processing of these profiles to improve their performance has been addressed in several studies. First and foremost, many studies have proposed to apply different feature selection techniques on APs to extract highly informative features and reduce their dimension. In [11], [12], [24], both linear (PCA, ICA) and nonlinear methods (ICA, KPCA, DAFE, DBFE, NWFE, etc.) have been investigated. A general framework as well as a systematic survey on spatial-spectral approaches
combining APs with these feature selection techniques have been investigated in [8].

Other work has focused on extra spatial processing of APs for better characterization of structural and textural information from the image content. Recent studies believe that when dealing with VHR remote sensing images from which regions and objects become more heterogeneous, APs may not provide a complete spatial characterization of pixels. Therefore, some efforts have been proposed to replace each AP sample response by the histogram or some first-order statistical features of the local patch around that AP’s pixel position. As a result, the local histogram-based APs (HAPs) [30], [31] and the local feature-based APs (LFAPs) [32], [33] have been proposed and proved to be more efficient for better dealing with local textures. Then, the extensions of these extra spatial processing methods on the self-dual profiles (using the tree of shapes) as well as on hyperspectral images have been provided [32].

Last but not least, we refer readers to some other frameworks using the sparse representation [34] or the deep learning approach [35] for post-processing of AP features. Also, some ensemble methods [36], [37] have been applied to better exploit and combine AP features to improve the classification performance.

IV. EXPERIMENTAL STUDY

This section describes our experimental study to evaluate the performance of the standard APs as well as some of their improvements and extensions. Supervised classification has been carried out on both panchromatic and hyperspectral image data in order to provide a comparative study. We first introduce the two data sets and the experimental setup. Then, classification results will be provided.

A. Data description

1) Reykjavik data set: The first data set is a panchromatic image of size 628 × 700 pixels acquired by the IKONOS Earth imaging satellite with 1-m resolution in Reykjavik, Iceland. This data consists of six thematic classes including residential, soil, shadow, commercial, highway and road. The image was provided with already-split training and test sets (22741 training samples and 98726 test samples). The input image together with its thematic ground truth map for testing and training sets are shown in Fig. 2(a).

2) Pavia University data set: The second data set is the hyperspectral image acquired by the ROSIS airborne sensor with 1.3-m spatial resolution over the region of Pavia University, Italy. The image consists of 610 × 340 pixels with 103 spectral bands (from 0.43 to 0.86 μm) and covers nine thematic classes: trees, asphalt, bitumen, gravel, metal sheets, shadows, meadows, self-blocking bricks and bare soil. For this image, 3921 training samples and 42776 test samples were split for classification task. The false-color image (made by combining the bands 31, 56 and 102), the ground truth map and the training set are shown in Fig. 2(b). As previously discussed, for this data set, we first performed the PCA on the image and the first four PCs (involving more than 99% of the total variance) were preserved for our experiments.

B. Setup

Supervised classification was conducted on the two data sets using the random forest classifier [38] with 100 trees. The number of variables involved in the training was set to the square root of the feature vector length. In order to evaluate and compare classification accuracy of different approaches, overall accuracy (OA), average accuracy (AA), and kappa coefficient (κ) have been taken into account. For attribute filtering, we exploited two attributes including the area and the moment of inertia. Ten area thresholds were adopted for the Reykjavik data as proposed by several papers [20], [39], [40]. For the Pavia University data, fourteen thresholds were automatically computed according to [24]. We have:

\[ \lambda_{a, \text{Rey}} = \{25, 100, 500, 1000, 5000, 10000, 20000, 50000, 100000, 150000\}, \]

\[ \lambda_{a, \text{Pav}} = \{770, 1538, 2307, 3076, 3846, 4615, 5384, 6153, 6923, 7692, 8461, 9230, 10000, 10769\}. \]

Next, the manual settings used in many studies [9], [13], [15] were adopted for the moment of inertia attribute as follows:

\[ \lambda_{i, \text{Rey}} = \lambda_{i, \text{Pav}} = \{0.2, 0.3, 0.4, 0.5\}. \]
In the following subsection, we report and compare the classification results yielded by the APs generated from different kinds of tree including: the max-tree (AP-maxT), the min-tree (AP-minT), one max-tree and one min-tree (standard APs) [6], the SDAPs [19], the α-APs and ω-APs [21]. We also provide the results of some effective post-processing techniques including the HAPs/HSDAPs [30], LFAPs/LFSDAPs [32] and the deep learning approach (deep-APs) [35]. Then, for the hyperspectral Pavia data, VAPs [15] are evaluated as well. Here, we perform standard implementation as well as equivalent parameter configuration of these methods to ensure a fair comparison.

C. Results

Tables I and II report the classification results of the Reykjavik and the Pavia data, respectively, yielded by the above mentioned methods. The calculation of each method’s feature dimension can be consulted from the related papers. Here, we provide some remarks in terms of classification performance. For both data sets, we observe that those extension methods can provide extra classification accuracy compared to the standard APs but behave differently for each image.

<table>
<thead>
<tr>
<th>Method</th>
<th>Dimension</th>
<th>Classification result</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>OA (%)</td>
</tr>
<tr>
<td>PAN</td>
<td>1</td>
<td>63.21</td>
</tr>
<tr>
<td>AP-maxT</td>
<td>16</td>
<td>73.31</td>
</tr>
<tr>
<td>AP-minT</td>
<td>16</td>
<td>72.37</td>
</tr>
<tr>
<td>AP</td>
<td>30</td>
<td>82.02</td>
</tr>
<tr>
<td>α-AP</td>
<td>16</td>
<td>77.38</td>
</tr>
<tr>
<td>ω-AP</td>
<td>16</td>
<td>76.68</td>
</tr>
<tr>
<td>SDAP</td>
<td>16</td>
<td>86.06</td>
</tr>
<tr>
<td>HAP</td>
<td>180</td>
<td>84.67</td>
</tr>
<tr>
<td>HSDAP</td>
<td>96</td>
<td>86.05</td>
</tr>
<tr>
<td>LFAP</td>
<td>60</td>
<td>87.44</td>
</tr>
<tr>
<td>LFSDAP</td>
<td>32</td>
<td>89.17</td>
</tr>
<tr>
<td>Deep-AP</td>
<td>1024</td>
<td>86.09</td>
</tr>
</tbody>
</table>

For Reykjavik image, efforts on changing the tree formation have provided some considerable effects. Indeed, the α-APs and ω-APs could outperform APs on each single max-tree or min-tree but still falls below the standard APs. Then, by using the tree of shapes, SDAPs significantly improved the accuracy with approximately 4% in OA (86.06% compared to 82.02%) and 5% in κ (0.824 compared to 0.773). Next, by post-processing the output profiles, techniques like HAP, LFAP and deep-AP have also provided important improvements. Consequently, the best classification result was obtained by using the local feature-based profiles with OA = 87.44% (κ = 0.841) using min-tree and max-tree (LFAP) and OA = 89.17% (κ = 0.863) using the tree of shapes (LFSDAP). Compared to the standard APs, an OA enhancement of 5.42% and 7.15%, respectively, was achieved.

For the hyperspectral Pavia data, we observe that APs built from different tree structures yielded different behaviors compared to the Reykjavik image. This time, the α-APs and ω-APs outperformed both APs and SDAPs. In particular, by using the ω-tree, one can achieve an OA = 96.10%, i.e. 4.44% and 1.82% better than standard APs and SDAPs, respectively. For post-processing methods, VAPs and deep-APs provided better performance compared to HAPs and LFAPs. These methods have been proved to be efficient within a spatial-spectral context usually applied to hyperspectral data. As a result, the best classification accuracy was achieved by deep-APs with OA = 99.02% and κ = 0.979. Compared to the standard APs, an enhancement of 7.36% in OA and 9% in κ was adopted.

V. Conclusion

We have conducted a survey on recent developments from morphological attribute profiles in the context of remote sensing image classification. Three key components have been focused including the AP adaptation on multi-channel image data, the use of different tree representations and the various AP post-processing procedures. Experimental study on one panchromatic and one hyperspectral image has been performed to provide a general evaluation of different methods compared to the original framework. This paper may serve as an overview of AP recent advances to readers as well as a guidance to researchers working on this framework and its alternatives within their work. We believe the exploitation and adaptation of APs in remote sensing imagery still remains an open research topic for on-going as well as future work.

ACKNOWLEDGMENT

This work was supported the Région Bretagne grant and the BAGEP Award of the Science Academy and the Tubitak grant 115E857. The authors would like to thank Prof. Jon Atli Benediktsson and Prof. Paolo Gamba for making available the Reykjavik image and the hyperspectral Pavia University data.
Algorithms for hierarchical segmentation based on the Felzenszwalb-Huttenlocher dissimilarity

Edward Cayllahua Cahuina∗†, Jean Cousty∗†, Yukiko Kenmochi∗, Arnaldo de Albuquerque Araujo ‡
Guillermo Cámara-Chávez §

∗ Université Paris-Est, LIGM, ESIEE Paris - CNRS, France
† Université Paris Descartes, Laboratoire MAP5, UMR CNRS 8145, France
‡ Universidade Federal de Minas Gerais, Computer Science Dept., Brazil
§ Universidade Federal de Ouro Preto, Computer Science Dept., Brazil

Abstract—Hierarchical image segmentation provides a region-oriented scale-space, i.e., a set of image segmentations at different detail levels in which the segmentations at finer levels are nested with respect to those at coarser levels. Most image segmentation algorithms, such as region merging algorithms, rely on a criterion for merging that does not lead to a hierarchy. Guimarães et al. proposed in 2012 a hierarchical graph-based image segmentation method relying on a criterion popularized by Felzenszwalb and Huttenlocher in 2004, hence hierarchizing the popular Felzenszwalb-Huttenlocher method. However, Guimarães et al. did not provide an algorithm to compute the proposed hierarchy. We propose a series of algorithms to compute the result of this hierarchical graph-based image segmentation method. For an image of size 321 × 481 pixels, the most efficient algorithm produces the result in half a second whereas the most naive one requires more than four hours.

I. INTRODUCTION

A hierarchical image segmentation is a series of image segmentations at different detail levels where the segmentations at higher detail levels are produced by merging regions from segmentations at finer detail levels. Consequently, the regions at finer detail levels are nested in regions at coarser levels. A hierarchical image segmentation is illustrated in Fig. 1.

Hierarchical image segmentation provides a multi-scale approach to image analysis. Hierarchical image analysis was pioneered by [1] and has received a lot of attention since then, as attested by the popularity of [2]. In [3], the global information is used to create the initial regions and then the region merging process is treated as a series of optimization problems. Mathematical morphology is also used in hierarchical image analysis with, e.g., hierarchical watersheds [4, 5], binary partition trees [6], scale-set theory [7], or quasi-flat zones hierarchies [8].

In [9] (see [10] for its preliminary version), the quasi-flat zone hierarchy is used to perform a hierarchical image segmentation. This work relies on the graph-based (GB) image segmentation algorithm proposed in [11]. GB algorithm uses a merging predicate to decide if, at a certain scale parameter, two adjacent regions of an image should be merged into a single one, thus, producing a segmented image. In its original form, GB algorithm does not directly lead to a hierarchical image segmentation. In [9], the merging predicate of GB algorithm is used along with the quasi-flat zone hierarchy to produce a hierarchical version of GB method called HGB.

Fig. 1: Illustration of a hierarchical image segmentation.

The HGB method produces satisfactory segmentation results (see [9]). Nonetheless, a precise algorithm to compute efficiently the result of HGB method is not provided in [9]. The core of HGB method is based on solving a minimization problem whose solution is the minimum observation scale at which adjacent regions in the image have to be merged. To solve this minimization, the method considers all positive real values to find such minimum observation scale.

In this article, we study HGB method and we focus on two problems that make difficult its implementation. A first difficulty is related to solving the minimization problem of HGB method for which a precise algorithmic solution is not given in [9]. We analyze this minimization process and propose three algorithms that solve it. The first one solves the minimization by searching the result in a large space of possible values, we then reduce this search space to avoid redundant computations, leading to two efficient algorithms. The second problem is related to the quasi-flat zone computation. One approach can
be to use an efficient algorithm, such as [12], to compute it at every step of HGB method. However, efficiency can be improved by only updating at each iteration the existing quasi-flat zone hierarchy instead of recomputing it from scratch. This is done with a procedure similar to the one proposed in [13, 14]. Overall, the most efficient proposed algorithm computes the result of HGB method for an image of size 321 × 481 pixels in about half a second whereas it takes over four hours with the most naive algorithm.

II. HIERARCHICAL GRAPH-BASED IMAGE SEGMENTATION

This section aims at explaining the method of hierarchical graph-based image segmentation (HGB) [10]. The hierarchy is constructed from an image via a graph representation, based on the notion of a quasi-flat zone hierarchy [8]. We first give a series of necessary notions, and then explain HGB method.

A. Basic notions

1) Hierarchies: Given a finite set \( V \), a partition of \( V \) is a set \( P \) of nonempty disjoint subsets of \( V \) whose union is \( V \). Any element of \( P \) is called a region of \( P \). Given two partitions \( P \) and \( P' \) of \( V \), \( P' \) is said to be a refinement of \( P \), denoted by \( P' \preceq P \), if any region of \( P' \) is included in a region of \( P \). A hierarchy on \( V \) is a sequence \( \mathcal{H} = (P_0, \ldots, P_\ell) \) of partitions of \( V \), such that \( P_{i-1} \preceq P_i \), for any \( i \in \{1, \ldots, \ell\} \).

2) Graph and connected-components partition: A graph is a pair \( G = (V, E) \) where \( V \) is a finite set and \( E \) is a subset of \( \{x, y\} \subseteq V \mid x \neq y \). Each element of \( V \) is called a vertex of \( G \), and each element of \( E \) is called an edge of \( G \). A subgraph of \( G \) is a graph \((V', E')\) such that \( V' \subseteq V \) and \( E' \subseteq E \). If \( X \) is a graph, its vertex and edge sets are denoted by \( V(X) \) and \( E(X) \), respectively.

Let \( x \) and \( y \) be two vertices of a graph \( G \). A path from \( x \) to \( y \) in \( G \) is a sequence \((x_0, \ldots, x_m)\) of vertices of \( G \) such that \( x_0 = x \), \( x_m = y \) and \( \{x_{i-1}, x_i\} \) is an edge of \( G \) for any \( i \) in \( \{1, \ldots, m\} \). The graph \( G \) is connected if, for any vertices \( x \) and \( y \) of \( G \), there exists a path from \( x \) to \( y \). Let \( A \) be a subset of \( V(G) \). The graph induced by \( A \) in \( G \) is the graph whose vertex set is \( A \) and whose edge set contains any edge of \( G \) made of two elements in \( A \). If the graph induced by \( A \) is connected, then we say that \( A \) is connected. The subset \( A \) of \( V(G) \) is a connected component of \( G \) if it is connected for \( G \) and maximal for this property. We denote by \( C(G) \) the set of all connected components of \( G \). Note that \( C(G) \) is a partition of \( V(G) \), which is called the connected-components partition induced by \( G \).

3) Quasi-flat zones hierarchies: Let us now present the quasi-flat zones hierarchies which provide a bijection between an edge-weighted graph and a hierarchy (see more details in [15]).

Given a graph \( G = (V, E) \), let \( w \) be a map from \( E \) into the set \( \mathbb{R} \) of real numbers. For any edge \( u \) of \( G \), the value \( w(u) \) is called the weight of \( u \) (for \( w \)), and the pair \((G, w)\) is called an edge-weighted graph.

Given an edge-weighted graph \((G, w)\), let \( X \) be a subgraph of \( G \) and let \( \lambda \) be a value of \( \mathbb{R} \). The \( \lambda \)-level edge set of \( X \) for \( w \) is defined by \( w_\lambda(X) = \{ u \in E(X) \mid w(u) < \lambda \} \), and the \( \lambda \)-level graph of \( X \) for \( w \) is defined as the subgraph \( w_\lambda(X) \) of \( X \), such that \( w_\lambda(X) \) is a connected, then we say that \( \lambda \) is a region of \( X \). Two regions \( x \) and \( y \) are adjacent if \( \{x, y\} \in E \). The weight of each edge is given by a dissimilarity measure between the linked pixels such as the absolute difference of intensity between them. The dissimilarity measure is defined by \( \Delta_{\text{inter}}(R_1, R_2) = \min \{w(\{x, y\}) \mid x \in R_1, y \in R_2, \{x, y\} \in E(T)\} \), while the within-component difference of a region \( R \) is defined by \( \Delta_{\text{intra}}(R) = \max \{w(\{x, y\}) \mid x, y \in R, \{x, y\} \in E(T)\} \).
leads to the observation scale of $R_1$ relative to $R_2$, defined by $S_{R_2}(R_1) = (\Delta_{\text{inter}}(R_1, R_2) - \Delta_{\text{intra}}(R_1)) / |R_1|$, where $|R_1|$ is the cardinality of $R_1$. Then, a symmetric metric between $R_1$ and $R_2$, called the observation scale dissimilarity, is defined by

$$D(R_1, R_2) = \max\{S_{R_2}(R_1), S_{R_2}(R_2)\}. \tag{1}$$

This dissimilarity is used to determine if two regions should be merged or not at a certain observation scale in the following.

2) Method: The HGB method is presented in Method 1. The input is an image represented by a graph $G$ with its associated weight function $w$, where the minimum spanning tree $T$ of $G$ is taken indeed. From $(T, w)$, HGB method computes a new weight function $f$ which leads to a new hierarchy $H = QFZ(T, f)$. The resulting hierarchy $H$ is considered as the hierarchical image segmentations of the initial image. Thus, the core of the method is the generation of the weight function $f$ for $T$.

To compute the new map $f$, the HGB method first initializes all values of $f$ to infinity (see Line 1). Then, an observation scale value $f(u)$ is computed for each edge $u \in E(T)$ in non-decreasing order with respect to the original weight $w$ (see Line 2). Note that each iteration in the loop requires computing the hierarchy $H = QFZ(T, f)$ (see Line 3). Once $H$ is obtained, the value $\lambda^*_H(u)$ of a finite subset $E$ of $\mathbb{R}$ is obtained by the minimization:

$$\lambda^*_H(\{x, y\}) = \min \{\lambda \in E \mid D(H^x_H, H^y_H) \leq \lambda\}. \tag{2}$$

We first consider the regions $H^x_H$ and $H^y_H$ at a level $\lambda$. Using the dissimilarity measure $D$ we check if $D(H^x_H, H^y_H) \leq \lambda$. Equation (2) states that $\lambda^*_H(\{x, y\})$ is the minimum value $\lambda$ that fulfills this minimization. Observe that the minimization involved in Equation (2) has a solution only if the maximum of $E$ is greater than the maximum possible dissimilarity value.

In the following, we assume that this assumption always holds true. Fig. 2 illustrates an example of application of Method 1.

As mentioned above, Guimarães et al. did not provide a practically efficient algorithm to compute Method 1. In order to fill this gap, the problem is twofold. Indeed, it is necessary to propose efficient (i.e., exact and fast) algorithms for (i) solving the minimization involved in Equation (2); and (ii) computing the quasi-flat zone hierarchy $QFZ(T, f)$ at each iteration of Method 1 (Lines 3 and 6).

III. ALGORITHMS FOR HGB METHOD

In this section, we investigate algorithms to compute the results of HGB method. In Sections III-A, III-B, and III-C, three algorithms to perform the minimization involved at Line 4 of Method 1 are presented. In Section III-C, we present non-incremental and incremental algorithms to obtain the quasi-flat zone hierarchy of a weight map as requested at Lines 3 and 6 of Method 1.

A. Naive minimization algorithm

We first present a naive algorithm, namely Algorithm 1, to compute the value $\lambda^*_H(\{x, y\})$ given a hierarchy $H$ and an edge $\{x, y\}$. According to Equation (2), it simply consists of considering the values of $E$ in increasing order until finding a value $\lambda \in E$ such that $D(H^x_H, H^y_H) \leq \lambda$. We remark that, when $E$ is a set of consecutive integers, for any $\lambda \in E$, the result of $n_E(\lambda)$ and $p_E(\lambda)$ can be obtained with the simple integer instruction $\lambda + 1$ and $\lambda - 1$, respectively.

Algorithm 1: HGB Naive minimization of Equation 2

```plaintext
Input : A hierarchy $H$, an edge $u = \{x, y\}$
Output: The value $\lambda^*$ such that $\lambda^* = \lambda^*_H(\{x, y\})$
1 $\lambda^* := \min \{\lambda \in E\}$ ;
2 while $D(H^x_H, H^y_H) > \lambda^*$ do
3 $\lambda^* := n_E(\lambda^*)$ ;
4 end
```

B. Minimization by range

In Algorithm 1, $D(H^x_H, H^y_H)$ is computed for every value $\lambda \in E$. However, it may well arise that at two successive values of $E$, the regions $H^x_H$ and $H^y_H$ remain the same and in this case so does $D(H^x_H, H^y_H)$. In this section, we present a second algorithm to compute $\lambda^*_H(\{x, y\})$ which allows us to reduce the amount of redundant calculation compared to Algorithm 1.

In order to obtain this reduction of redundant calculation, we search for a partition of $E$ (which, according to Equation (2), is the range of the possible values of $\lambda^*_H(\{x, y\})$)
into a finite number of discrete intervals such that, in each interval \( I = [I_{\text{min}}, I_{\text{max}}] \subseteq \mathbb{E} \), for any two values \( \lambda_1 \) and \( \lambda_2 \) in \( I \) the regions containing \( x \) and \( y \) at level \( \lambda_1 \) and at level \( \lambda_2 \) remain unchanged. Thus, in this case, we would have \( D(H_{x}^{\lambda_1}, H_{y}^{\lambda_1}) = D(H_{x}^{\lambda_2}, H_{y}^{\lambda_2}) = D_I \) which means in short that, in such interval \( I \), the dissimilarity should be computed only once. Then, it can be observed that, for any value \( \lambda \) in \( I \cap D_I \), the dissimilarity between the regions containing \( x \) and \( y \) at level \( \lambda \) of the hierarchy \( \mathcal{H} \) is below the value \( \lambda \) and that, for any value \( \lambda \) in \( I \cap D_I \), the dissimilarity between the regions containing \( x \) and \( y \) at level \( \lambda \) of the hierarchy \( \mathcal{H} \) is above \( \lambda \). Hence, the solution \( \lambda^{\ast}_{\mathcal{H}}(\{x,y\}) \) to our optimization problem can be obtained by browsing all the intervals of the considered partition of \( \mathbb{E} \).

In order to obtain such partition of \( \mathbb{E} \), we remark that if there is no edge in \( E(T) \) whose weight is between \( \lambda_1 \) and \( \lambda_2 \), then the \( \lambda_1 \)-level edge set of \( T \) for \( f \) is equal to the \( \lambda_2 \)-level edge set of \( T \) for \( f \), which implies in turn that the \( \lambda_1 \) level of \( \mathcal{H} = QFZ(T, f) \) is equal to the \( \lambda_2 \) level of \( \mathcal{H} \). Hence, to obtain a partition of \( \mathbb{E} \), such as the one described in the previous paragraph, it is sufficient to ensure that for every considered interval \([I_{\text{min}}, I_{\text{max}}]\), there is no edge whose weight is strictly between \( I_{\text{min}} \) and \( I_{\text{max}} \). In order to find the desired partition \( \mathbb{E} \), it is relevant to consider the successive values of the range \( \mathbb{R}_f = \{f(u) \mid u \in E(T)\} \) of the weight function \( f \). More precisely, from the above discussion, we deduce the following property.

Property 1: Let \( \mathcal{H} \) be a hierarchy, let \( f \) be a map from \( E(T) \) to \( \mathbb{E} \) such that \( \mathcal{H} = QFZ(T, f) \) and let \( \{x,y\} \) be any edge of \( T \). Then, we have

\[
\lambda^{\ast}_{\mathcal{H}}(\{x,y\}) = \min\{\max(\overline{m}_{\mathbb{E}}(D(H_{x}^{\lambda}, H_{y}^{\lambda})), \overline{m}_{\mathbb{E}}(p_{\mathbb{E}}(\lambda))) \mid \lambda \in \mathbb{R}_f, D(H_{x}^{\lambda}, H_{y}^{\lambda}) \leq \lambda\}. \tag{3}
\]

Thanks to Property 1, we compute \( \lambda^{\ast}_{\mathcal{H}}(\{x,y\}) \) by browsing the values of \( \mathbb{R}_f \) in increasing order until a value \( \lambda \) such that \( D(H_{x}^{\lambda}, H_{y}^{\lambda}) \leq \lambda \) is found and by taking the value \( \lambda^{\ast}_{\mathcal{H}}(\{x,y\}) \) which is simply the maximum between \( \overline{m}_{\mathbb{E}}(D(H_{x}^{\lambda}, H_{y}^{\lambda})) \) and \( \overline{m}_{\mathbb{E}}(p_{\mathbb{E}}(\lambda)) \). In order to make such process computable, it is necessary to browse the range of \( f \) in increasing order. To this end, we propose to store the values of \( f \) in a sorted linked list. Algorithm 2 provides a precise description of this process. It can be observed that when the value \( p_{\mathbb{E}}(\lambda^{\ast}_{\mathcal{H}}(\{x,y\})) \) is not yet present in the range of \( f \), the linked list representing this range is updated so that it is ready for the next iteration of the main loop in Method 1. It has to be also noted that in Method 1, the weight of every edge is initialized to the maximal value of \( \mathbb{E} \). In other words, the linked list must be initialized in Method 1 with the singleton \( \{\max(\lambda \in \mathbb{E})\} \).

C. Minimization by range

In the previous section, we reduce the size of the search space of the minimization defined in Equation (2) by considering the range \( \mathbb{R}_f \) of the function \( f \) (i.e., a characteristic function of the considered hierarchy \( \mathcal{H} \)) instead of the set \( \mathbb{E} \) of all possible scales of the hierarchy \( \mathcal{H} \) (see Property 1). In this section, we show that this search space can be further reduced, leading to a third algorithm for computing the value \( \lambda^{\ast}_{\mathcal{H}}(\{x,y\}) \), given any hierarchy \( \mathcal{H} \) and any edge \( \{x,y\} \).

In order to obtain this second reduction, we observe in Equation (2) that the only regions of the hierarchy involved in the minimization are those containing \( x \) and \( y \). Therefore, while searching for the value \( \lambda^{\ast}_{\mathcal{H}}(\{x,y\}) \), it is unnecessary to consider a scale of \( \mathcal{H} \) (i.e., a value in \( \mathbb{R}_f \)) at which the regions containing \( x \) and \( y \) are the same as those at the preceding scale. In other words, rather than considering the scales in \( \mathbb{R}_f \) for which there is a global change in the hierarchy, one can focus on the scales for which the change of the hierarchy is local to \( x \) and \( y \), i.e., when the change involves a region containing either \( x \) or \( y \).

Let \( x \) be any vertex of \( V \) and let us denote by \( B_{\mathcal{H}}(x) \) the set which contains every region \( R \) of the hierarchy \( \mathcal{H} \) such that \( x \) belongs to \( R \). The set \( B_{\mathcal{H}}(x) \) is called the branch of \( x \) in \( \mathcal{H} \). The level of a region \( R \), denoted by \( \text{level}_{\mathcal{H}}(R) \), in \( \mathcal{H} \) is the lowest index of a partition that contains \( R \) in \( \mathcal{H} \). The (branch) range of \( \mathcal{H} \) for \( x \), denoted by \( \mathbb{R}_f^x \), is defined as the set that contains the level of every region of the branch of \( x \) in \( \mathcal{H} \): \( \mathbb{R}_f^x = \{\text{level}_{\mathcal{H}}(R) \mid R \in B_{\mathcal{H}}(x)\} \). Using this notion of a branch range the following property can be deduced. The difference with Property 1 is that the range of \( f \) (being such that \( \mathcal{H} = QFZ(T, f) \)) is replaced by the union of the branch ranges of \( \mathcal{H} \) for \( x \) and for \( y \).

Property 2: Let \( \mathcal{H} \) be a hierarchy and let \( \{x,y\} \) be any edge of \( T \). Then, we have:

\[
\lambda^{\ast}_{\mathcal{H}}(\{x,y\}) = \min\{\max(\overline{m}_{\mathbb{E}}(D(H_{x}^{\lambda}, H_{y}^{\lambda})), \overline{m}_{\mathbb{E}}(p_{\mathbb{E}}(\lambda))) \mid \lambda \in B, D(H_{x}^{\lambda}, H_{y}^{\lambda}) \leq \lambda\}, \tag{4}
\]

where \( B = \mathbb{R}_f^x \cup \mathbb{R}_f^y \).

Due to Property 2, to compute \( \lambda^{\ast}_{\mathcal{H}}(\{x,y\}) \), it is sufficient to browse in increasing order the levels of the regions in the branches of \( x \) and of \( y \) until a value \( \lambda \) such that \( D(H_{x}^{\lambda}, H_{y}^{\lambda}) \leq \lambda \) is found. Finally, the value \( \lambda^{\ast}_{\mathcal{H}}(\{x,y\}) \) is determined as the maximum between \( \overline{m}_{\mathbb{E}}(D(H_{x}^{\lambda}, H_{y}^{\lambda})) \) and \( \overline{m}_{\mathbb{E}}(p_{\mathbb{E}}(\lambda)) \), where \( B = \mathbb{R}_f^x \cup \mathbb{R}_f^y \). In order to propose such an algorithm,
we need to browse in increasing order the levels of the regions in the branches of \( x \) and of \( y \). This can be done with a tree data structure, called a component tree, which represents the hierarchy. The component tree is used for various image processing tasks and is well studied in the field of mathematical morphology (see, e.g., [16] for its definition on vertex weighted graphs, [17] for the case of edge-weighted graphs and quasi-flat zone, and [18] for their generalization to directed graphs). In classification, this tree is often called the dendrogram of the hierarchy.

As any tree, the component tree of \( H \) can be defined as a pair made of a set of nodes and of a binary (parent) relation on the set of nodes. More precisely, the component tree of \( H \) is the pair \( T_H = (N, \text{parent}) \) such that \( N \) is the set of all regions of \( H \) and such that a region \( R_1 \) in \( N \) is a parent of a region \( R_2 \) in \( N \) whenever \( R_1 \) is a minimal (for inclusion relation) proper superset of \( R_2 \). Note that every region in \( N \) has exactly one parent except the region \( V \) which has no parent and is called the root of the component tree of \( H \). Any region which is not the parent of another one is called a leaf of the tree. It can be observed that any singleton of \( V \) is a leaf of \( T_H \) and that conversely any leaf of \( T_H \) is a singleton of \( V \).

In order to browse the branch of \( x \) in \( H \) from its component tree, it is enough to follow the next steps: (1) start with the node \( C \) that is the leaf \( \{x\} \), (2) consider the parent of \( C \), and (3) repeat step (2) until the root is found. Furthermore, it can be observed that the \( \text{level}_H \) attribute is increasing in the branch of \( x \); for any non-root node \( C \) in \( N \), the level of the parent of \( C \) is never less than the level of \( C \). Hence, the branch browsing process also allows browsing the branch range of \( H \) for \( x \) in increasing order. According to Property 2, in order to find the value \( \lambda_H^{x,y}(\{x,y\}) \), for any edge \( \{x,y\} \) of \( T \) and any hierarchy \( H \), we have to consider the union of the ranges of \( H \) for \( x \) and for \( y \), sorted in increasing order. This can be done by simultaneously browsing in the component tree \( T_H \) the branches of \( x \) and of \( y \). Algorithm 3 provides a precise description of a complete algorithm to find \( \lambda_H^{x,y}(\{x,y\}) \) using such a simultaneous branch browsing.

**Algorithm 3: HGB Minimization by branch**

**Input**: The component tree \( T = (N, \text{parent}) \) of a hierarchy \( H \), an edge \( u = \{x,y\} \) of \( T \), an array \( \text{level} \) that stores the level of every region of \( H \)

**Output**: The value \( \lambda^* \) such that \( \lambda^* = \lambda_H^{x,y}(\{x,y\}) \)

1. \( C_x := \{x\} \); \( C_y := \{y\} \); \( \lambda := -\infty \); \( \lambda_{\text{prev}} := -\infty \);
2. while \( D(C_x, C_y) > \lambda \) do
3.    \( \lambda_{\text{prev}} := \lambda \);
4.    \( \lambda := \min(\text{level}[\text{parent}[C_x]], \text{level}[\text{parent}[C_y]]) \);
5.    if \( \text{level}[\text{parent}[C_x]] = \lambda \) then \( C_x := \text{parent}[C_x] \);
6.    if \( \text{level}[\text{parent}[C_y]] = \lambda \) then \( C_y := \text{parent}[C_y] \);
7. end
8. \( \lambda^* := \max(\lambda_{\text{prev}}, \hat{\lambda}(D(H_x, H_y))) \);

In this section, we focus on Lines 3 and 6 of Method 1 that is, on computing the quasi-flat zone hierarchy of a weight map \( f \). This computation is repeated at every iteration of the method (i.e., for every edge of the tree \( T \)). Hence, finding an efficient way to perform this task in the context of Method 1 presents a high speedup potential.

A first implementation for this task consists simply of computing, at every iteration, the quasi-flat zone hierarchy of \( f \) using an efficient algorithm such as the one presented in [12]. However, from one iteration of the main loop of Method 1 to the next one, only one weight of the graph is updated and therefore most parts of the component tree remain unchanged (see, for instance, Fig. 3). Hence, rather than recomputing from scratch the whole component tree at each iteration, one can guess that an important speedup can be reached by updating only the part of the component tree which is affected by the single weight update considered at the present iteration. Such computation is referred to as an incremental quasi-flat zone update. In [13] and [14], the authors propose an algorithm to merge the component trees of two disjoint (adjacent) image blocks in order to obtain the component tree of the image consisting of these two blocks. Since the weight of an edge is updated only once during the whole execution of Method 1, from the initial value \( \max\{\lambda \in E\} \) to its final value, the algorithms described in [13] and [14] can be adapted to the problem of this article. The update algorithm modifies the tree structure in the following manner: first, given an edge \( u = \{x,y\} \) of updated weight \( \lambda = f(u) \), the components \( H_x^\lambda \) and \( H_y^\lambda \) are identified in the tree and then a
TABLE I: Execution times from the image of Fig. 4(a) (321 \times 481 pixels). The resulting hierarchy contains 5218 levels.

<table>
<thead>
<tr>
<th>QFZ Algorithm</th>
<th>Minimization Algorithm</th>
<th>Execution times (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total</td>
<td>QFZ</td>
</tr>
<tr>
<td>Non-Incremental</td>
<td>Algorithm 1</td>
<td>14666.08</td>
</tr>
<tr>
<td>Non-Incremental</td>
<td>Algorithm 2</td>
<td>13392.51</td>
</tr>
<tr>
<td>Non-Incremental</td>
<td>Algorithm 3</td>
<td>15186.25</td>
</tr>
<tr>
<td>Incremental</td>
<td>Algorithm 1</td>
<td>1487.96</td>
</tr>
<tr>
<td>Incremental</td>
<td>Algorithm 2</td>
<td>15.42</td>
</tr>
<tr>
<td>Incremental</td>
<td>Algorithm 3</td>
<td>0.49</td>
</tr>
</tbody>
</table>

A new node is created in the tree structure at level \( \lambda + 1 \), which represents the union of these two components. Finally, the algorithm identifies the ancestors of these components in the tree, and updates the parent/hood relationship of these nodes. This is done until the root is found. Consequently, only the components containing \( x \) and \( y \) are involved in the update algorithm and we do not need to recompute a whole hierarchy at every iteration.

IV. ASSESSMENTS

The experiment aims at measuring and comparing the execution times of all the variations of our algorithms, which are previously presented for HGB method; as we have three variations for the minimization step (Line 4 in Method 1), Algorithms 1, 2 and 3, and two variations for quasi-flat zone computation (Lines 3 and 6 in Method 1), the non-incremental one [12] and the incremental one [13, 14], the total number of all the combinations is six. The algorithms were implemented in C and executed on a computer with a 3.2 GHz CPU, 8GB RAM. The six algorithms were executed on the image of Fig. 4 (a). Fig. 4 (b) shows the resulting hierarchical segmentation, from which we see that a large number of regions and hierarchical levels were produced.

Table I shows the results for all the variations. We observe that using the incremental quasi-flat zone computation provides a great gain in efficiency compared to the non-incremental approach. For the minimization step, Algorithm 1 is the least efficient of all. It is important to notice that Algorithm 3 is much faster than Algorithm 2, which validates that minimization by branch is the most efficient of the three algorithms to solve the minimization problem. For further assessment, we tested our fastest algorithm on the 500 images of the Berkeley dataset leading to an average execution time of 0.19 ± 0.02 seconds.

V. CONCLUSIONS

We first investigated the HGB method [10] with the aim of proposing practical algorithms for its implementation on images. We focused on two steps for improving efficiency: (i) the minimization involved in Equation (2), and (ii) the computation of the quasi-flat zones hierarchies. Concerning (i), we presented two properties which allow us to improve an algorithm to compute the minimum value \( \lambda_H^*(\{x, y\}) \) step by step. The most efficient one is Algorithm 3, as confirmed by the assessments (see Table I). In order to compute efficiently the quasi-flat zone hierarchy (ii), the incremental update strategy [13] was used. The improvement over the non-incremental strategy is also confirmed by the experiment.

REFERENCES

Intrinsic Quality Analysis of Binary Partition Trees

Jimmy Francky Randrianasolo$^1$, Camille Kurtz$^2$, Pierre Gańcarski$^1$, Éric Desjardin$^3$, Nicolas Passat$^3$

$^1$ Université de Strasbourg, CNRS, ICube, France
$^2$ Université Paris-Descartes, LIPADE, France
$^3$ Université de Reims Champagne-Ardenne, CReSTIC, France

Abstract—The binary partition tree (BPT) is a well-known hierarchical data-structure, frequently involved in image segmentation procedures. The efficiency of segmentation based on BPTs depends on the segmentation process (“how to use a BPT?”), but also on the quality of the data-structure (“how to build a BPT?”). In this article, we propose a scheme for BPT quality analysis, with the purpose of answering the latter question. It relies on the observation of the very structure of a BPT, with respect to a given ground-truth example. Our hypothesis is that such intrinsic scheme can bring relevant clues about the ability of a BPT to provide correct segmentation results. Experiments carried out on satellite images illustrate the relevance of this scheme.

Index Terms—Binary partition tree, supervised quality evaluation, image segmentation, mathematical morphology.

I. INTRODUCTION

The Binary Partition Tree (BPT) [1] is a hierarchical data-structure that can be used for image modelling. It is a binary tree that can represent the image content at different levels of detail. By contrast to other hierarchical models (see, e.g., [2]–[7]) mostly built from the only information contained in the image, the BPT construction relies both on the information embedded in the image, and on external information, namely a prior knowledge related to the structures of interest.

When BPTs are involved in segmentation procedures, the quality of the result then depends on the way to handle the BPT (for defining an optimal partition of the image support), but also the way to build this BPT. This construction step is crucial. Indeed, a badly constructed BPT will not allow for a good segmentation result, by providing an non-adapted search space. In other words, when considering BPT-based image segmentation, a good segmentation algorithm has to be applied on a good image tree data-structure.

Many articles have been devoted to BPT-based segmentation, especially for remote sensing [8]–[16]. In this context, various metrics were investigated (spectral, spatial, geometric, etc.) for embedding a prior knowledge. Their design strongly influences the resulting hierarchical structures and thus the quality of the subsequent segmentation results. But, surprisingly, there exist very few works devoted to evaluate the capacity of a BPT to provide relevant segmentation results.

It is fundamental to distinguish segmentation evaluation, which has been widely investigated (e.g., in [17]–[22]), and BPT evaluation. Indeed, the latter is the evaluation of a search space of putative segmentation results; the purpose is then to help the user to choose a right BPT, not to use it the right way. To the best of our knowledge, the only framework for assessing the quality of a hierarchy of partitions for segmentation purpose was proposed in [23]. It consists of selecting, in the tree, a set of segments matching an ideal partition that is forced to be in the hierarchy; its selection is expressed as a linear fractional combinatorial optimisation problem. Such approach requires a full ground-truth partition of the image whereas only partial segments representing some objects of interest may be available for some particular applications.

Our purpose is to evaluate the quality of a BPT or, equivalently, its construction process. In this paper, we aim to show that reliable clues for such a quality analysis can be obtained by directly investigating the BPT structure with respect to partial ground-truth (GT) examples. Indeed, our hypothesis is that a BPT provides —by its inner structure and its spatial embedding in the image support— some information about its relevance and its ability to extract specific objects of interest.

This article is organised as follows. In Section II, we first remind some background notions related to the BPT. Then, in Section III, we present an example-based subtree extraction for reducing the data-space to consider for each GT example. In Section IV, a pre-processing is presented, in order to estimate the relevance of the intrinsic quality analysis. Combinatorial and quantitative analyses are then proposed, in Section V, for intrinsic BPT quality evaluation. In Section VI, experiments on remote sensing images are described.

II. BINARY PARTITION TREE

An image $I$ is defined on a set of points $\Omega$. These points are spatially organized; this is modelled by a neighbourhood relationship, i.e. an adjacency (irreflexive, symmetric) relation $\sim$ on $\Omega$. A BPT associated to the graph $(\Omega, \sim)$ is a tree $T = (N, \preceq)$, i.e. a directed, connected, acyclic graph ($\preceq$ reflects the construction link between nodes). The set $N$ is partitioned into three subsets: $R = \{N_\Omega\}$ such that $N_\Omega = \Omega; B \subset 2^\Omega$; and $L = \{N_x\}_{x \in \Omega}$ such that $\forall x \in \Omega, N_x = \{x\}$. The nodes of these subsets satisfy the following properties:

\[
\forall N \in R \cup B, d^+ (N) = 2 \quad (1)
\]

\[
\forall N \in L, d^+ (N) = 0 \quad (2)
\]

\[
\forall N \in B \cup L, d^- (N) = 1 \quad (3)
\]

\[
\forall N \in R, d^- (N) = 0 \quad (4)
\]

\[
\forall N \in R \cup B, N = \bigcup_{N' \preceq N} N' \quad (5)
\]

This work was partially supported by the French Agence Nationale de la Recherche under grants ANR-12-MONU-0001, ANR-15-CE23-0009 and ANR-17-CE23-0015.
where \( d^+(N) = |\{N' \in \mathcal{N}, N \varsubsetneq N'\}| \) and \( d^-(N) = |\{N' \in \mathcal{N}, N' \varsubsetneq N\}| \). A BPT \( \mathcal{T} \) of \((\Omega, \preceq)\) provides a family \( \mathcal{N} \) of subsets of \( \Omega \). These subsets are hierarchically organised from the whole set \( \Omega \), to the singleton sets \( \{x\} \), \( x \in \Omega \), with respect to the inclusion relation. This hierarchical organisation is characterised by the fact that an element \( N \) of \( \mathcal{N} \) is associated, via \( \preceq \), to exactly two elements \( N_1, N_2 \) of \( \mathcal{N} \) that form a binary partition of \( N \). In other words, we have \( N = N_1 \cup N_2, N_1 \cap N_2 = \emptyset \) and \( N_1, N_2 \neq \emptyset \). The elements of \( \mathcal{N} \) are the nodes of \( \mathcal{T} \), the node \( \Omega \) is the root of \( \mathcal{T} \), the singleton nodes of \( \mathcal{L} \) are the leaves of \( \mathcal{T} \).

III. Example-based subtree extraction

Our purpose is to evaluate how well a BPT \( \mathcal{T} \) is adapted to provide nodes matching with a segment representing a GT example \( G \). Some nodes of \( \mathcal{N} \) do not intersect \( G \); in such case, they are useless for the analysis with respect to \( G \). Then, we focus only on the nodes \( N \in \mathcal{N} \) such that \( N \cap G \neq \emptyset \). Their subset is computed, in a bottom-up fashion, by first selecting all the leaves \( L \in \mathcal{L} \) that intersect \( G \), and then preserving iteratively all the parent nodes connected to any such leaves by the \( \preceq \) relation, until the root \( N_0 \) (Fig. 1(a)).

The obtained subset of nodes \( N_G \subseteq N \) induces a subtree \( \mathcal{T}_G \) of \( \mathcal{T} \), of root \( N_0 \) and of leaves \( \mathcal{L}_G = \{ L \in \mathcal{L}, L \cap G \neq \emptyset \} \). This subtree may not be binary. Indeed, there may exist nodes \( N \in \mathcal{N}_G \) that have one child \( N' \in \mathcal{N}_G \) (i.e., \( N \varsubsetneq N' \)). This happens when the only other node \( N'' \subseteq N \) such that \( N \varsubsetneq N'' \) does not intersect \( G \).) Such nodes \( N \) are called unary nodes (Fig. 1(b)), by contrast with the other binary nodes that do have two children nodes in \( \mathcal{N}_G \).

The intersection of a unary node \( N \) with \( G \) is the same as for its only child \( N' \); if \( N \varsubsetneq N' \), we have \( N \cap G = N' \cap G \), while \( N' \subseteq N \). Then, a unary node increases the amount of false positive (FP) material with respect to \( G \), compared to its descendants. In particular, this is true in the upper part of the tree \( \mathcal{T}_G \), between the root \( N_0 \) and the first binary node \( N_G \) of \( \mathcal{T}_G \). Indeed, there exists, within \( \mathcal{N}_G \), a sequence of successive nodes \( N_0 \varsubsetneq N_1 \varsubsetneq \ldots \varsubsetneq N_k = N_G \) (\( k \geq 0 \)) such that all \( N_i \) (\( 0 \leq i < k \)) are unary. We can relevantly remove from \( \mathcal{T}_G \) all these nodes \( N_i \) and only preserve, as pseudo-root, the first binary node \( N_G \) (red node in Fig. 1(a)). By construction, \( N_G \) is the smallest node of \( \mathcal{N} \) that includes \( G \).

IV. Relevance of the intrinsic quality analysis

A BPT \( \mathcal{T} \) is built in a bottom-up fashion. The construction process starts from the set of leaves \( \mathcal{L} \) and progressively creates new nodes by iteratively merging pairs of adjacent nodes. These pairs are chosen with respect to a given metric, and the inclusion relation between two nodes and their merged union practically defines the \( \preceq \) relation.

The initial set of leaves is often chosen as \( \mathcal{L} = \{ N_x \}_{x \in \Omega} \), i.e. each leaf corresponds to exactly one point of the image. For practical reasons, \( \mathcal{L} \) can sometimes be defined as a set of larger nodes (e.g., flat zones or superpixels). This partition is generally non-correlated to the node-merging strategy. Nevertheless, it has an important impact on the relevance of the proposed evaluation framework. In particular, two properties of this initial partition are crucial.

a) Granularity: The granularity \( \gamma \) is defined as the ratio between the size of \( G \) (number of points) and the size of \( \mathcal{L}_G \) (number of leaves). It is defined as

\[
\gamma = \frac{|\mathcal{L}_G|}{|G|}
\]

and lies in \([0, 1]\). The higher the granularity, the most relevant the intrinsic quality analysis carried out on \( \mathcal{T}_G \). Indeed, for \( \gamma = 1 \), the number of leaves that intersect \( G \) is equal to the number of points of \( G \). Then, each leaf contains exactly one point of \( G \), and the ability of the BPT to allow for a segmentation that correctly fits \( G \) highly depends on the way to merge the nodes. By contrast, if \( \gamma = 1/|G| \), then one leaf...
already includes $G$, and the way of further building $\mathcal{T}_G$ has no influence on the segmentation of $G$.

b) Discordance: The discordance $\delta$ is defined as the relative quantitative error on the size of $G$ induced by $\mathcal{L}_G$. It is defined as
\[ \delta = \frac{1}{|G|} \sum_{L \subseteq \mathcal{L}_G, L \neq G} \min\{|L \setminus G|, |L \cap G|\} \]  
and lies in $[0, 1]$. The lowest the discordance, the most relevant the intrinsic quality analysis carried out on $\mathcal{T}_G$. Indeed, $\delta = 0$ means that the initial partition provides a set of leaves that perfectly fits $G$. Then, the quality of the BPT directly depends on the ability to merge these nodes. By contrast, $\delta \approx 1$ means that many leaves of $\mathcal{L}$ partially intersect both $G$ and the remainder of $\Omega$. Then, the ability to finally obtain a "good" segmentation of $G$ is low and weakly depends on the ability to merge the nodes when building $\mathcal{T}$.

For $G$ sufficiently large, a high value of granularity implies a low value of discordance. However, the counterpart is not true: a low value of discordance can be obtained for a low value of granularity. Note that when the initial partition is composed of leaves that are points of the image, we have $\gamma = 1$ and $\delta = 0$. In such case, the quality of the BPT only depends on the node merging process.

V. INTRINSIC QUALITY ANALYSIS

A. Combinatorial analysis

The subtree $\mathcal{T}_G$ is composed of $n$ nodes (including 1 root and $l$ leaves) and $n - 1$ edges $\sim_i$. Observing the status of these nodes provides us with quality clues of the BPT.

A node $N \in N_G$ is pure (resp. impure) if $N \subseteq G$ (resp. $N \not\subseteq G$). The classification of the leaves of $\mathcal{L}_G$ into pure / impure can be done by observing their support, compared to $G$ (Fig. 2). We set $l_p$ (resp. $l_i$) the number of pure (resp. impure) leaves, respectively (we have $l_p + l_i = l$). The purity / impurity of the other nodes can then be computed iteratively: if $N \in N_G$ is a unary node, then it is pure; if $N \in N_G$ is a binary node, with $N \sim_i N', N''$, it is pure if $N'$ and $N''$ are pure, otherwise it is impure.

A "good" BPT construction should preserve as much as possible the purity of nodes; avoid to merge pure and impure nodes; and avoid to increase the size of impure nodes. Thus:

- the merging in $\mathcal{T}$ leading to a unary node in $\mathcal{T}_G$ is a bad operation, as it creates from impure —and sometimes pure— nodes, an impure node with a greater amount of points outside of $G$;
- the merging in $\mathcal{T}$ leading to a binary node from two pure nodes in $\mathcal{T}_G$ is a good operation, as it allows one to converge towards $G$;
- the merging in $\mathcal{T}$ leading to a binary node from two impure nodes in $\mathcal{T}_G$ is a good operation, as it neither deteriorates pure areas, nor increases the amount of points out of $G$;
- the merging in $\mathcal{T}$ leading to a binary node from a pure and an impure node in $\mathcal{T}_G$ is a bad operation, as it makes the result diverging from $G$.

It is possible to count the number of each kind of nodes: $u_i$, $b_{pp}$, $b_{ii}$ and $b_{pi}$, for the unary nodes, and binary nodes built from pure-pure, impure-impure, pure-impure couples. We have $u_i + b_{pp} + b_{ii} + b_{pi} + l_p + l_i = n$, and a good BPT should minimise $u_i$ and $b_{pi}$, while maximising $b_{pp}$ and $b_{ii}$. In particular, a "perfect" BPT should satisfy:
\[ u_i = 0 \]  
\[ b_{pp} = l_p - 1 \]  
\[ b_{ii} = l_i - 1 \]  
\[ b_{pi} = 1 \]  
(except when $l_i = 0$, where $b_{pp}$ should be equal to $l - 1$, and all others to 0).

From this classification of nodes, and the combinatorial analysis of their population, it is then possible to build a wide range of structural measures that quantify the difference of quality between BPTs. Examples of such measures will be proposed in Section VI.

B. Quantitative analysis

A quantitative assessment of the quality of the BPT can also be carried out by observing the lowest set including $G$ and the greatest set included in $G$ which can be built from $\mathcal{T}_G$.

Let us first focus on the lowest set including $G$. By construction, it is the root of $\mathcal{T}_G$, namely $N_G$. The interesting information carried by $N_G$ is the amount of points outside of $G$ (Fig. 3(a)). More precisely, this amount $|N_G| - |G|$ has to be compared to the amount that could be theoretically
obtained from the initial partition of the leaves \( \mathcal{L}_G \), namely 
\[ \sum_{L \in \mathcal{L}_G} |L| - |G| \]. Computing the difference, or the ratio, 
between these two values allows us to assess the quantitative 
error related to the existence of unary nodes in \( \mathcal{T}_G \), i.e. the 
addition of non-relevant zones to the expected exhaustive 
segmentation of \( G \). The lower this value, the better the ability 
of the BPT to take advantage of the adequacy of the initial 
partition to the GT segment. Second, let us focus on the 
greatest set included in \( G \) (Fig. 3(b)). It is defined as the 
union of all the pure nodes of \( N_p \) whose parents are impure; 
we note this set \( N_{p}^c \). The first interesting value is the amount 
of points of \( G \) outside this pure set: 
\[ |G| - \sum_{L \in \mathcal{L}_p} |L| \], where \( \mathcal{L}_p = \{ L \in \mathcal{L} \mid L \subseteq G \} \). 
The lower this value, the better the ability of the BPT to take 
advantage of the potential adequacy of the initial partition to 
the GT segment. The second interesting value of interest is 
the size of \( N_{p}^c \), i.e. number of nodes required to form the best 
segmentation lower than \( G \). The lower this value, the better 
the ability of the BPT to avoid over-segmentation.

Of course, these two values have to be considered jointly. 
Indeed, a low value of \( |N_{p}^c| \) is not meaningful if the union 
of the nodes of \( N_{p}^c \) is highly degraded with respect to the 
initial partition. Similarly, an optimal union of the nodes of 
\( N_{p}^c \) is meaningless if these nodes are numerous (i.e., could not 
be merged together by the BPT, and were instead fused into 
impure greater nodes).

VI. EXPERIMENTAL STUDY

In this section, our aim is to emphasise in which extent our 
intrinsic analysis framework can be used for the quality 
evaluation of BPTs. In particular, it may be seen as a companion 
tool for the extrinsic analysis framework proposed in [24].

A. EXPERIMENTAL PROTOCOL

Our experiments use a crop (500 \(^2\) pixels) of Fig. 4(a). We 
consider 3 types of BPTs: \( BPT_{std} \), \( BPT_{ndvi} \) and \( BPT_{ndwi} \), 
each one built using a metric based on the radiometric 
intensity, the Normalized Difference Vegetation (NDVI) and Water 
Indices (NDWI), respectively. In order to show the influence 
of the initial partition \( \mathcal{L} \) on the intrinsic evaluation analysis, 4 
BPTs are built for each type, by using various initial partitions: 
\( \mathcal{L}_1 \) is composed of all pixels from the image support (250,000 
regions); \( \mathcal{L}_2 \) is composed of 18,750 regions; \( \mathcal{L}_3 \) is composed 
of 12,500 regions; and \( \mathcal{L}_4 \) is composed of 6,250 regions.

When \( \gamma \) and \( \delta \) are satisfactory, we use our framework to 
estimate the ability of these BPTs to generate potential nodes 
well matching a set \( \mathcal{L}_G \) of 13 GT examples from each we 
extract a subtree \( \mathcal{T}_G \). Examples of GT examples are illustrated 
in Fig. 5.

To obtain values normalised in \([0,1]\), we define two scores 
\[ c_{b1} = (b_{pp} + b_{ii})/(u_{i} + b_{pp} + b_{ii} + b_{pi}) \]  
\[ c_{b2} = b_{pp}/(b_{pp} + b_{pi}) \] (12) (13)

The closer these values from 1 (resp. 0), the more (resp. less) 
acceptable the BPTs. In particular, for a GT example, we 
assume that a BPT provides an acceptable matching region 
if these scores are higher than 0.5.

We used the definitions proposed in Section V-B for the 
quantitative analysis. For a better readability, we set 
\[ q_{t1} = |N_G \setminus G| - |( \bigcup_{L \in \mathcal{L}_G} L) \setminus G| = |N_G \setminus \bigcup_{L \in \mathcal{L}_G} L| \] (14) 
\[ q_{t2} = |N_G \setminus G|/(| \bigcup_{L \in \mathcal{L}_G} L|) \] (15)

We also simplify some notations by using 
\[ \nu_p = (\sum_{L \in \mathcal{L}_p} |N|) \] and 
\[ \lambda_p = (\sum_{L \in \mathcal{L}_p} |L|) \]. Thus, we can define 
\[ q_{t3} = (|G| - \nu_p) - (|G| - \lambda_p) = \lambda_p - \nu_p \] (16) 
\[ q_{t4} = |N_{p}^c| \] (17)

B. RESULTS AND DISCUSSION

Table I presents the combinatorial scores obtained from 
\( BPT_{std} \) built from the initial partition \( \mathcal{L}_1 \). By construction 
of \( \mathcal{L}_1 \), the values of \( \gamma \) and \( \delta \) are optimal. We can notice that 
the scores are generally around 0.9 and above 0.5, namely our 
point of reference. This suggests that the intrinsic structure 
of \( BPT_{std} \) can provide acceptable segmentation results for the 
GT examples \( G_1, G_9, G_{11} \) and \( G_{12} \). We can observe that the 
\( c_{b1} \) and the \( c_{b2} \) scores are high. Indeed, the values of 
\( u_{i}, b_{pp}, b_{ii} \) and \( b_{pi} \) are not far from the values that correspond to a 
perfect BPT (see Eqs. (8–11)).

Table II presents some statistical information about various 
\( BPT_{std} \) built using \( \mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3 \) and \( \mathcal{L}_4 \) as initial partitions,
with respect to the GT segment $G_1$. The values of $cb_1$ and $cb_2$ for the first and the third lines suggest that the BPTs built by using $L_1$ is able to provide acceptable matching regions for the GT example $G_1$ while the other BPTs are not. The low scores of the BPTs built by using $L_2$, $L_3$ and $L_4$ are correlated with the non-optimal values of $\gamma$ that are far from 1.

From another point of view, the values in Table III show the differences between the various BPTs: one of them induces a lower amount of unary nodes ($BPT_{std}$) compared to the others ($BPT_{ndvi}$, $BPT_{ndwi}$). This suggests that the hierarchical structure of the $BPT_{std}$, for the GT example $G_1$, is likely to better provide acceptable segment matching.

For a first quantitative analysis, Table IV gives information about the absolute ($qt_1$) and relative ($qt_2$) amount of false positives, induced by the construction of the BPT. The lower these values, the better the ability of the BPT to use at best the potential adequacy of the initial partition to the GT segment. Here, the $BPT_{std}$ built from the initial partition $L_2$ only provides interesting prediction for the GT example $G_1$, where the estimated false positives ($qt_1 = 62$) are less than the half of the size of the GT example. In the other cases, we observe a high difference between the size of the chosen root of the subtree and the size of the GT segment. In the context of urban imaging, the excessive size of these roots can be explained by the presence of disturbing objects (e.g., shadows, spots on the roof) that make small pieces of regions more similar to other far located objects than to their neighbours. Those kinds of segments tend to not fuse rapidly with their neighbours and often persist until the end of the BPT construction.

In Table V, we observe that, for $BPT_{std}$, the size $|N_G|$ of the selected root is close to that of $|G|$ for the GT segment $G_1$. The values of $qt_1$ for this GT then suggest that the numbers of points representing potential false positives when segmenting are not high compared to the values for $BPT_{ndvi}$ and $BPT_{ndwi}$. This result shows that the use of the metric based on radiometric intensities is more relevant for this study. This is probably due to the fact that the studied image contains urban objects, with a corresponding GT map mainly composed of built areas. Then, the metrics based on NDVI and NDWI are weakly adapted for the detection of such kinds of objects.

A second quantitative analysis aims to estimate the amount of false negative points induced by the construction process. Table VI presents quantitative information about the subtrees extracted from the $BPT_{std}$ built from an initial partition $L_3$. The score $qt_3$ estimates the increase of false negative points during the construction process of the tree, while $qt_4$ shows the relevance of each subtree by showing the number of maximal pure nodes well merged in the hierarchical structure.

For the GT example $G_{12}$ (Fig. 5(d)), the values of $|G| - \nu_p$ are visually illustrated in Fig. 6, showing in orange the set of subregions that are already well-formed in the extracted subtree. We notice both from Table VII and Fig. 6 that the hierarchical structure of the subtrees extracted from the three types of BPTs are interesting since they generate matching subregions with fair size. The values of $qt_4$ in Table VII suggest that the $BPT_{ndvi}$ and the $BPT_{ndwi}$ may not be

| $|G|$ | $|N_G|$ | $|N_G| \setminus |G|$ | $|(U \setminus L_G) \setminus |G| |$ | $qt_1$ | $qt_2$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>182</td>
<td>259</td>
<td>77</td>
<td>15</td>
<td>0</td>
<td>62.0</td>
</tr>
<tr>
<td>22378</td>
<td>75948</td>
<td>53570</td>
<td>605</td>
<td>52965</td>
<td>88.545</td>
</tr>
<tr>
<td>566</td>
<td>101525</td>
<td>100959</td>
<td>78</td>
<td>100881</td>
<td>1294.346</td>
</tr>
<tr>
<td>541</td>
<td>30894</td>
<td>30353</td>
<td>67</td>
<td>30286</td>
<td>453.030</td>
</tr>
</tbody>
</table>

| $|G|$ | $\nu_p$ | $\lambda_p$ | $|G| - \nu_p$ | $|G| - \lambda_p$ | $qt_3$ | $qt_4$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>182</td>
<td>25</td>
<td>45</td>
<td>137</td>
<td>139</td>
<td>18</td>
<td>1</td>
</tr>
<tr>
<td>22378</td>
<td>20803</td>
<td>21603</td>
<td>1575</td>
<td>775</td>
<td>800</td>
<td>69</td>
</tr>
<tr>
<td>566</td>
<td>505</td>
<td>539</td>
<td>61</td>
<td>27</td>
<td>34</td>
<td>11</td>
</tr>
<tr>
<td>541</td>
<td>423</td>
<td>453</td>
<td>118</td>
<td>88</td>
<td>30</td>
<td>3</td>
</tr>
</tbody>
</table>

| $|G|$ | $|G| - \nu_p$ | $|G| - \lambda_p$ | $qt_3$ | $qt_4$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>182</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22378</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>566</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>541</td>
<td>45</td>
<td>32</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| $|G|$ | $|G| - \nu_p$ | $|G| - \lambda_p$ | $qt_3$ | $qt_4$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>182</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22378</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>566</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>541</td>
<td>45</td>
<td>32</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| $|G|$ | $|G| - \nu_p$ | $|G| - \lambda_p$ | $qt_3$ | $qt_4$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>182</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22378</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>566</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>541</td>
<td>45</td>
<td>32</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| $|G|$ | $|G| - \nu_p$ | $|G| - \lambda_p$ | $qt_3$ | $qt_4$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>182</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22378</td>
<td>10</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>566</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>541</td>
<td>45</td>
<td>32</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
relevant for the GT segment $G_{12}$. Indeed, the values of $q^4_t$ are high, while the amount $q^3_t$ of false negative points is acceptable although the set of maximal pure nodes almost matches with the GT example $G_{12}$ (see Fig. 6). For a case where the value of $q^4_t$ may be the best for a BPT, its high value of $q^4_t$ predicts a high risk of potential bad node fusions.

VII. CONCLUSION

In this paper, we proposed a scheme for BPT quality analysis, by observing the intrinsic information contained in the hierarchical data-structure. With respect to a set of user-defined GT segments, it may help to better understand the potential results that we can obtain from the BPTs. Experiments, on examples of combinatorial and quantitative analysis on remote sensing images, illustrate how the intrinsic information of a BPT can lead to the estimation of its potential usefulness for further segmentation process. It is important to notice that the relevance of this analysis highly depends on the initial partition used during the construction of the BPT (see Section IV). When such partitions are correctly chosen, the proposed framework then provides us with relevant intrinsic information about the quality of BPTs, and can then be used as a fair perspective or retrospective evaluation tool.

As a perspective of this work, extending the proposed strategy to multiple GT examples will be interesting since the evaluation will be done for a partial partition containing (non-)connected segments of reference. In [24], we proposed a scheme for extrinsic analysis of the quality of the BPTs allowing the user to use various metrics of quality. Coupling these two —intrinsic and extrinsic— schemes will be relevant since it will give access to a more complete evaluation of the quality of the BPTs in the context of image segmentation. Adapting the proposed method to other kinds of hierarchies will also be investigated, opening the gate to other applications.

REFERENCES


Oral Session E

Automatic Marking of Descriptive Answers

Tuesday May 15, 2018, 10:00 AM

Vu Tran Minh Khuong, Huy Quang Ung, Cuong Tuan Nguyen and Masaki Nakagawa
*Clustering Offline Handwritten Mathematical Answers for Computer-Assisted Marking*

Huy Quang Ung, Vu Tran Minh Khuong, Anh Duc Le, Cuong Tuan Nguyen and Masaki Nakagawa
*Bag-of-features for clustering online handwritten mathematical expressions*

Palaiahnakote Shivakumara, Umapada Pal, Tong Lu, Tapabrata Chakraborti and Michael Blumenstein
*A New Roadmap for Evaluating Descriptive Handwritten Answer type*

Tsunenori Ishioka and Masayuki Kameda
*AI-based Automated Japanese Short-answer Scoring and Support System*

Hemmaphan Suwanwiwat, Abhijit Das, Miguel Ferrer, Umapada Pal and Michael Blumenstein
*An Investigation of Discrete Hidden Markov Models on Handwritten Short Answer Assessment System*
Clustering Offline Handwritten Mathematical Answers for Computer-Assisted Marking

Vu Tran Minh Khuong, Huy Quang Ung, Cuong Tuan Nguyen, Masaki Nakagawa
Department of Computer and Information Sciences
Tokyo University of Agriculture and Technology
Tokyo, Japan
vtmkhuong@gmail.com, ungquanghuy93@gmail.com, ntcuong2103@gmail.com, nakagawa@cc.tuat.ac.jp

Abstract—This paper presents clustering of handwritten mathematical answers scanned from paper exams to help makers to mark them efficiently. We extract features from multiple levels of recognition: low-level directional features, bag-of-symbols, bag-of-relations and bag-of-positions. We apply the k-means++ algorithm for each level of features and all the combinations with better performance. Moreover, we propose an incremental refinement approach suitable for computer-assisted exam marking. Then, we propose a marking cost function to measure clustering for our purpose and compare it with the conventional clustering-based approach.

Keywords—clustering, handwritten answers, computer-assisted marking, offline handwritten mathematical expressions.

I. INTRODUCTION

Examinations play important roles in education. They help teachers evaluate learners’ understanding and abilities to solve problems in order to prepare for the next stage of teaching. They also motivate learners and guides further learning. However, marking exams takes a large amount of time and effort. If it takes time to return marking results, the effect of feedback for the learners decreases. School teachers need to mark large quantities of student assignments and exams.

Mark sheets and Computer/Web-based testing have been introduced to solve this problem. Due to the current level of IT, however, they often restrict the questions to selection types. Descriptive questions have been generally excluded in those IT-empowered examinations. The time and cost to mark answers are reduced and marking errors are lessened but they sacrifice to measure learners’ deep understanding and their ability to think. Even if an examinee does not understand a question or cannot answer it, he/she can get a correct mark by selecting a correct answer through guessing. It has also the bad side effect for learners to “select” rather than “think”.

Automatic marking is a solution to the exam marking problem. Extensive research has been carried out on essay assessment [1], [2], [3] and even handwritten essay scoring was reported [4]. Plenty of research has been made on autograding of programming [5], [6], [7], but automatic marking is mostly unable to handle partially correct answers. Among others, the most practical problem of the automatic marking is that it cannot be used without learners’ confirmation of marking.

Large-scale examinations such as national-wide qualification examinations and entrance examinations do not provide such opportunities for examinees to confirm marking so that the marking of those examinations cannot be automated beyond selection types.

Another solution is computer-assisted marking. It may provide efficiency and even reliability for marking. One of the most promising approaches is to cluster answers. If answers are well clustered, they can be marked efficiently and marking errors and fluctuations will be decreased. Since the final marking is made by human markers, examinees’ anxieties will be also reduced. For large-scale examinations whose marking must be made in a short period without errors, computer-assisted marking is a promising solution. For daily exams as well, teachers’ burden will be reduced and more time will be devoted for the explanation to learners.

In Japan, it has been decided by the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT) that National Center Test for University Admissions (NCTUA) will include descriptive questions as well as selection questions from 2020. They are added to evaluate examinees’ deep understanding and their abilities to think. It will also foster their attitude to think. It will have a large impact on high school education and all K12 education. More descriptive questions will be employed in daily exercises, exams, semester exams and so on. Therefore, computer-assisted marking of descriptive answers is attracting attention to make marking of descriptive answers more efficient and more reliable.

NCTUA will employ mark sheet (paper) for the time being, since it is difficult to employ any electronic devices without troubles for 500,000 applicants concurrently all over Japan. However, tablets and electronic papers will be used for small-scale exams so that online methods from tablet input must be studied but offline methods from scanned input are essential for large-scale examinations.

We focus on clustering handwritten mathematical expressions (HMEs) for computer-assisted marking since mathematical expressions are typical descriptive answers. Examinees’ understanding and problem-solving abilities are most clearly revealed.

We must solve several problems, however, to make computer-assisted exam marking useful. First, HME
recognition is one of the most difficult handwriting recognition problems. We must be able to cluster them even their recognition is fragile. Secondly, there are many clustering algorithms which come from many different approaches. We need to choose a clustering mechanism suitable for the nature of data such as the number of clusters, the data distribution, etc. Thirdly, due to the enormous ambiguity in HMEs, clustering math answers should be employed in an interactive marking systems where the markers could verify and regrade the “impure” clusters.

Fourthly, we need a measure to evaluate how “well” a clustering method supports markers in marking process or to compare the performance of two methods with each other.

In the rest of this paper, Section II presents an overview of related work. Section III describes the features we use in this study and their combination. Section IV propose a cost function for evaluating clustering. Then, section VI describes the experiments and results. Lastly, section VII summarizes this work and discuss future works.

II. RELATED WORKS

We already mentioned automatic essay scoring and autograding of programing. We focus on the methods to judge correctness of answers. There are only a few publications yet.

S. Basu et al. proposed a clustering approach for English short answer grading [8]. Their approach uses k-means to group student responses into clusters and subclusters, then they automatically mark clusters and subclusters by means of predefined answer keywords. Finally, the system allow teachers to read, mark, and provide feedback on those groups of answers at once. Note that the number of clusters and sub-clusters is fixed, which is inflexible when the actual number of answer clusters is different from the fixed number.

M. Brooks et al. proved that clustered interface is superior in terms of ease of use and overall effectiveness [9]. However, there are only a few systems focused on handwritten patterns. Moreover, the number of existing systems for mathematical examinations is even fewer.

A. Singh et al. developed a grading system which allows teachers mark faster by using a dynamic rubric [10]. During the grading process, the system allows the teachers to add items to the rubric at any moment. In practice, this system proves its efficiency in reducing the marker’s time and effort significantly.

III. MULTI-LEVEL FEATURES

Low-level pattern features can be employed for clustering. They are free from HME recognition accuracy but they are not robust to the various ways to write an HME. Features produced from several levels of recognition, such as recognition candidates of symbols produce useful distinctive information although recognition results are fragile due to the immaturity of HME recognition.

A promising solution to this problem is to extract features from HMEs at multiple levels of recognition: low-level directional features, bag-of-symbols, bag-of-relations and bag-of-positions.

A. Directional features

Directional feature extraction has been established for offline character images [11], there are four main steps in the process: nonlinear normalization, directional decomposition, Gaussian blurring, and sampling. Firstly, the HME image is normalized to the size of 64×64 by using nonlinear normalization. Secondly, eight-direction planes are used corresponding to eight chain code directions in order to compute the magnitude and direction of gradients in the normalized image. Then, each plane is divided into 8×8 regions. A low-pass Gaussian filter is applied for all the regions with some overlap to enhance the robustness to positional distortions. Finally, all the directional magnitudes of all the regions and all the planes are concatenated into a single vector.

In this work, we normalize the size of HMEs to the average size of the expressions for the answer instead of normalizing them to a fixed size such as 64×64. Assume the size is the height $H \times$ the width $W$. Then, we decide how many partitions to apply. For a given image of the size $H \times W$, we calculate the average height and width of the connected components within the image (denoted as $H_{comp}$ and $W_{comp}$ respectively), then we calculate the number of regions $M \times N$ as shown in (1).

$$M = \frac{H}{H_{comp}}, N = \frac{W}{W_{comp}}$$

Because the size of feature vectors need to be consistent, we choose the largest number of sub-regions for HMEs answered for the same question.

B. Bag-of-symbols

Bag-of-symbols is one of the most popular features employed for clustering. Symbols in a mathematics examination could be digits, letters, operators, fence symbols, etc. In this work, we assume an HME includes mathematical symbols which are mentioned in [12]. Hence, the bag-of-symbols is a 101-dimensional vector. Instead of using the occurrence counts of the symbols, however, we allow the features take some “fuzzy” values. We employ convolutional neural network (CNN) which has been achieved much success in recognizing handwritten mathematical symbols [13] [14]. We apply CNN for each component in the image to obtain the vector of probabilities of symbol classes. Then, we add all the vectors to form the feature vector of an entire HME image.

For symbols which are composed of multiple components, we redefine a set of prior knowledge rules for combining them. For example, we define a pair of the “.” signs could form the “=” sign. If these components together are recognized as a single symbol by CNN with a probability higher than a threshold, these components are reclassified as the symbol. Each rule is associated with a weight set manually because some rules are more common than the others.
C. Bag-of-relations

Bag-of-relations represents how many each type of relation occurs in a HME and it is computed from those of each component. Bag-of-relations of each component counts frequencies of its relations with its neighbor components. We divide two-dimensional space into nine regions which are originated from the component under consideration and numbered as shown in Fig. 1. Each region represents a specific relation between components, for example, the region “0” represents the inside relation (which appears in root expressions), the region “1” represents the above relation, etc.

For each extracted component, we determine its neighbor components and the relations by means of xy-projection. In order to identify the neighbors of the component under consideration, we examine the bounding box of these components. Two components are neighbors to each other if there is no other component whose bounding box overlaps the line connecting the centers of their bounding boxes (Fig. 2). Fig. 3 shows an example of extracting the Bag-of-relations from an HME image. For each component, we produce a Bag-of-relations feature vector. After extracting the component’s Bag-of-relations, we calculate the cumulative summation of the components’ for forming the final feature vector of the pattern.

D. Bag-of-positions

Bag-of-positions represents where components exist in a HME image. We extract the bounding box of an input HME and divide it into uniform bins of rectangles (MxN bins).

In this work, the grid dimension is chosen based on the structure of the true answer. The number of vertical row is determined by the number of vertical relations (8, 1, 2, 4, 5, 6) plus one and the number of horizontal columns is determined by that of horizontal relations (2, 3, 4, 6, 7, 8) plus one. If “=” exists, one more column is added. If multiple rows include horizontal relations, the maximum number of horizontal relations is chosen. Fig. 4 shows an example. It is vertically split into 3 rows according to two vertical relations and horizontally split to 5 according to a horizontal relation between “=” and the fraction bar, two horizontal relations in both the numerator and the denominator, one for the right side of “=” and one for the “plus one”. For each component, we compute the distribution of the component’s pixels in each bin by (2):

\[
P(G, C) = \frac{\text{# pixels of C inside G}}{\text{# pixels of component C}}
\]

Then, we apply the Gaussian Filter to each bin and obtain the final feature vector as the cumulative summation of all the components.

\[
P = (\Sigma_c P(G_{11}, C), \Sigma_c P(G_{12}, C), \ldots, \Sigma_c P(G_{NM}, C)) \quad (3)
\]

where \(G_{nm} = (1 \leq n \leq N, 1 \leq m \leq M)\) is the bin at \(n^{th}\) row and \(m^{th}\) column.

E. Normalized weighted combination approach

Combining multiple features is expected to form more robust features for clustering HMEs. One of the most popular ways is using weighting approach. The advantage of this approach is it does not require the feature vector size to be the same. However, the difference among value ranges is a considerable problem because of the gross influence of a specific feature type.

In order to eliminate this problem, we normalize the feature vectors before the clustering process. For each feature type, we normalize all feature vectors in the feature space by the inverse of the average distance between two elements. Given two HME patterns, \(P_1\) and \(P_2\), we calculate the distances of the four feature types and use weights to combine them as shown in (4):

\[
d = \Sigma_i w_i d_i(n_i f_1(P_1), n_i f_1(P_2))
\]

where \(w_i\) is the weight corresponds to feature \(f_i\) and \(d_i\) is the distance mapping defined on the feature space \(f_i\), \(n_i\) is inverse of the average distance between two elements in the feature space \(f_i\).

IV. INCREMENTAL REFINEMENT APPROACH

In practice, it usually needs a marker to verify and regrade “impure” answers within a cluster during the marking process where “impure” answers are the answers which are not assigned to the most frequent class in that cluster. If an “impure” answer is detected, it is reallocated to a separated set by the examiner. This interaction must be reflected to the further clustering. For the remaining unmarked clusters, an element is moved to one of the newly created sets if its distance to the centroid of that new set is shorter than to the centroid of that cluster. The algorithm is presented in Algorithm 1. Fig. 5 illustrates the algorithm. Here, we define the distance between a point and a cluster to be the distance from the point to the centroid of that cluster. The distance from a point to a set of points is the

\[
\text{Distance to Cluster} = \text{Minimum distance between point and all points in cluster}
\]
minimum distance from that point to each point in the set points.

Algorithm 1

Input: Set of pairs of clusters and centroids \( W, U = \{ (w_1, u_1), (w_2, u_2), \ldots, (w_k, u_k) \} \).
1. Initialize the set of “impute” subsets of samples \( S = \{ \emptyset \} \).
2. Choose an initial unmarked cluster \( w_i \) and show it to an examiner.
3. Get the marking results for that cluster in which \( M_i \) is the set of major answers and \( m_{i1}, m_{i2}, \ldots, m_{ip} \) are the subsets of minor answers.
4. Append the subsets \( m_{i1}, m_{i2}, \ldots, m_{ip} \) into \( S \).
5. For each unmarked clusters \( w_k \), \( w_k \in W \), \( U \)
a. For each data point in that cluster \( p \in w_k \)
   i. Find the subset \( m_{id} \) which is the closest subset to the data point \( p \).
   ii. If distance between \((p, m_{id})\) < distance set \((p, u_i)\)
       \( m_{id} = m_{id} \cup \{ p \} \).
6. Repeat step 2 until all answers have been marked

We use the within-cluster mean of squared distance to choose the cluster for marking. For each pattern \( p_j \) in a cluster \( w_i \) with its centroid \( u_i \), the within-cluster mean of squared distance is calculated as in (3). Then, we choose the initial cluster whose WCMSD is the smallest.

\[
WCMSD(w_i) = \frac{1}{|w_i|} \sum_{i=1}^{l} ||p_i - u_i||^2
\]

V. COST FUNCTION

There are many measurements such as purity, rand index, normalized mutual information, homogeneity and completeness for evaluation of clustering. The rand index views clustering as a series of decisions, while normalized mutual information, homogeneity and completeness are based on entropy.

![Image](https://via.placeholder.com/150)

Fig. 5. Incremental refinement approach

In clustering-based marking system, we assume that for each cluster, a marker will mark the major set of answers collectively and pick up the minor ones for manual marking separately. Hence, the purity is basically the most appropriate measurement for ME clustering problem.

Assume that there is a set of clusters \( W = \{ w_1, w_2, \ldots, w_k \} \) with \( K \) being the number of clusters and we have a set of classes \( C = \{ c_1, c_2, \ldots, c_J \} \) with \( J \) being the number of classes. To compute the purity, each cluster is assigned to the class which is most frequent in the cluster called major class. Then, for a cluster \( w_k \) in \( W \), let \( M_k \) is the set of samples in the major class \( c_j \) called major set of this cluster. The number of samples which are correctly assigned in this cluster is the number of those in the major set:

\[
|M_k| = \max_{1 \leq j \leq J} |w_k \cap c_j|
\]

Then, the purity is the accuracy of this assignment on the whole dataset divided by the total number \( N \) of samples as shown in (7):

\[
Purity(W, C) = \frac{1}{N} \sum_{k=1}^{K} |M_k|
\]

High purity is easy to achieve when the number of clusters is huge. For example, if \( K \) equals to \( N \), we obtain the purity of one. Thus, the purity itself does not show the clustering quality.

We propose a cost function in order to evaluate the effort of the marker during the marking process. Our cost function augments the deficiency of the purity by considering both the purity and the number of the clusters. Given a cluster of answers, if there exist more than two different sets of answers, the marker tend to select out the minor answers and retain the major one. As a result, the major answers just need a single marking operation while the minor ones require the same amount as the manual approach.

Assuming that the marker takes the marking and verification cost (in terms of time) per answer, let \( T \) be the marking cost (in terms of time). The verification cost is expected to be lower which is denoted as \( \alpha T \) (\( 0 \leq \alpha \leq 1 \)). Considering the cluster \( w_i \) which is separated into major answer set \( M_i \) and minor sets, the marking cost of cluster \( w_i \) is calculated as (8):

\[
\text{cost}(w_i, C) = C_{\text{marking}} + C_{\text{verification}} = (1 + |w_i| - |M_i|) \times T + \alpha |w_i| \times T
\]

where \( C_{\text{marking}} = (1 + |w_i| - |M_i|) \times T \) is the marking cost by marking the single major set and \( |w_i| - |M_i| \) samples in the minor sets, and \( C_{\text{verification}} = \alpha |w_i| \times T \) is the verification cost for all answers in the cluster \( w_i \).

Hence, the cost of the whole dataset is given in (9):

\[
\text{cost}(W, C) = \sum_{w \in W} \text{cost}(w_i, C)
\]

\[
= (|W| + N - \sum_i |M_i|) \times T + \alpha NT
\]

We can rewrite (9) using the purity term as follows:

\[
\text{cost}(W, C) = \sum_{w \in W} \text{cost}(w_i, C)
\]

\[
= (|W| + (1 - Purity(W, C) + \alpha) \times N) \times T
\]

For the simplification purpose, we ignore the verification term which is not affected by the quality of clustering. Let’s rewrite the cost function with removing the verification cost as shown in (11):

\[
\text{SimplifiedCost}(W, C)
\]

\[
= (|W| + (1 - Purity(W, C)) \times N) \times T
\]

Proposition: \( |C| \leq \text{SimplifiedCost}(W, C) \leq NT \)
Proof:
From the (7) and (11) we obtain (12):
\[
\text{SimplifiedCost}(W, C) = (|W| + N - \sum_{k=1}^{K} |M_k|) \times T
\]  

(12)

Firstly, we prove that the minimum value of the cost function is \(|C|T\). Since the purity cannot exceed 1, we get:
\[
N - \sum_{k=1}^{K} |M_k| \geq 0 
\]

(13)

We consider two cases:
1) If \(|W| \geq |C|\), we add \(|W|\) to the both sides of (13):
\[
|W| + N - \sum_{k=1}^{K} |M_k| \geq |W| \geq |C|
\]

(14)

2) If \(|W| < |C|\), there exists \(|C| - |W|\) classes that are not assigned to any cluster. Hence, there are at least \(|C| - |W|\) samples which belong to minor sets. The number of samples which belong to minor sets is calculated as \(N - \sum_{k=1}^{K} |M_k|\). Therefore, we obtain (15):
\[
N - \sum_{k=1}^{K} |M_k| \geq |C| - |W| \Rightarrow |W| + N - \sum_{k=1}^{K} |M_k| \geq |C|
\]

(15)

The equation is obtained when the clustering algorithm produces a perfect clustering result (purity is 1) and the number of clusters is exactly equal to the number of classes.

Secondly, we prove the maximum value of the cost function is \(NT\). In each cluster, the number of samples in the major set is greater than or equal to 1. Hence, the total number of samples which belong to the major set is greater than or equal to the number of clusters. We obtain (16):
\[
\sum_{k=1}^{K} |M_k| \geq |W| \Rightarrow |W| - \sum_{k=1}^{K} |M_k| \leq 0
\]

(16)

Add \(N\) to the both sides of (16), we obtained (17):
\[
|W| + N - \sum_{k=1}^{K} |M_k| \leq N
\]

(17)

The equation is realized when the number of clusters equals to the number of samples (\(|W| = N\), i.e., each sample belongs to its own cluster).

From Proposition 1, we normalize the value range into the interval of \([0,1]\) by dividing it by \(NT\). Our final cost function is (18):
\[
f(W, C) = \frac{|W|}{N} + (1 - \text{Purity}(W, C))
\]

(18)

Note that in the worst case, \(f\) approaches to 1, implying that the marking cost approaches to the manual marking cost. In the HME clustering problem, we try to minimize this cost function with respect to the number of clusters \(K\) and the purity.

VI. EXPERIMENTS AND RESULTS
A. Dataset
In this work, we used the dataset 1 which was collected from 23 students. Each student answered 22 questions. In each question, each student wrote three answers which consist of one correct answer and two incorrect ones. Hence, the total number of samples is 1518. When running experiments, we applied clustering HME answers question by question and we reported the average purity across these questions.

In order to evaluate the performance on a dataset with a larger number of different answers, we applied incremental refinement on the dataset 2. It consists of 3,150 HMEs for 50 different MEs written by 21 students, with each student written HMEs on three different interfaces: without any guiding line, with a center line and with center, top and bottom lines.

B. Experiments on Normalized weighted combination
Firstly, we evaluate the clustering performance of the feature types individually. We used k-means++ which is a popular clustering algorithm to evaluate the clustering performance. For each type, we set \(k\) (the only parameter of k-means++) equal to the number of classes for evaluating its discrimination ability fairly. The average purity is presented in Table I. The bag-of-symbols shows the highest purity on both dataset 1 and dataset 2 (0.7642 ± 0.1779 and 0.8587, respectively). We also compared bag-of-positions without and with Gaussian filter. Their averages are similar but the deviation with Gaussian filter is much smaller so that we use Gaussian filter for the rest of the evaluation.

We used 5-fold cross-validation on Dataset 2 in order to estimate the purity of the normalized weighted combination approach. The estimated purity is 0.8748 ± 0.0421. This result is better than any single type of features.

<table>
<thead>
<tr>
<th>Feature type</th>
<th>Dataset 1</th>
<th>Dataset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directional features (Dir)</td>
<td>0.6465 ± 0.1835</td>
<td>0.7803</td>
</tr>
<tr>
<td>Bag-of-symbols (BoS)</td>
<td>0.7642 ± 0.1779</td>
<td>0.8587</td>
</tr>
<tr>
<td>Bag-of-relations (BoR)</td>
<td>0.5743 ± 0.1171</td>
<td>0.3479</td>
</tr>
<tr>
<td>Bag-of-positions (BoP)</td>
<td>0.5572 ± 0.0098</td>
<td>0.2010</td>
</tr>
</tbody>
</table>

C. Experiments on Incremental refinement
In order to evaluate the performance of our proposed incremental refinement approach, we compared it with the clustering-based approach with respect to the number of clusters. We used the normalized weighted combination approach for this experiment.

The first experiment is on the dataset I, treating samples totally. The actual number of classes (different answers) is 62. The result is presented in Table II.

<table>
<thead>
<tr>
<th># clusters</th>
<th>Purity</th>
<th>Marking cost (eq. (13))</th>
<th>Cost reduction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Clustering- based approach</td>
<td>Incremental refinement</td>
</tr>
<tr>
<td>40</td>
<td>0.4944</td>
<td>0.5321</td>
<td>0.5221</td>
</tr>
<tr>
<td>45</td>
<td>0.5135</td>
<td>0.5162</td>
<td>0.4917</td>
</tr>
<tr>
<td>50</td>
<td>0.5631</td>
<td>0.4699</td>
<td>0.4613</td>
</tr>
<tr>
<td>55</td>
<td>0.5744</td>
<td>0.4620</td>
<td>0.4455</td>
</tr>
<tr>
<td>60</td>
<td>0.6107</td>
<td>0.4298</td>
<td>0.4210</td>
</tr>
<tr>
<td>65</td>
<td>0.6226</td>
<td>0.4204</td>
<td>0.4197</td>
</tr>
<tr>
<td>70</td>
<td>0.6504</td>
<td>0.3959</td>
<td>0.3873</td>
</tr>
<tr>
<td>75</td>
<td>0.6319</td>
<td>0.4177</td>
<td>0.4171</td>
</tr>
</tbody>
</table>
In this dataset, the marking cost of the incremental refinement is reduced from the clustering-based approach. As the number of clusters is less than or approximately to the true number of classes, the purity is not high, then the incremental refinement reduces the marking cost more efficiently (usually greater than 1.8%).

In contrast, as the number of clusters is significantly more than the true number of classes, the cost is not so much reduced by the incremental refinement.

We also applied the incremental refinement on each question individually. Table III presents the result. The actual number of classes is three in each question. Again, if the number of clusters is too many relative to the true number of classes, the incremental refinement approach does not reduce the marking cost.

**TABLE III. COST REDUCTION OF INCREMENTAL REFINEMENT ON DATASET 1 QUESTION BY QUESTION (ACTUAL CLASSES ARE 3)**

<table>
<thead>
<tr>
<th># clusters</th>
<th>Cost reduction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>6.0</td>
</tr>
<tr>
<td>3</td>
<td>-0.8</td>
</tr>
<tr>
<td>4</td>
<td>-1.4</td>
</tr>
<tr>
<td>5</td>
<td>-2.2</td>
</tr>
<tr>
<td>6</td>
<td>-1.8</td>
</tr>
</tbody>
</table>

Finally, we applied the incremental refinement on the dataset 2 and presented the result in Table IV. If the number of clusters is much higher than the actual number of classes, however, the marking cost of the incremental refinement is usually higher than the clustering-based approach. This is because the purity is improved when the number of clusters increases. At a certain point, however, the purity is high enough so that the minor samples tend to be outliers. These outliers affect the incremental refinement. In practice, even though the incremental refinement does not work well when the purity is very high, the marking cost is still acceptable.

**TABLE IV. MARKING COST ON DATASET 2 (ACTUAL CLASSES ARE 50)**

<table>
<thead>
<tr>
<th># clusters</th>
<th>Purity</th>
<th>Marking cost (eq. (13))</th>
<th>Cost reduction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0.7759</td>
<td>0.2368 - 0.2232</td>
<td>5.8</td>
</tr>
<tr>
<td>45</td>
<td>0.7775</td>
<td>0.2368 - 0.2244</td>
<td>5.2</td>
</tr>
<tr>
<td>50</td>
<td>0.8371</td>
<td>0.1787 - 0.1768</td>
<td>1.1</td>
</tr>
<tr>
<td>55</td>
<td>0.8552</td>
<td>0.1622 - 0.1486</td>
<td>8.4</td>
</tr>
<tr>
<td>60</td>
<td>0.8667</td>
<td>0.1524 - 0.1486</td>
<td>2.5</td>
</tr>
<tr>
<td>65</td>
<td>0.8937</td>
<td>0.1270 - 0.1254</td>
<td>1.2</td>
</tr>
<tr>
<td>70</td>
<td>0.9486</td>
<td>0.0737 - 0.0740</td>
<td>-0.4</td>
</tr>
<tr>
<td>75</td>
<td>0.9695</td>
<td>0.0543 - 0.0556</td>
<td>-2.3</td>
</tr>
</tbody>
</table>

**VII. CONCLUSIONS**

In this study, we evaluated the performance of clustering by a variety of features from offline handwritten mathematical expressions. We also evaluated their combination with improved performance. In order to reflect the marker’s effort on marking mathematical answers, we proposed a cost function with respect to the number of clusters and the purity of the clustering result. We proposed an incremental refinement approach as a heuristic solution for minimizing the proposed cost function. The result shows that our incremental refinement approach utilizes the marker’s marking information to reduce their marking effort when the number of clusters is not too high relative to the number of classes.

**REFERENCES**


Bag-of-features for clustering online handwritten mathematical expressions

Huy Quang Ung, Vu Tran Minh Khuong, Anh Duc Le, Cuong Tuan Nguyen, Masaki Nakagawa
Department of Computer and Information Science
Tokyo University of Agriculture and Technology
Tokyo, Japan
{ungquanghuy93, vtmkuong, leducanh841988, ntcuong2103}@gmail.com, nakagawa@cc.tuat.ac.jp

Abstract—This paper presents clustering of online handwritten mathematical expressions (HMEs) to help human markers to mark them efficiently and reliably. We propose bag-of-features from online handwritten mathematical expressions. It consists of 6 levels of features from low-level pattern features to high-level symbolic and structural features which are obtained from recognizing online HMEs. Experiments are conducted on our dataset. The best clustering result is 0.9185 for purity, which is obtained by applying the combination of both low-level and high-level features with the k-means++ algorithm.

Keywords—clustering, handwritten mathematical expressions, computer-assisted marking, bag-of-features, online patterns

I. INTRODUCTION

Nowadays, touch-based and pen-based devices are getting more and more popular as learning media. Children and students use them to read textbooks and make exercises. They are suitable to write mathematical expressions (ME) for learners. Thanks to recognition interfaces of online handwritten mathematical expressions (HMEs), it could be better for learners to express MEs than using editors such as Microsoft Equation Editor, MathType or high-quality typesetting system like LaTeX.

The research on recognition of HMEs started since the late 20th and early 21st century. It is recently very active due to the high demands for applications on tablets. The competitions on recognition of HMEs have been continued under the series of CROHME [1] with the result of the recognition performance increasing.

By handwriting recognition, these devices let learners to answer questions by writing answers rather than selecting answers by keyboard or mouse. Descriptive questions can far better test learners understanding and abilities to think than selection type questions for which learners can select correct answers by chance. Moreover, descriptive questions foster learners to think rather than select. With this progress, many e-learning interfaces based on pen-based devices have been researched [2] [3] [4] and employed for practical applications.

If the recognition result is verified and confirmed by a learner, HME recognition can be incorporated into self-learning and e-testing applications. Although a learner have to make additional work, the learner can get the immediate feedback. If it is not verified, however, it is better utilized for computer assisted marking. Examinees only have to write answers as on paper exams and examiners can get the benefit of computer processing. One of the most promising computer assistances is to cluster answers. The goal of this approach is to group similar answers that could be graded into the same score. If answers are well clustered, they can be marked efficiently and marking errors and fluctuations will be decreased. Since the final marking is made by human markers, examinees’ anxieties will be also decreased. Teachers can spend more time for explanation to learners which is produced by decreased burden of marking exams.

In fact, the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT) has decided that National Center Test for University Admissions (NCTUA) will include descriptive questions from 2020 to test the deep understanding and the abilities to think as well as to foster thinking rather than selecting. It will have a large impact on high school education and all K12 education. More descriptive questions will be employed in daily exercises, exams and semester exams and so on. Therefore, computer-assisted marking of descriptive answers is attracting attentions to make marking of descriptive answers more efficient and more reliable.

Although NCTUA will employ mark sheet (paper) for the time being, since it is difficult to employ any electronic devices without troubles for 500,000 applicants concurrently all over Japan, tablets and electronic papers will be used for small scale exams so that online methods from tablet input must be studied.

We must solve several problems, however, to make computer-assisted exam marking useful. First, ME recognition is one of the most difficult handwriting recognition. We must be able to cluster them even their recognition is fragile. Secondly, there are many clustering algorithms which come from many different approaches. We need to choose a clustering mechanism suitable for the nature of data such as the number of clusters, the data distribution, etc.
Extensive research has been carried out on essay assessment [5] [6] [7] and even handwritten essay scoring was reported [8]. There are not so many publications yet. Basu et al. proposed a method for clustering answers for English short answer grading [9]. They trained a similarity metric to calculate a distance between two different answers with logistic regression, then, employed a modified k-Medoids and LDA clustering algorithm for forming clustered and subclusters of answers. Owing to the clustering, graders just score each group by using one operation and also give a feedback for each group of answers at the same time. As a result, cost of grading process is reduced in their experiment. Brooks et al. continued the approach to design a cluster-based interface [10], and showed effectiveness of the interface in term of allowing graders to make high-quality marking and give a feedback for clusters and sub-clusters of answers at once. Moreover, it could reflect levels of examinee’s understanding to modify current teaching method.

For assessment of handwritten paper-based work, A. Singh et al. [11] introduced a web-based system which allows students to upload their scanned assignments and teachers to mark them and give a feedback for each kind of the answers as well as categorizing them. Following to user reports for 4 years in use, it has shown the effectiveness of the system in speed, consistency and flexibility.

Although the problem of the clustering of online HMEs has not been researched well up to now, we can generalize our issue to be a problem of extracting features from sequential data for clustering. In this issue, there is similar work on natural language processing, that is to construct a vector space for not only clustering. In this issue, there is similar work on natural language processing, that is to construct a vector space for not only clustering. In this issue, there is similar work on natural language processing, that is to construct a vector space for not only clustering.

A conventional method to address this problem is bag-of-words [12]. However, a drawback is that the word order is lost, and hence, it could not guarantee uniqueness. A better solution proposed by Le and Mikolov [13] is to construct a continuous vector space, where semantically similar sequences are mapped to nearby points, by learning to predict a next word in a word sequence. Besides, these vectors can be applied to machine learning algorithms such as k-means, support vector machine, etc.

To cope with ambiguity of HMEs and fragility of HME recognition, we extract features from low-level pattern features to high-level symbolic and structural features which are obtained from recognizing online HMEs. We employ bag-of-features composed of low-level directional features and recognition-based features that consist of bag-of-symbols, bag-of-relations, bag-of-positions, position-based bag-of-symbols and position-based bag-of-relations. Low-level features are free from recognition accuracy but not robust to various ways to write an ME. Features from several levels of recognized HME may be fragile due to the immaturity of HME recognition but they can provide useful distinctive features. We also consider a method for combining these types of features to improve the performance and overcome the above-mentioned problem of bag-of-words model. Finally, a clustering method can be applied to these types of features for evaluation.

The rest of this paper is formulated as follows: Section II introduces our types of features in details. Section III presents problems related to the cost of a clustering-based marking. Section IV presents experiments for evaluating proposed features, and results. Finally, Section V gives some conclusions and discussions.

II. MULTI-LEVEL FEATURES

We propose two main types of features: low-level features and recognition-based features as shown in Fig. 1.

A. Low-level feature

As low-level features, we extract directional features. In order to avoid the instability of online sampling of handwriting such as stroke order variations, we convert online handwritten patterns to offline patterns and extract image-based directional features of all strokes. We employ a version of directional features (Dir) implemented by Phan et al. [14]. We apply the Sobel operator to compute gradient vectors on the image that is normalized by a non-linear normalization method, then, decompose these vectors into 8 chain code directions to obtain 8 planes of corresponding directions. After that, we divide these planes into equal $D_{ver} \times D_{hor}$ regions and apply a Gaussian filter for blurring. Finally, by taking a Cartesian product of all columns and all rows, we can obtain a feature vector of the length $D_{ver} \times D_{hor} \times 8$.

B. Recognition-based features

The recognition-based features are obtained from a symbol relation tree (SRT) produced by HME recognition, which represents symbols and relations in an HME as shown in Fig. 2. It consists of 4 types of features: bag-of-symbols, bag-of-relations, bag-of-positions, position-based bag-of-symbols, and position-based bag-of-relations. Fig. 1 shows an example.

According to the CROHME 2016 competition for online HME recognition [1], the recognition rates of HMEs are not high enough (at most 67.65%) since all the symbols and relations must be correctly recognized, but accuracies of segmentation and classification of both symbols and spatial relationships are more than 90%. These results express that it is possible to accept results of the segmentation and recognition of symbols and relations. Based on this understanding, we introduce recognition
based features. We can capture these features from several candidates of recognition, but we select the top candidate to extract features.

1) Bag-of-symbols

Bag-of-symbols (BoS) represents how many each symbol appears in a HME. Then, we put it into a vector representing the number of occurrences in an available list of symbols as shown in Fig. 2.

2) Bag-of-relations

According to [15], for each HME, we have 6 types of a relation between symbols: horizontal, superscript, subscript, over, under, and inside. Similarly with BoS, bag-of-relations (BoR) represents how many each type of relation occurs in a HME in the form of a vector.

3) Bag-of-positions

While BoS and BoR reflect information about symbols and relations, bag-of-positions (BoP) represents information of symbols’ positions based on a regional division on the SRT. For this idea, the SRT is divided into \( M \times N \) regions, each of which contains at most one symbol. This division also reflects the relationship among symbols. These parameters, \( M \) and \( N \), could be set as the size of SRT. However, since the size of these trees are variant and depend on each ME, we could not generate the feature vectors in the same dimensionality. In order to address this issue, we can use the size of the largest SRT in the answers to normalize other HMEs as shown in Fig. 3. Then, the regions containing a symbol is marked 1 otherwise 0 and expressed by a matrix \( P_{M\times N} = (p_{ij}) \) with \( p_{ij} \in \{0,1\} \). To make these features more robust, we apply a Gaussian filter of \( 3 \times 3 \) for bluring. Finally, a feature vector is formed by taking a Cartesian product of all columns and all rows for each type of features.

4) Position-based bag-of-symbols and bag-of-relations

In BoS and BoR, the captured information is discrete and does not express obviously the structure of an HME. Besides, BoP just provides features of symbols’ positions without mentioning the type of symbols. Therefore, we propose an approach to add positional information while extracting symbols and relations. Firstly, we divide the SRT into \( M \times N \) regions as presented in the part of BoP. Then, from each region, we extract BoS and BoR with applying a Gaussian filter to make information between a region and its neighbours more robustly as shown in Fig. 4. We extend the region to its neighbors or making zero padding so that we get \( 3 \times 3 \) regions. We convolute the values of \( 3 \times 3 \) regions with the \( 3 \times 3 \) Gaussian mask and sum all the values for the value the region. We finally obtain 2 types of feature vectors called: position-based bag-of-symbols (PbBoS) and position-based bag-of-relations (PbBoR) by taking a Cartesian product of all columns and all rows for each type of features.

C. Combination method

A single type of bag-of-features may not contain enough information for clustering, so that we combine them to improve the performance. The simple idea is to concatenate multiple feature vectors into a single vector. However, it might be ineffective, because different feature vectors have different meaning and belong to separate spaces. We could not treat them as the same. In this paper, we use weighting parameters to
optimize their performance, so that a distance between two samples is calculated according to (1).

\[
\text{Dis}(HME_1, HME_2) = \sum_{i=1} \alpha_i d(f_i(HME_1), f_i(HME_2))
\]

(1)

Where \( HME_1 \) and \( HME_2 \) are two HMEs to compute the distance, \( f_i \) are a type of features, \( i \) is the number of types of features, \( d(v_1, v_2) \) is Euclidean distance between two vectors \( v_1 \) and \( v_2 \), and \( \alpha_i > 0 \) is a weighting parameter for each type of features satisfying \( \sum_{i=1}^{i} \alpha_i = 1 \). These parameters are determined by applying the enumeration method with \( \alpha_i \in \{0.1, 0.2, 0.3, ..., 0.9\} \).

III. Clustering-Based Marking

By clustering-based marking systems, a marker will mark the major set of answers for each cluster collectively and pick up the minor ones for manual marking separately.

Hence, the cost of the marking process depends on how many samples are the same answer and how few different answers are included in each cluster. We use purity as a measurement for evaluating result of the clustering task. This measurement is calculated according to (2).

\[
Purity(G, C) = \frac{1}{H} \sum_{k=1}^{K} \max_{1 \leq i \leq J} |g_k \cap c_i|
\]

(2)

where \( K \) and \( J \) are the number of clusters and classes respectively, \( H \) is the number of samples, \( G = \{g_1, g_2, ..., g_K\} \) is a set of obtained clusters, \( C = \{c_1, c_2, ..., c_J\} \) is a set of classes.

It is worth noting, however, that high purity is easy to achieve when the number of clusters is huge. For example, if \( K \) equals to \( H \), we obtain the purity of one. Thus, the purity alone does not show the clustering quality. The number of clusters should not be too small. Moreover, if we consider the marker’s work to view clustered answers, there must be an appropriate number to display answers so that the marker can confirm the correctness or incorrectness. They are important factors to evaluate the clustering based marking method. Nevertheless, purity is the most important factor so that we employ purity to evaluate experiments.

IV. EXPERIMENT AND RESULTS

This section presents evaluation of the proposed clustering method.

A. Dataset

For evaluation, we employ a clustering dataset that is collected from 21 students. Each student writes 3 times of 50 HMEs on 3 kinds of writing interfaces (without any guiding line, with a center line, and with center, top and bottom lines) respectively. As the result, the total number of online HMEs is 3150 samples and they belong to 50 classes. This database contains common symbols that belong to 101 classes used in Task 2a of CROHME 2016 competition [1].

B. Recognition system

We employ the online HME recognition system by Le et al. [16] which joined the CROHME 2016 competition. Within a group that trained their systems only by the provided database, this system achieved the best recognition rate of 92.27% on the task of classifying isolated math symbols (Task 2a) and the second rank (43.94%) on the task of recognizing formulas, in which the rate of both segmentation and classification of symbols and spatial relationships in each formula is 86.05% and 82.11% respectively. Details of the system are presented on [16].

C. Experimental setting and measurement

1) Experimental setting

We utilize k-means algorithm for the clustering task with Euclidean distance. For evaluation of proposed features, the number of clusters in our experiments is set as the number of classes in the clustering database.

When applying the k-means algorithm, since it is sensitive to initial points and could be easy to fall into local minima [17], initialization method influence strongly on the clustering result. In this research, we make our experiments with three methods as shown below. The aim of this task is to evaluate how well our features work on different conditions of initialization.

Firstly, by randomly picking up seeds, we could see the clustering performance when applying our proposed features in random conditions.

The second method is k-means++ [18], which is a popular variant of the k-means algorithm. The idea of this method is to try to spread out initial points by setting probability for each point in the database during selecting.

The last one is called “ideal” method, in which we assume that we already identify \( k \) points belong to \( k \) classes respectively and use them to set as seeds for the k-means algorithm. This initialization is expected to make the algorithm converge to a global minimum. Although it could not be utilized for real instances, we might be able to evaluate our features in the ideal case.

2) Measurement

We evaluated the performance of the clustering task when applying single types of features and combined features. Since this combination requires the weighting parameters to form the distance metric, we split the dataset into 5 subsets and trained the distance metric by the training set formed by 4 subsets and test the performance to the remaining set, rotated the roles of subsets 5 times and took their average (5-fold cross-validation). For combined features, we used the random initialization method to optimize these parameters in training. Table I shows the optimized parameters and their performance on 5 training subsets when combining Dir, BoS, BoR and BoP (A), as well as Dir, BoS, BoR, PbBoS, and PbBoR (B).

Table II shows purity by all the types of features and their combinations with the three different initializations. Within single types of features, directional features is fairly more robust than the others with random initialization. Moreover, BoS
expresses slightly better results with k-means++ and the ideal condition. The features formed by only extracting information about relations or positions have quite low purity in this experiment. Adding positional information to BoS or BoR, namely PbBoS and PbBoR, does not increase purity alone. Combining BoS, BoR, and BoP or PbBoS and PbBoR again does not increase purity. On the other hand, the combination of Dir with BoS, BoR, and BoP or BoS, BoR, PbBoS, and PbBoR clearly outperforms other single types of features, and limited combinations of two symbolic and structural features, which implies the effect of combining both low-level and high-level features. However, the combination of all proposed features makes the performance of the clustering task decreasing probably because BoP as well as PbBoS plus PbBoR are both positional features and they are somewhat redundant.

TABLE I. OBTAINED PARAMETERS AND THEIR PERFORMANCE ON 5 TRAINING SETS

<table>
<thead>
<tr>
<th>Training set</th>
<th>Purity</th>
<th>BoS (α₁)</th>
<th>BoR (α₁)</th>
<th>BoP (α₂)</th>
<th>Dir (α₂)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.8917</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.7</td>
</tr>
<tr>
<td>2</td>
<td>0.8821</td>
<td>0.7</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.8944</td>
<td>0.4</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>4</td>
<td>0.9183</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>5</td>
<td>0.9170</td>
<td>0.3</td>
<td>0.1</td>
<td>0.1</td>
<td>0.3</td>
</tr>
</tbody>
</table>

TABLE II. AVERAGE RESULTS OF SINGLE TYPES OF FEATURES AND THE COMBINATIONS ON 5 TESTING SETS

<table>
<thead>
<tr>
<th>Types of features</th>
<th>Purity by three different initializations</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Random</td>
<td>k-means++</td>
<td>Ideal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dir</td>
<td>0.7947±0.032</td>
<td>0.8658±0.034</td>
<td>0.9875±0.013</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoS</td>
<td>0.7955±0.032</td>
<td>0.8705±0.015</td>
<td>0.9907±0.004</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoR</td>
<td>0.5954±0.049</td>
<td>0.7581±0.097</td>
<td>0.8100±0.087</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoP</td>
<td>0.6131±0.139</td>
<td>0.7724±0.186</td>
<td>0.7530±0.145</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PbBoS</td>
<td>0.6856±0.058</td>
<td>0.8508±0.008</td>
<td>0.9431±0.040</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PbBoR</td>
<td>0.6195±0.107</td>
<td>0.7548±0.104</td>
<td>0.7916±0.092</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoS + BoR + BoP</td>
<td>0.7186±0.040</td>
<td>0.8567±0.033</td>
<td>0.9832±0.011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PbBoS + PbBoR</td>
<td>0.6540±0.039</td>
<td>0.8074±0.026</td>
<td>0.9410±0.011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dir + BoS + BoR + BoP</td>
<td>0.8757±0.029</td>
<td>0.9042±0.029</td>
<td>0.9958±0.002</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dir + BoS + BoR + PbBoS + PbBoR</td>
<td>0.8575±0.024</td>
<td>0.9185±0.015</td>
<td>0.9950±0.003</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dir + BoS + BoR + BoP + PbBoS + PbBoR</td>
<td>0.8040±0.037</td>
<td>0.8942±0.013</td>
<td>0.9848±0.012</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

V. CONCLUSION

For the purpose of creating a computer-assisted marking system on mathematics based on clustering approach, this paper presented bag-of-features for the clustering task of online HME. These features consist of information of a HME sample from a low-level image-based features to high-level symbolic and structural features obtained from HME recognition. In addition, we also proposed an approach to combine all of them for enhancing the performance of the clustering task. The best clustering result, obtained by combining both low-level and high-level features consisting of Dir, BoS, BoR, PbBoS and PbBoR with k-means++ initialization method on our dataset, is around 0.9185 of purity. The combination of Dir, BoS, BoR and BoP is also comparable although it is slightly simpler. Besides, according to the experiments in ideal conditions of initialization of the k-means algorithm, the performance of the both methods could reach 0.99 of purity.

There remains some works to do. Although we collected HME samples in three different writing interfaces, we must analyze their effects. We should collect more samples of HME answers to validate the results. Moreover, we need to study the evaluation criteria on clustering methods on top of purity.

REFERENCES


A New Roadmap for Evaluating Descriptive Handwritten Answer Type

Palaiahnakote Shivakumara
Faculty of Computer Science and Information Technology
University of Malaya, Kuala Lumpur, Malaysia,
hudenpsk@yahoo.com

Umapda Pal
Computer Vision and Pattern Recognition Unit
Indian Statistical Institute, Kolkata, India.
Email: umapada@isical.ac.in

Tong Lu
National Key Lab for Novel Software Technology,
Nanjing University, Nanjing, China
Email: lutong@nju.edu.cn

Tapabrata Chakraborti
Computer Vision and Pattern Recognition Unit
Indian Statistical Institute, Kolkata, India.
tachak59@gmail.com

Michael Blumenstein
University Technology of Sydney
Australia, email: michael.blumenstein@uts.edu.au

Abstract—In computational pedagogy, a relatively simple OCR system can robustly evaluate objective response types automatically without human intervention. This saves time, cost and man hours. Thus, the next questions becomes whether it is possible to develop an automated system for evaluating descriptive handwritten answers type or not. Recent experience show that human evaluation of long examination responses is quite subjective and prone to challenges like inexperience, negligence, lack of uniformity in case of several evaluators, etc. In this work, we present the roadmap for an automated vision system that evaluates the descriptive answers based on extracting relevant words and finding relationship between the words according to weights of the words. We introduce context features to handle variations of the words written by the different users to estimate the final score.

Keywords—Handwriting analysis, Cleaning document, Text line segmentation, Answer script evaluation.

I. INTRODUCTION

One of the established systems in document image analysis is an Optical Character Recognizer (OCR) for printed documents that has achieved more than 98% recognition rate [1]. OCR engines are now available in different languages in the market for many applications where digitization is a prerequisite [1]. An open research problem is to achieve the same success for recognizing degraded, historical and handwritten documents. This type of handwritten OCR can be used in several real life applications, such as postal automation, bank cheque processing, signature verification etc. However, due to large variations in writing styles (due to different persons, paper/ink quality, double side writing, writer's mood etc.) developing OCR for handwritten text is considered a challenging task in the field of automated document image analysis [2]. Developing an automated robust system for descriptive answer evaluation poses further challenges because of its sensitivity to individual writing and higher complexity, scope, context and length.

There is existing work that evaluates short answers using stroke-based and/or structural features and classifiers for grading [3, 4]. These methods are suitable for short answers and work well for limited cases. Therefore, descriptive answer evaluation requires a new roadmap in order to develop a working vision system. As per our knowledge, there is no single attempt to develop a comprehensive method in that direction till date. The main reasons are (1) large variations in writing styles (2) word extraction from the answers poses several challenges due to non-uniform spacing between the words while writing and touching etc, (3) proving that the extracted word are relevant to evaluate answers written by the different students. One can expect that we have to handle such nuances as synonyms and different ways of expression due to individual difference in thinking, knowledge, writing ability, vocabulary, etc. (4) deciding degree of relevance of answers (5) finding relationship between relevant words to assign weights to assign score, which in turn helps in confirming answers (6) deciding final score based on weights of the words. This is challenging because different students write answers differently which may contain few words, more words or no relevant words. Consequently, as the number of students increases, the complexity of the problem increases. Therefore, there is a major challenge to develop a working system that can be scalable for any number of students with minimum cost and high reliability.

The main objective of the paper is to provide a roadmap for developing an automated system that can evaluate the handwritten descriptive answers without human intervention such that we have an objective uniform evaluation under one scheme. To achieve this goal, we divide the main objective into sub-objectives which will be discussed in details in the proposed methodology section.

II. RELATED WORK

In this section, we review the methods on short answer evaluation as there are no methods currently available for evaluating descriptive answers. Suwanwiwat et al. [5] proposed an off-line restricted set for handwritten word recognition for automated student identification through a short answer question assessment system. The method explores Gaussian grid and modified directional features for recognition. Srivastava and Bhattacharyya [6] proposed a captive short answer evaluator. In this method, natural language concept is used for recognizing words written by students. Suwanwiwat et al. [7] proposed short answer question examination using an automatic off-line handwriting recognition system. This method combines features obtained by water reservoir model and Gaussian grid features for word recognition. Duenas et al. [8] proposed automatic prediction of item difficulty for short answer type questions. The method extracts a large number of
features and then uses a classifier for prediction. Hasanah et al. [9] proposed an information extraction approach for automatic short answer grading. The method uses several matching techniques such as parse tree matching, regular expression matching, Boolean parse matching, Syntactic parse matching, and Semantic word matching for grading. Suwanwiwat et al. [10] proposed novel combined features for a handwritten short answer assessment system. The method is an extension of their previous method by adding more number of features and uses a classifier for recognition of text. Rababah and Taani proposed [11] an automated scoring approach for Arabic short answers to essay questions. This method finds similarity between answers written by students and predefined model answer. For finding similarity, the method uses cosine similarity measure. Based on degree of similarity, the method assigns grade for each question. Srihari et al. [3] proposed automatic scoring of short handwritten essays in reading comprehension tests. The method initially segments text lines and words from each document image. It then proposes fusion of analytic and holistic methods together with contextual processing based on trigrams. Finally, lexicons to recognize handwritten words are derived from the reading passage, the testing prompt, answer rubric and student responses. Srihari and Singer [4] proposed a role of automation in the examination of handwritten items. In this method, integrating developed tools into a unified framework is discussed. They also provide overall advantages, disadvantages and future directions for evaluating essays type answers. Meena and Raj [12] proposed evaluations on descriptive type answers using hyperspace analog to language and self-organizing map. The method explores clusters of Kohonen self-organizing map for evaluation.

In light of the above discussion, it is noted that none of the methods addressed the issues of descriptive answer type evaluation except the framework suggested in [4] and [12]. This shows that the problem is still an open challenge. Most of the methods used handwritten recognition for evaluating short answer assessment. When we increase number of student answers and the number of lines such as answers that span full pages or half pages, these methods may not perform well. Hence, in this proposal, we try to provide a new idea in the form of roadmap to develop a system to evaluate the descriptive answers automatically.

III. PROPOSED ROADMAP

Recent trends suggest the use of deep learning to solve complex tasks across all problems of machine intelligence [13]. This inspires the present roadmap for solving the complex problem of evaluation of descriptive answer type. Based on the proposed roadmap, one can develop vision systems to find viable solutions in the future. It is noted that while writing answers, students usually do some extra rough hand work to remember or recall the concepts or to calculate intermediate results. As a result, a handwritten page contains many unwanted writings that are sketchy with irregular structure. These noisy sketches hinder subsequent steps such as text line segmentation, word segmentation and recognition because noisy sketches may cause touching lines, lose actual shapes of the characters, etc. Therefore, the first step needs to be cleaning the handwritten document image.

Generally, due to time constraints and practice, students write answers without worrying about orientation or the spacing between words and text lines. This makes text line segmentation more complex and challenging. Conventional methods that work based on connected component analysis and binarization based methods may not work well for handwritten document images. So the second step is text line and words segmentation from cleaned handwritten document images.

Extracting relevant words from answers written by different students in different styles without any restrictions of grammar, words and sentences is really challenging for researchers. However, we believe that each question must be associated with a few technical words according to the question. The idea is to spot such technical words, if any, in the descriptive answer. With the help of words in a model answers (schema), it is possible to find the degree of relevancy. Therefore, the next challenge is to spot these words based on degree of relevance with respect to words in model answers.

![Fig. 1. Framework of the proposed roadmap.](image)

In order to understand the descriptive answers, we consider spotted words as seeds. Then the method finds neighboring words in the text line of seed words to find the relationship between seed words and neighbor words semantically based on dictionary, natural language processing concepts, etc. This results in understanding the sentences of descriptive answers. Based on this understanding, we need to assign weights which should help us to give grades. Finally, weight of each sentence is analyzed further to assign the total grade for the whole descriptive answers. In other words, the method should find degree of relevance at text line level to assign the final grade. The schematic diagram of the proposed roadmap can be seen in Fig. 1. The idea to find solutions for each of the above steps is discussed in subsequent sections.
A. Preprocessing for Cleaning Document and Removing Strike out Text

Since our target is to consider descriptive handwritten answers like essay type answers, one can expect side works, such as intermediate calculations before computing final calculations, writing a few keywords to recollect the answers, and strike-out text while writing answers. These are the main causes which prevent achieving good accuracy for subsequent steps, namely, text line, word segmentation, keyword spotting and estimating degree of relevance at word and text line levels. Therefore, it is necessary to remove such noise from the raw handwritten document images. Cleaning handwritten document is not a new problem for the document analysis field [14]. However, most of the methods depend on binarization and structural points, such as end, junction and intersection points appeared in different zones [13]. Since the above cases are not simple to identify as noisy edges or components, it is required to use grayscale images without binarization because these give more details compared to binarization. It is true that direction of pixels of an actual text have regular pattern, whereas unconstrained noisy text which has irregular direction pattern. We need to explore such regular direction patterns along with gradient values, which give low values for noisy text and high values for regular text due to difference in force and speed in writing regular text versus noisy text.

In order to achieve the above distinct direction patterns, we can use simple but effective gradient directional features. This is because of the fact that gradient direction moves perpendicular to edge direction. For actual edges, we get regular pattern with directions, while for noisy text, one cannot expect regular pattern due to irregular writing without any constraints. The same directional features can be used for removing strike out text; the strike over a text provides long run in the same direction. This feature can be used to separate the directions which represent text and strike. The schematic diagram can be seen in Fig. 2 for developing the method to clean handwritten document images.

![Fig. 2. Schematic diagram for cleaning and removal of strike out text.](image)

B. Text Line and Word Segmentation

With typing becoming more common, current students are losing practice in writing. New devices such as smart phone and tablet help us to reduce writing tasks further. As a result, one can expect writing in irregular fashions becoming more common with non-uniform spacing between words and/or lines. Thus segmenting text lines and words is challenging especially from lengthy essays. According to literature review [15], most of the existing methods use foreground information to segment text lines. However, for documents such as answer scripts, where one can come across rulers which confuse baseline and strokes, it is necessary to use background information along with foreground information in gray domain.

To achieve good segmentation, one can explore the same gradient directional features as discussed in the previous section. This is because when there is text, gradient direction can form symmetry due to edges [16]. If there is no text, which is nothing but spacing between the lines and words, the gradient direction may not form symmetry [16]. In addition, where there is text, we can expect high gradient values and where there is no text, we can expect low gradient values. Integration of these two features helps us to identify the space between lines and words. Once we identify the space between text lines, the same symmetry properties can be used for word segmentation. This idea is simple and effective compared to studying foreground or background with binary information. In addition, this idea is independent of scripts and orientations. The logic is shown in Fig. 3, which helps understand the flow of the steps.

![Fig. 3. Text line and word segmentation procedure.](image)

C. Degree of Relevancy for Word Spotting

The previous steps provide segmented words for the descriptive answer type. The real challenge though is to find relevant words which can prove that the written answer is relevant to a particular question because each individual can write different words for the same answer. In addition, instead of writing actual key words, they may write some expanded explanations representing the meaning of the technical words. Therefore, we need to look for finding relevant words first to confirm the written answer is related to the correct question. There is a genre of methods in document analysis to search for specific words without recognition, namely, word spotting. The big question, however, is how to estimate relevancy. It is a fact that every question has a general schema to provide keywords and important sentences, which are ingredients for confirming whether the written answer is relevant for efficient grading. This is called the model answers database. The method uses Identify applicable funding agency here. If none, delete this text box.
words in the model answer database to estimate the degree of relevancy. Usually every answer must contain a few technical keywords. Therefore, it is simple to spot the words that have the maximum similarity between the words of the answer and model answers [17, 18]. This results in seed words which represent the answers of a particular question.

If an answer does not provide any technical words according to the words in model answer, we need to find the degree of relevancy with the help of synonyms chosen from dictionary and other resources. For this situation, we need to explore natural language processing concepts to find relevant meaning of the words which can match with those in the model answer database. The degree can be estimated based on matching a number of pixels or characters [19, 20]. Based on degree of relevancy and the words in the model answer, we need to assign grades which can be used for the final grading. The logic for estimating degree of relevancy at word level with the help of model answers (schema) is shown in Fig. 4.

D. Degree of Relevancy at Text Line level for Final Grading

The idea presented in the previous section provides seed words which represent the correct answers of a particular question. Now, we need to consider seed words as starting words to find the other words of the same sentence in order to understand the meaning of the whole sentence. The method should find relationship between the words of the sentence with the seed. This is possible by exploring context features between the seed and the other words of the sentence. The context can be defined based on the degree of relevancy of neighbor words with the seed word. With the help of context and words in model answers, we can predict the meaning of the whole sentence. This process should continue for all the seed words of all the sentences. After confirming right sentences for the right question, the method should assign weights and this again should be done for all sentences.

Once the method finds the coherence between sentences according to model answers, the weights of seed words should be revised to correct if there is any mistake. At the same time, we need to revise weights at sentence level before the final grading. This is because sometimes, we may not get all the expected words, no words, false words, etc. In this case, we need a relevance feedback mechanism to revisit seed words and weights. Finally, the weights of all the sentences together are considered for the final grading of the answer. The whole idea is shown in Fig. 5, where we can see how relevance feedback works for revising weights.

![Fig. 5. Degree of relevancy at text line level for final grading.](image)

IV. EXPERIMENTAL ANALYSIS

As evident from the literature survey, there is no existing work for evaluating descriptive answers. Therefore, there is no standard database available in literature. To evaluate the proposed framework, one would have to collect answer scripts written by various students at different ages, which must include different scripts: documents with varying orientations, noise, sketchy writing, striked out text, artifacts and written by different students. As the number of answer scripts increases, the complexity of the dataset increases too. According to our knowledge, dataset creation and ground truth generation itself may require several years especially in a country where we can expect a large population, multiple scripts and a resulting large number of possible variations in writing. Since there are new systems which have the capacity to handle big data, it is now possible to develop an automatic system in current times.

The proposed framework involves many steps, namely, cleaning, strike-out text removal, text line and word segmentation, keyword spotting, degree of relevancy estimation at word level and degree of relevancy estimation at text line level. To evaluate each step of the proposed framework, we need to define measures. For cleaning, it is necessary to calculate recognition rate at character level or word level prior to cleaning and after cleaning. For the other steps there are standard measures, namely, recall, precision and F-measure. More details for recall, precision and F-measure can be found in [16].

The robustness and effectiveness of the proposed framework are to be tested on a large dataset which comprises multi-script, multi-oriented images, etc. The results should be compared with human scores for validation. In addition, the proposed system may be compared with the short answer evaluation to test robustness and effectiveness.
V. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a new roadmap for developing an automatic system for evaluating descriptive handwritten answers. We have proposed a new solution for each step of the proposed framework. For instance, gradient directional pattern for separating noisy text from actual text, text line and words segmentation, degree of estimation for word spotting, degree of relevancy estimation at word and text line level with the help of model answers and natural language processing concepts. The proposed concepts are script, orientation and writer independent. In addition, the proposed concepts are robust to noise, low resolution and degradation to some extent. The only main constraint is that the scope of the work is limited to descriptive answers that have been written in according to order of the questions.

There are many challenges ahead if we consider the possible variations in writing and population of students. Sometimes, students may not follow the order of questions. Instead, they choose their own choice or order of the question to write answers. It is also true that students write answers for different parts of the same question in different places by referring the question number. Developing generalized systems which work at subject level, college level, university level, province level, country level and world level has been an open research problem for several decades as it involves countably infinite number of variations and big data. Therefore, the major question is scalability and accuracy of unified systems. At the same time, collecting dataset and generating ground truth are equally important to judge the ability of such systems.

ACKNOWLEDGMENT

The work described in this paper was supported by the Natural Science Foundation of China under Grant No. 61672273, No. 61272218 and No. 61321491, the Science Foundation for Distinguished Young Scholars of Jiangsu under Grant No. BK20160021

REFERENCES


AI-based Automated Japanese Short-answer Scoring and Support System

Tsunenori Ishioka  
Research Division  
The Center for University Entrance Examinations  
Tokyo, Japan  
tunenori@rd.dnc.ac.jp

Masayuki Kameda  
c/o Research Division  
The Center for University Entrance Examinations  
Tokyo, Japan  
m-kamedax@nifty.com

Abstract—We have developed an automated Japanese short-answer scoring and support machine for new National Center written test exams. Our approach is based on the fact that accurate recognition of textual entailment and/or synonymy has been almost impossible for several years. The system generates automated scores on the basis of evaluation criteria or rubrics, by determining semantic similarity between the model answers and the actual written answers as well as a certain degree of semantic identity and implication. Human raters can revise the scores. Owing to the need for the scoring results to be classified at multiple levels, we use random forests to utilize many predictors effectively rather than use support vector machines. An experimental prototype operates as a web system on a Linux computer. To evaluate the performance, we compared human scores with the automated scores for a case in which 3–6 allotment points were placed in 8 categories of a social studies test as a trial examination. The differences between the scores were within one point for 70–90 percent of the data when high semantic judgment was not needed. We also applied the method to the second round of entrance examinations given by the University of Tokyo. The experiment in five test issues of a world-history test was conducted with 20 allotment points. The differences between the scores were within 3 points for 16 of 20 data provided by the NTCIR QALab-3 task office. These figures do not reflect human corrections.

Index Terms—writing test, automated scoring, machine learning, random forests, recognizing textual entailment

I. INTRODUCTION

An educational advisory body to the Japanese government has decided that writing tests will be introduced into the new national center test for university entrance examinations, as announced in a final report [5] at the high school and university articulation meeting by the Ministry of Education, Culture, Sports, Science and Technology. The use of AI-based computers was proposed to stabilize the test scores efficiently. The required type of writing test is a short-answer test, where a correct answer is expected to exist. Therefore, the test is scored by judging agreement of the meaning with the correct answer.

This type of writing test is called a short-answer test. Since short-answer scoring involves technical difficulty, the number of characters is restricted to 120 characters at most from dozens of characters. Two characters in Japanese are generally equivalent to one word in English. A short-answer test is widely considered to be more authentic and reliable for measuring ability compared with a multiple-choice test. If technical problems related to the short-answer test are solved, the potential demand for its use, as well as that for the national center test, will be enormous.

A short-answer scoring system as well as essay assessor [6] has also been developed because of its importance, various technical problems remain unsolved. In 2012, a Kaggle competition for short answer scoring had completed [2]. Each answer was approximately 50 words in length. The winner, Luis Tandalla [8], made the best score of 0.77166 evaluated with the quadratic weighted kappa error metric [3], which measures the agreement between two raters (system and human). The real number of 1 shows complete agreement between raters. Whereas human benchmark took the score of 0.90013. Automated assessment is not yet in the stage of practical application.

Therefore, we conceived of a support system for short written tests where a human rater can correct the automated score by referring to the original scores. When the human rater agrees with the result of the automated score, he/she can just approve the score indicated by default and can produce the corresponding mark. We chose to leave room for human raters to overwrite it without making it a perfect automated scoring system.

To evaluate the performance, we use two types of written tests. One is typical short-answer test, which model answers are consist of 20–60 Japanese characters in length. The other is the most voluminous test of 450–600 characters, where eight keywords to be included are specified. This limits the scope of the desirable answers; that is, variations in how to write correct answers can be restricted.

In what follows, Section 2 indicates the test issues and the model answers used in a trial examination for university entrance examinations. Section 3 shows the specifications of our proposed system. Section 4 presents our evaluation of the performance. Section 5 concludes with a summary.

JSPS KAKENHI Grant Number JP17H01843.
II. TEST ISSUES USED IN A TRIAL EXAMINATION

A. Gakken nation-wide trial examination

We assigned a theme in three subjects of world history, Japanese history, and geography of the “Gakken nation-wide trial examination” in fiscal year 2015. The world history test set includes four written test issues and two test issues each for geography and Japanese history; the total is 8 test issues.

Table I shows the “content” asked and the “correct answer,” which are given to test examinees in a distributed booklet of “test answers and explanations.”

B. World history for Tokyo University’s second round examinations

We also assigned five issues in the subject of world history for Tokyo University’s second round examinations in the past. The world history test set includes several types of written tests, and we evaluated the test issues required for the most voluminous test of 450–600 characters.

Table II shows the “content” asked and the “mandatory words/phrases,” which are given by test writers to the examinees.

Besides these, the following are given: (1) three model answers per issue, (2) partial sentences generated from the model answers, and (3) its importance as evaluated by professional raters. However, these are omitted due to space limitations.

The allocated number of points to every test issue is 20. If mandatory words or phrases are missing, 5 points are deducted per omission. Also, if the amount of words exceeds the limit, the score is halved. These are based on our speculation about the actual scoring standards of Tokyo University’s entrance examinations.

III. SPECIFICATIONS OF THE SCORING SUPPORT SYSTEM

A. Outline

Our system is for automated scoring and for supporting human raters. The approach functions as follows.

1) A system automatically judges each answer posed on whether or not its prepared key phrases agree with those of the model answer using the “scoring criteria” from a surface-like point of view.

2) The system gives not only a temporary score based on the criterion-based judgment but also a prediction score offered by machine learning based on the understanding of other human raters or supervised data. A certain degree of semantic meaning is also used.

3) A human rater can certify the prediction score by which a system presents this information as reference. He or she can correct this and overwrite based on his/her judgment.

To reduce the time and effort, the system precision should possess a certain degree of fitness with human ratings; more than 80% of the precision is desirable for tentative targets.

The flowchart of our system is as shown in Figure 1.

(a) Before scoring, we collected a lot of score data from various human raters and performed a machine learning of “Random Forests” [1]. The degree of fitness with the scoring guideline is also necessary. On the basis of these learning results, we set up a scoring engine to return the scores for new answers.

(b) The system generates a scoring screen written in the Hyper Text Markup Language.

(c) A user or human rater opens a scoring screen of (b) using a web browser on his/her terminal machine. Then, a Common Gateway Interface (CGI) program is activated.

(d) New learning model can be reconstructed using expert evaluation.

B. Scoring Screen

Figure 2 shows a screen shot of our prototype system. “The answer sentence that should be scored” (in red ink) is located in the upper part of the system; the middle part has some scoring criteria such as “synonyms and permitted different transcriptions,” “model or correct answers that warrant a full mark,” “partial phrases that warrant partial scores,” and “mandatory phrases.” For the “model answer” and “partial correct phrases,” the system judges the degree of fitness with the answer sentence to be scored; the system also judges whether or not the answer sentence includes “mandatory phrases,” whether or not it is meaningfully composed, and whether or not it exceeds the character limit; if the answer should be written as a noun or noun phrase, the system judges whether or not it matches the specified “type” format. These judgments are either yes or no, and toggle buttons are used.
A human rater reviews these judgments and revises them if necessary. Tentative scores located in the lower part are based on the aforementioned alternative judgment. The right-hand window is to determine the final score. The initial mark is settled by which predictive probability based on the past learned results gives the maximum. The probability values are also indicated.

When no learning data exist, that is to say, when no pre-scored data about the relevant test item exist, the message to that effect is shown in the top windows: no probability and no initial mark are naturally determined.

C. Automatic screen creation from a scoring criterion file

Our system is a Web application. Thus, the screen indicated by figure 2 is generated by HyperText Markup Language. We built the mechanism to make this HTML file automatically from a plain scoring criterion file that a computer beginner can handle.

Figure 3 is a plain original file that makes a screen like the one in figure 2. Two or three elements are set for criteria. In order, the label, allotment of points, and correspondence are located. The tab is the delimiter.

Synonyms and different transcriptions are recorded in “syno,” which appeared in “gold” as a model answer and in “part” as a partially correct phrase. “Syno” is not always limited to a definite lexical meaning. When it semantically the same meaning, it is also permitted. “Part” includes two types; one is possible to add to a partial point, and the other is for which a maximum is taken. If there are multiple same labels (for example, part1), we use the maximum of the points; different labels (for example, part 1 and part 2) can add the allotted points. “Lack” is a mandatory phrase; if no phrases exist, the point is deducted. A comma can be used for the meaning of “both.” “Vol” shows the number of characters available. “None” shows a nonsense sentence, and “goji” shows a wrong word such as kanji that does not exist. Minus points indicate points to be deducted.

We use “fitness” as the degree of the relationship between the written answer and “model answer” designated in “gold” or “partial correct phrases” in “part.” We define this as the harmonic mean of two kinds of relationships: one is the degree of the reference during the sentence keywords from the viewpoint of a written answer; the other is that from a model answer. These relationships are just like precision and recall often used in information retrieval, e.g., a Google search. This harmonic mean or “fitness” is called an F-measure taking a float number from 0 to 1. Our system rounds this to either 0 or 1 as a toggle button occurrence, and it shows a non-rounded value as a reference for the user.

D. Difference between the tentative score and the mechanical prediction score

The example indicated by figure 2 is a case of World history B2 #3. The answer indicated here is as follows: Indigenous people in a place of conquest originally had both “Jizya” and “kharaj” imposed on them, but only “kharaj,” like it was with Arabs, later came to be imposed on them. Compared with the model answer of “Even when indigenous people in a place of
A. Evaluation of the classification

We built a prediction model using machine learning of random forests (RFs) V4.1 [4]. The predictors include not only the degree of fitness indicated in Figure 3 but also semantic (cosine) similarity between the answer, model answer, and test item sentences. The reasons for using the RFs in the methods are as follows:

1) When using many predictor variables, the classification often functions effectively.
2) The degree of contributions can be estimated to determine effective predictor variables quantitatively in the classification.
3) RFs are suitable for this because test scoring requires multiple classifications with values of 0–3 or 0–6.

TABLE II
CONTENT AND MANDATORY WORDS/PHRASES OF WRITTEN TEST ISSUES

<table>
<thead>
<tr>
<th>Test issue # (Allotment)</th>
<th>Content and the mandatory words/phrases</th>
</tr>
</thead>
<tbody>
<tr>
<td>B792W10-1 (20 pt.)</td>
<td>[content] Outline what kind of historical development Egypt has made since the birth of civilization. (540 characters in Japanese) [mandatory words] Battle of Actium/ Islam/ Ottoman Empire/ Saladin/ Nile River/ Nasser/ Napoleon/ Muhammad Ali/</td>
</tr>
<tr>
<td>C792W10-1 (20 pt.)</td>
<td>[content] From the 19th century to the beginning of the 20th century, a rapid increase occurred in the number of immigrants from China to the Americas and Southeast Asia. What kinds of circumstances can be considered, and how did the influx of people who migrated abroad affect the political movements in Mainland China. Describe these points within 450 characters in Japanese. (450 chars) [mandatory words] Abolition of colonial slavery/ Sugarcane plantation/ Gold Rush/ Incarceration/ Opium War/ Straits Settlements/ Interest collecting campaign/ Sun Yat-sen/</td>
</tr>
<tr>
<td>G792W10-1 (20 pt.)</td>
<td>[content] Explain in less than 510 letters how the tendency to promote war or to suppress war has appeared in the three periods of the Thirty Years’ War, the French Revolution, and World War I. (510 chars) [mandatory words] Treaty of Westphalia/ League of Nations/ Fourteen Points/ On the Law of War and Peace/ Total warfare/ Conscription system/ Nationalism/ Declaration on peace/</td>
</tr>
<tr>
<td>L792W10-1 (20 pt.)</td>
<td>[content] Discuss the roles of the Netherlands and the Dutch people in world history from the end of the Middle Ages to the present, with a look at the prospects of integration beyond the state. (600 chars) [mandatory words] Grotius/ coffee/ The Pacific War/ Nagasaki/ New York/ Habsburg Family/ Maastricht Treaty/ South African War/</td>
</tr>
<tr>
<td>P792W10-1 (20 pt.)</td>
<td>[content] From the Vienna conference to the end of the 19th century, argue about how the changes in Russia’s foreign policy brought about the international situation in Eurasia, while paying attention to the response of Western powers. (600 chars) [mandatory words] Afghanistan/ Ily province/ Primorsky/ Crimean War/ Treaty of Turkmenchay/ Berlin Conference/ Poland/ Lushun/</td>
</tr>
</tbody>
</table>

In fact, the winner of short-answer scoring Kaggle competition, Luis [8], used this RFs algorithm.

For eight test issues indicated in Table I, we compared human ratings with the estimate based prediction model. The best rate that a prediction and a correct answer were identical is 78% in the cases of Geometry B #4. The worst rate is 43% in the cases of Japanese History B1 #2, which was surprisingly low. This is because about 80% of examinees got zero scores; thus, machine learning does not work well. We omitted cross matrices between human ratings and the estimate because of space limitations.

Table III shows the probability in which the differences between the scores were within one point. These values consist of 71–95% removing the case of world history B2 #3, which is necessary for correct understanding of the meaning. It shows the performance of the classification was in the level available.

<table>
<thead>
<tr>
<th>Issue #</th>
<th>Prob.</th>
<th>Issue #</th>
<th>Prob.</th>
</tr>
</thead>
<tbody>
<tr>
<td>World History B2 #1</td>
<td>0.75</td>
<td>Japanese History B2 #1</td>
<td>0.86</td>
</tr>
<tr>
<td>World History B2 #3</td>
<td>0.48</td>
<td>Japanese History B2 #3</td>
<td>0.71</td>
</tr>
<tr>
<td>Japanese History B1 #2</td>
<td>0.76</td>
<td>Geometry B #1</td>
<td>0.91</td>
</tr>
<tr>
<td>Japanese History B1 #4</td>
<td>0.88</td>
<td>Geometry B #4</td>
<td>0.95</td>
</tr>
</tbody>
</table>

IV. PERFORMANCE EVALUATION

A. Evaluation of the classification

We built a prediction model using machine learning of random forests (RFs) V4.1 [4]. The predictors include not only the degree of fitness indicated in Figure 3 but also semantic (cosine) similarity between the answer, model answer, and test item sentences. The reasons for using the RFs in the methods of many machine learning techniques are as follows:

1) When using many predictor variables, the classification often functions effectively.
2) The degree of contributions can be estimated to determine effective predictor variables quantitatively in the classification.
3) RFs are suitable for this because test scoring requires multiple classifications with values of 0–3 or 0–6.
Fig. 2. Short-answer scoring and support system screen (In case of world history B2 #3)

syno jizya jizyah
syno Muslim Islam
gold  5  "Even when indigenous people in a place of conquest were Muslim, they were exempted from Jizya. Even when an Arab had land in a place of conquest, kharaj was imposed."
part1 2  "Even when indigenous people in a place of conquest were Muslim, they were exempted from Jizya."
part2 2  "Even when an Arab had land in a place of conquest, kharaj was imposed."
lack1 -1 "had land"
lack2 -2 jizya
lack2 -2 kharaj
lack2 -5 jizya,kharaj
vol -5 60-40
nons -5
goji -1

Fig. 3. Scoring criterion file (labels, allotment of points, and correspondences are tab delimited.)

In RFs, each tree is constructed using a different bootstrap sample from two-thirds of the original data. The remaining one-third of cases is used for the test data.

The default procedure runs 500 times and forms the final classification tree using the most votes, when using the RandomForest package implemented in R [4]. The error rate for the test set can be obtained at the same time. The numerical values shown in Table III were obtained using this procedure. The sample sizes, namely the numbers of examinees, ranged in 70–120, depending on the subjects. If we compare the original data with the estimates using the final RF model, the concordance rate will be near 100% because too many predictive variables were compared with the sample size.

For reference, the measure of quadratic weighted kappa ($\kappa$), which were used for Kaggle competition, for the eight test issues are shown in Table IV. The $\kappa$ for World History B2 #1 and #3 are comparatively small as well as the probabilities cited in Table III.

### Table IV

<table>
<thead>
<tr>
<th>Issue #</th>
<th>$\kappa$</th>
</tr>
</thead>
<tbody>
<tr>
<td>World History B2 #1</td>
<td>0.36</td>
</tr>
<tr>
<td>World History B2 #3</td>
<td>0.28</td>
</tr>
<tr>
<td>Japanese History B2 #1</td>
<td>0.63</td>
</tr>
<tr>
<td>Japanese History B2 #3</td>
<td>0.73</td>
</tr>
</tbody>
</table>

**B. Variables that contribute to the classification**

RFs evaluate the importance of variables in distinction using an index of the Gini coefficient. The bigger the coefficient, the more the classification is affected. Due to space limitations, we do not variables arranged in descending order of the Gini coefficient. However, the three dominant contributions in our cases are as follows:

**QA_sim:** The cosine similarity between test item sentences and the answer.
The authors would like to thank everybody in the Gakken group for offering the test sets, scoring criteria, and human scores. This project was supported by JSPS KAKENHI Grant Number JP17H01843.

\section*{Acknowledgment}

The numerical results have been shown when human correction did not enter. If human corrections are made, the performance will be improved. The modifications are quite simple, so that the labor is minimal.

\section*{References}

\begin{thebibliography}{9}
  \bibitem{2} The Hewlett Foundation. \textit{Short Answer Scoring}. Automated Student Assessment Prize, Phase Two, Sep 2012.
  \bibitem{5} MEXT. \textit{Publishing the final report of high school and university articulation meeting}. Ministry of Education, Culture, Sports, Science and Technology in Japan, Mar 2016.
  \bibitem{8} Luis Tandalla. \textit{Scoring Short Answer Essays}. The Hewlett Foundation: Short Answer Scoring, Sep 2012.
\end{thebibliography}
An Investigation of Discrete Hidden Markov Models on Handwritten Short Answer Assessment System

Hemmaphan Suwanwisat\(^a\), Abhijit Das\(^b,c,f\), Miguel A. Ferrer\(^c\), Umapada Pal\(^d\), and Michael Blumenstein\(^c\)

\(^a\)Information Technology Academy, James Cook University, Cairns, Australia, art.suwanwiwat@jcu.edu.au
\(^b\)Institute for Integrated and Intelligent Systems, Griffith University, Queensland, Australia, abhijit.das@griffithuni.edu.au
\(^c\)IDeTIC, University of Las Palmas de Gran Canaria, Las Palmas, Spain, mferrer@dsc.ulpgc.es
\(^d\)Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India, umapada@isical.ac.in
\(^f\)Inria Sophia Antipolis – Méditerranée, France

Abstract—This paper presents an investigation of an off-line automatic assessment system utilising discrete Hidden Markov Models. A set of geometric features were extracted from handwritten words and were later classified by HMMs. There were two training datasets employed in the experiments; the first training dataset contained all correct answers to the questions whereas another training dataset contained both correct and incorrect answers to the questions. Datasets contained 3,000 and 3,400 handwritten samples, respectively. The experiments yielded promising results whereby the highest recognition rate of 91.90% with a 100% accuracy was achieved on our database.

Keywords—off-line automatic assessment system, Hidden Markov Models (HMMs), fixed-point arithmetic, geometric features

I. INTRODUCTION

Even though presently computer-based examinations have become widely accepted, paper-based examinations are still in use worldwide throughout all levels of education, including but not limited to secondary and tertiary levels. Despite the fact that paper-based examinations have been in use all these years, to the best of the authors’ knowledge, literature regarding off-line automatic assessment systems is limited [1], [2] and [3]. Recognising off-line handwritten words is challenging when compared to recognising on-line handwritten words. There are a number of disadvantages in attempting to recognise off-line handwritten words because there is no real-time information available. Apart from that, whereas on-line recognition systems use both temporal and spatial information, only spatial information is available for off-line cases [4].

Recognising handwriting of students while answering questions in examinations can be considered difficult, as the students may be writing with significant stress and as a result could be writing in a way where legibility is reduced. Also there can be a high variance in the artefacts employed for examination (such answer sheet paper quality, colour, type of pen used, etc).

For essay or short-answer question assessment types, it is known that manually marking these types of exams is tedious, time consuming and most of all, error prone. To overcome this problem, an off-line Short Answer question automatic Assessment System (SAAS) is proposed in this paper. This study investigated the use of discrete Hidden Markov Models (HMMs) on short answer words.

HMMs have been used in both off- and on-line handwriting recognition systems. As stated by Plötz et al. [5], the sliding windows principle is an important milestone for successful Markov-model-based handwriting recognition, especially for off-line handwriting systems. HMMs are widely employed in automatic off-line recognition applications, including industrial ones. Hence, this study proposes a SAAS employing HMMs.

This study’s contributions include:

1) Exploring the efficiency of employing discrete HMMs on the proposed short-answer question words. There have not been any experiments performed on the SAAS system using HMMs previously. This study shows encouraging results by employing the stated classifier.

2) Investigating the effect of the numbers of training samples on classification rates. The previous studies [3] employed 80% of the total number of samples in the datasets, whereas in this proposed research, the training datasets contained 10 – 50% and 80% of the total number of samples in the datasets. It was found that by employing HMMs, the best classification results were obtained when only 10 – 20% of the total proportion of the datasets were used to train the classifiers.

The features employed in the proposed system were geometrical features [6] which were based on two vectors that represent the envelope description and the interior stroke distribution in polar and Cartesian coordinates. Since HMMs were employed, no segmentation of the images was required.

There are two main training datasets utilised in this investigation. For the first type, the training dataset only contained correct answer handwritten samples to the questions. For the second type, however, the training dataset contained both correct and incorrect answer handwritten samples to the questions. There were also two main testing datasets, similar to the training datasets; the first testing dataset only contained
correct answers to the questions, and the second one contained both correct and incorrect answers to the questions. There were altogether 18 sub-datasets employed in this study. In total, there were 3,000 and 3,400 handwritten samples in the two main dataset types employed in this study. More details are described in Section II.

The remainder of this paper is divided into three sections as follows: Section II describes the research methodology employed in this study, while the report on experimental results can be found in Section III. Conclusions and discussion of the future research can be found in Section IV.

II. METHODOLOGY

Figure 1 illustrates the methodology employed in this investigation in relation to the use of discrete HMMs and the proposed SAAS. The proposed methodology, classification technique and processes including handwritten short answer words preparation, pre-processing, word segmentation, and HMMs are discussed in this section.

A. Short Answer Handwritten Words

The answers to the questions employed in this study were designed to be a few words per question, which suits the purpose of the proposed short answer question assessment system. The answers to the questions were straightforward for example “What does IT stand for?”, The correct answer can only be “Information Technology” although the writers may write the words using different cases i.e. “information technology”, “Information technology”, etc.

The handwritten samples were obtained from datasets employed in [3]. There were 3,000 samples in the dataset which were obtained from 30 words written by 100 writers. A further 400 handwritten samples of commonly incorrect answers were collected to be used in another training dataset. The total number of samples was increased to 3,400 samples. Examples of handwritten short answers can be seen in Figure 2.

B. Datasets

There are two main types of training datasets and there are two main types of testing datasets. The first type of training dataset (TR I) contained only correct answers to the questions. The second type of training dataset (TR II) contained both correct and incorrect answers to the questions. For testing datasets, the two main types of datasets were also applied. The first testing dataset (TE I) contained only correct answers to the questions whereas the second testing dataset (TE II) contained both correct and incorrect answers to the questions. From the four main datasets (TR I, TR II, TE I, and TE II), there are 3 training and testing dataset combinations.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>No of Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>3,000</td>
</tr>
<tr>
<td>CII</td>
<td>3,400</td>
</tr>
<tr>
<td>CIII</td>
<td>3,400</td>
</tr>
</tbody>
</table>

The first combined dataset (CI) contained TR I and TE I which means that only correct answers to the questions were contained in this dataset. Since there are only correct answers to the question in this dataset, the size of the dataset is 3,000 samples. The second combined dataset (CII) contained TR I and TE II which means that the training dataset contained only correct answers to the questions whereas the testing dataset contained both correct and incorrect answers to the questions. There were 3,400 samples in this dataset since there are incorrect samples included in the dataset. For the last combined dataset (CIII), both correct and incorrect answers to the questions comprised both training and testing datasets. Same as CII, CIII contained 3,400 samples in its dataset (see Table I). The False Acceptance Rate (FAR) and False Rejection Rate (FRR) score statistics are also summarised in Table II.
Each of the combined datasets (CI, CII, and CIII) were further divided into six sub-datasets. Each sub-dataset contained different amounts of training and testing samples. Table II below shows the percentages of amounts of samples in each of the sub-datasets. The numbers of training samples in training datasets were between 50 – 10% and 80%. Therefore, the numbers of testing samples in testing datasets were between 50 – 90% and 20% respectively (see Table II). The different numbers of training samples in the datasets were meant for the investigation on the efficiency of discrete HMMs towards the proposed SAAS.

### TABLE II. EACH SUB-DATASET – NUMBER OF TRAINING AND TESTING PERCENTAGES

<table>
<thead>
<tr>
<th>Sub-dataset</th>
<th>Training (%)</th>
<th>Testing (%)</th>
<th>FAR and FRR for CI</th>
<th>FAR and FRR for CII and CIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>S I</td>
<td>80%</td>
<td>20%</td>
<td>FAR=30*20</td>
<td>FRR=34*20</td>
</tr>
<tr>
<td>S II</td>
<td>50%</td>
<td>50%</td>
<td>FAR=30*50</td>
<td>FRR=34*50</td>
</tr>
<tr>
<td>S III</td>
<td>40%</td>
<td>60%</td>
<td>FAR=30*60</td>
<td>FRR=34*60</td>
</tr>
<tr>
<td>S IV</td>
<td>30%</td>
<td>70%</td>
<td>FAR=30*70</td>
<td>FRR=34*70</td>
</tr>
<tr>
<td>S V</td>
<td>20%</td>
<td>80%</td>
<td>FAR=30*80</td>
<td>FRR=34*80</td>
</tr>
<tr>
<td>S VI</td>
<td>10%</td>
<td>90%</td>
<td>FAR=30*90</td>
<td>FRR=34*90</td>
</tr>
</tbody>
</table>

### C. Image Acquisition

All handwritten samples were scanned with 300 dpi resolution and stored in a grey-level format. The images were then binarised and segmented at the word level. Words were segmented and checked to ensure that there were no segmentation errors. Noise removal, skew and slant normalisations were performed on each image.

### D. Feature Extraction Technique

Geometrical features [7] were employed in this investigation. Originally, this feature extraction technique was created for off-line signature verification. This investigation was also conducted to find out whether this technique is suitable for the verification of the students from their handwriting as well. The geometrical features are based on two vectors. They represent the interior stroke distribution of polar and Cartesian coordinates and the envelope description [7]. Outline detection and representation, feature vectors based on polar coordinates and feature vectors based on Cartesian coordinates are briefly described as follows:

1) **Outline Detection and Representation:** Morphological operations were used to calculate the outline. A dilatation was applied in order to reduce the word variability, after that the outline extraction process was simplified by a filling operation. After filling, a number of objects were detected, then a horizontal dilatation was performed until all the objects were connected. As a result, a sequence of the outline’s Cartesian coordinates, being its length, was obtained.

2) **Feature Vector Based on Polar Coordinates:** In order to represent a handwritten word outline in polar coordinates, it was decided to select equidistant samples of the envelope and represent each sample as a three-component feature vector being 1) the derivative of the radius, 2) its angle, and 3) the number of black pixels that the radiuses crossed when sweeping from one selected point to the next.

The radius function is calculated as the number of pixels from the geometric centre to each outline selected point as:

\[
d_1 = X_t - C_x, d_2 = Y_t - C_y
\]

The angle of each selected contour sample is calculated by means of the arctan function implemented through a lookup table:

\[
\theta_t = \arctan \frac{X_{at}/T_Y}{Y_{at}/T_X}, t = 1, 2, ..., T_r
\]

3) **Feature Vector Based on Cartesian Coordinates:** This vector is also based on the envelope and the signature strokes density parameterisation, however in this scenario, using Cartesian coordinates. The envelope was divided through the geometric centre into top and bottom halves. The height of the top half and bottom halves. The height of the top half at equidistant points, obtaining the sequence. After that, the bottom-half sequence was obtained.

The envelope was then divided into two halves again, and subsequently the left and right-hand sides were obtained through the geometric centre. As a result, two sequences were obtained. The feature vector sequence was composed of four dimensional vectors, the first component of these vectors was designed to help the HMM synchronisation. A full explanation of these algorithms can be found in [7].

### E. Hidden Markov Models (HMMs)

In this study, discrete HMMs [7] were selected to model each word’s feature; this is to avoid making an assumption on the form of the underlying distribution. Each of the words (answers) was modelled with two left-to-right HMMs. The number of states in each signer’s HMM words is thirty-five. This topology only allowed transitions between each state to itself and to its immediate right-hand neighbours. The classification, decoding, and training problems were solved with the Forward-Backward algorithm, the Viterbi algorithm, and the Baum-Welch algorithm.

The K-means algorithm was used for the training process to create multi-labelling VQ which made a soft decision about which code words were the closest to the input vector. To verify each answer, the log likelihood of the two HMMs that modelled the answer was obtained. The fusion of both scores can be performed by regarding the problem as a classification or a combination problem. If scores obtained were greater than the threshold, the answer was accepted.
The HMM software employed in this study was the GPDShmm toolbox which can be freely downloaded from http://www.gps.ulpgc.es/download/index.htm [8]. All the experiment performed were executed using Matlab in Windows 7 environment.

F. Experiment Evaluation Rates

The SAAS evaluations employed two rates, being classification and accuracy rates. The first rate, classification rate, was used to indicate the rate that the words in the testing datasets were recognised. The second rate, accuracy rate, was the rate which indicated the accuracy of the proposed system when the recognised words matched the answers to each of the questions.

III. EXPERIMENTAL RESULTS AND DISCUSSION

This section reports results obtained from the experiments performed. As described earlier in Section II, there are three main types of dataset (CI, CII, and CIII) employed in the experiments.

Whereas CI contained only correct answers to the questions in its dataset, CII and CIII contained both correct and incorrect answers in their datasets.

The difference between CII and CIII was that CII did not use incorrect answers in the training process whereas CIII did. Each type of data was further divided into six sub-datasets; the results of employing each sub-dataset are described as follows:

a) Classification Rates Obtained from Employing CI (Trained with TR I and Tested with TE I).

The results of each sub-dataset are displayed in Table III. It can be seen from Table III that the best classification rate of 91.90% was obtained when the discrete HMMs were trained with 10% (300 samples) of the total dataset and tested with 90% (2,700 samples) of the total dataset.

It can be noted that the classification rates seemed to fluctuate more as the number of samples in the training datasets were decreasing; this was different from the results obtained when CI was used in the experiments (see Table III).

b) Classification Rates Obtained from Employing CII (Trained with TR I and Tested with TE II).

In this dataset, the training dataset did not contain any of the incorrect answers to the questions, however, the testing dataset did. The results of each sub-dataset are displayed in Table IV.

It can be seen from Table IV that the best classification rate of 89.15% was attained when only 10% (300 samples) of the total dataset was used for training. This result was similar to the highest result of CI. However, having incorrect answers to the questions in the testing dataset lowered the best classification rate by 2.75%.

It can be noted that the classification rates seemed to fluctuate more as the number of samples in the training datasets were decreasing; this was different from the results obtained when CI was used in the experiments (see Table III).

c) Classification Rates Obtained from Employing CIII (Trained with TR II and Tested with TE II).

From Table V, it can be observed that the best classification rate was increased when the discrete HMMs were trained with incorrect answers to the questions as well as the correct ones. The highest classification rate of 89.88% was obtained when the classifier was trained with either 80% or 20% (2,700 and 300 samples, respectively).

As expected, the classification rates increased when the incorrect answers were also used in training. The improvement went up to 0.73%. This 0.73% may appear to be nominal, however, in SAAS, this is very important. It may cause students to fail their exam if the system couldn’t mark their paper correctly.

Similar to CII, small fluctuations across 3,000 samples could be seen as the numbers of samples in the training datasets
decreased; this was different from the results obtained when CI was used in the experiments (see Table III). Since only small fluctuations were observed, this could not be considered statistically meaningful.

Consistent classification rates obtained from experiments performed on both CII and C III (see Table IV and V) suggested that the proposed SAAS system is robust with respect to the sizes of the training datasets.

It could be noted that by using CIII, the gap between CI (which only contained correct answers to the questions) was reduced to 2.02% compared to a 2.75% gap between CI and CII. The comparison results between each dataset’s best classification rates, together with their corresponding settings, are displayed in Table VI.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training and Testing Ratio (%)</th>
<th>Best Classification Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI – S VI</td>
<td>Train 10% - Test 90%</td>
<td>91.90</td>
</tr>
<tr>
<td>CII – S VI</td>
<td>Train 10% - Test 90%</td>
<td>89.15</td>
</tr>
<tr>
<td>CIII – S V</td>
<td>Train 20% - Test 80%</td>
<td>89.88</td>
</tr>
</tbody>
</table>

It could be noted from Table VI, that the best classification rates were obtained when the numbers of training samples were small; from the experiments, the suitable range was when the HMMs were trained with 10 – 20% (300 – 640 samples) of the total dataset sizes of 3,000 and 3,400 samples, respectively.

DET curves of the experiments can be found in figure 3. Along the X-axis the FAR scores are plotted and along Y-axis the FRR scores.

d) Comparison between the proposed SAAS employing discrete HMMs and other off-line word recognition techniques found in the literature: As discussed earlier under the Introduction Section, the amount of research conducted on off-line SAASs could be considered quite small; as a result the comparison in this study was performed with other off-line word recognition techniques found in the literature.

<table>
<thead>
<tr>
<th>System – Feature Extraction Techniques</th>
<th>DS</th>
<th>CR (%)</th>
<th>AR (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>English Numeral Recognition – Hybrid Features (Moment of Inertia and Projection) [8]</td>
<td>3,500</td>
<td>91.7</td>
<td>91.7</td>
</tr>
<tr>
<td>Arabic Handwriting Recognition System - baseline estimation – HMMs/MLP [9]</td>
<td>736</td>
<td>89.03</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>1,077</td>
<td>54.00</td>
<td>99.00</td>
</tr>
<tr>
<td>The proposed SAAS – Geometrical Features - HMMs</td>
<td>3,400</td>
<td>91.90</td>
<td>100</td>
</tr>
</tbody>
</table>

Fig. 3. DET curves of the experiments.
Upon observing the comparison in Table VII, it can be seen that when employing HMMs with the same dataset used in the previous study [11], the classification rate attained from this study is lower than [11] by less than 3%, however, it must be noted that since the proposed system is a SAAS, the accuracy rate is crucial. Any accuracy rate less than 100% would be considered unacceptable and unusable as students may fail their exam even though they answered the questions correctly. Given this reason, it can be considered that the proposed SAAS, employing HMMs, yielded better results. Furthermore, the proposed system only employed 10 – 20% of the total samples in the datasets for training compared to 80% used in the previous study [11] to achieve this result. Hence, it can be considered an efficient system.

The other existing SAASs were also able to achieve high accuracy rates of 99 – 100% [1] and [2]. However, the dataset sizes are relatively small (145 and 1,077, respectively). Furthermore, the classification rates were lower than the rate achieved in the present study (65%, 54%, and 91.90%, respectively). The proposed classification system can be considered comparable to some of those found in the literature [8], [9] and [10].

IV. CONCLUSION AND FUTURE WORK

This study proposed a SAAS employing discrete HMMs; the geometrical features used were the envelope description and the interior stroke distribution in polar and Cartesian coordinates. Using HMMs, no explicit segmentation was required.

The experimental results were encouraging; the range of classification was from 85.67 – 91.90% with 100% accuracy. It must be noted that for SAASs, accuracy rates are crucial; any accuracy rates less than 100% would be considered unacceptable. Furthermore, by employing HMMs, the amount of samples required for training was reduced dramatically from 80% [11] to only 10 – 20%. This means that it is more efficient to employ HMMs in SAASs as the results are promising even though, only a small number of samples were required, and yet a 100% accuracy rate was attained.

Some suggestions are presented here for future work to improve the classification rates. Different algorithms and settings of HMMs can be applied. Furthermore, hybrid classifiers (e.g. HMMs & SVMs, HMMs & ANNs, etc.) can be employed. Rather than utilising a whole word recognition approach, segmentation-based recognition may be applied to SAAS. Different techniques such as deep learning can also be investigated on the datasets. More complex datasets (i.e. increasing from word to sentence level, larger dataset sizes, multilingual) can be collected and employed in future work.

ACKNOWLEDGEMENT

The authors sincerely appreciate the volunteers who submitted their samples for datasets development, and those who helped in the collection process. Upon request, the database is available for download to the research community.

REFERENCES

Oral Session F

Tracking and Motion

Tuesday May 15, 2018, 2:50 PM

Serguei Mokhov, Miao Song, Jashanjot Singh, Joey Paquet, Mourad Debbabi and Sudhir Mudur
   Toward Scalable Visual Digital Evidence Visualization and Multimodal Interaction using Computer Vision Techniques

David-Alexandre Beaupre, Guillaume-Alexandre Bilodeau and Nicolas Saunier
   Improving Multiple Object Tracking with Optical Flow and Edge Preprocessing

André Apitzsch, Roman Seidel and Gangolf Hirtz
   Cubes3D: Neural Network based Optical Flow in Omnidirectional Image Scenes

Yucao Tang and Guillaume-Alexandre Bilodeau
   Evaluation of trackers for Pan-Tilt-Zoom Scenarios

Vida Adeli, Ehsan Fazl-Ersi and Ahad Harati
   Enhancing Human Action Recognition through Temporal Saliency
Toward Multimodal Interaction in Scalable Visual Digital Evidence Visualization Using Computer Vision Techniques and ISS

Sergeu A. Mokhov  
Concordia University, Montreal, Canada  
mokhov@encs.concordia.ca

Miao Song  
Concordia University, Montreal, Canada  
m_song@encs.concordia.ca

Jashanjot Singh  
Concordia University, Montreal, Canada  
s_jashan@encs.concordia.ca

Joey Paquet  
Concordia University, Montreal, Canada  
paquet@encs.concordia.ca

Mourad Debbabi  
Concordia University, Montreal, Canada  
debbabi@encs.concordia.ca

Sudhir Mudur  
Concordia University, Montreal, Canada  
mudur@encs.concordia.ca

Abstract—Visualization requirements in FORENSIC LUCID have to do with different levels of case knowledge abstraction, representation, aggregation, as well as the operational aspects as the final long-term goal of this proposal. It encompasses anything from the finer detailed representation of hierarchical contexts to FORENSIC LUCID programs, to the documented evidence and its management, its linkage to programs, to evaluation, and to the management of GIPSY software networks. This includes an ability to arbitrarily switch between those views combined with usable multimodal interaction. The purpose is to determine how the findings can be applied to FORENSIC LUCID and investigation case management. It is also natural to want a convenient and usable evidence visualization, its semantic linkage and the reasoning machinery for it. Thus, we propose a scalable management, visualization, and evaluation of digital evidence using the modified interactive 3D documentary system – Illimitable Space System – (ISS) to represent, semantically link, and provide a usable interface to digital investigators that is navigable via different multimodal interaction techniques using Computer Vision techniques including gestures, as well as eye-gaze and audio.

Index Terms—Illimitable Space System, FORENSIC LUCID, DFG, GIPSY, forensic computing, motion capture, computer vision

I. INTRODUCTION

We propose a scalable management, visualization, and evaluation of digital evidence using the modified interactive 3D documentary component of the Illimitable Space System (ISS) to represent, semantically link, and provide a usable interface to digital investigators.

The cyberforensic analysis is one phase of the cybercrime investigation where the investigators strive to produce credible inferences based on evidential information. The source of this information is usually the phases that precede the analysis such as evidence acquisition and encoding. Also, this information can come from an esoteric set of resources that involves computers but is not limited to that seem fit as an evidence by the investigators [1], [9].

LUCID programs are data-flow programs that can be visually composed and illustrated as data-flow graphs as well. FORENSIC LUCID is one such LUCID dialect that enables investigators to specify and reason about cyberforensic cases. It represents the context of the evaluation of the evidences’ by encoding them along with witness stories, evidential statements and modeling the crime scene to cross-validate claims against the model and perform event reconstruction, potentially within large swaths of digital evidence [1], [9].

In 2004, Gladyshev [10] introduced the first formal approach to cybercrime investigation. Their approach uses Finite State Automata to describe the digital system as a Finite State Machine for event reconstruction. However, it has an associated learning curve and quite complex for investigators without a formal background in computer science. FORENSIC LUCID is designed to explicitly address these drawbacks and aims to be usable, expressive, sound and complete.

One of the many goals of FORENSIC LUCID is usability via scalable visualization of enormous data under investigation. Recently, there have been significant improvements in the domain of modern 2D and 3D virtual reality environments, which can be easily navigated via a variety of different multimodal interaction techniques. FORENSIC LUCID aims to up the ante by providing such usability improvements by leveraging modern multimodal techniques in virtual and augmented reality space for the investigators instead of writing a FORENSIC LUCID program to navigate seamlessly. A combination of gestures, audio commands, eye gaze and hardware controllers are potential candidates to provide navigational and interaction abilities to the investigators. It will enable us to extend the Lucid DFG programming onto FORENSIC LUCID case modeling and specification [1], [9].

The purpose here is to determine the applicability of these findings to FORENSIC LUCID and investigation case management. It is also natural to want a convenient and usable evidence visualization, their semantic linkage and the appropriate hardware for the same. The visualization requirements in the context of FORENSIC LUCID revolve around the different levels of the case knowledge abstraction, its representation,
aggregation, and the operational aspects as the final long-term goal of this proposal. It encompasses everything from the finer detailed representation of hierarchical contexts to FORENSIC LUCID programs, to the documented evidence and its management. It also includes its linkage to programs, to evaluation and to the management of GIPSY software-defined networks along with an ability to arbitrarily switch between those views combined with usable multimodal interaction [1], [9].

II. RELATED WORK

In the context of data-flow programming languages, there are quite a few research works and proposals that revolve around graph-based visualizations. The work of Faustini proved in particular a visualization of any INDEXICAL LUCID program as a DFG [11].

In 1995, Jagannathan defined one of the first graph-based visualizations proposals for LUCID programs. He defined different graphic intensional and extensional models for GLU programming [12]. Further, in 1999 Paquet’s doctoral work with multidimensional intensional programs extended on it, followed by the visual parallel programming idea of Stankovic, Orgun, et al. [13] [14].

In 2004, Ding provided practical implementation of Paquet’s foundational work within GIPSY in the form of 2D DFGs [13], [15] [15]. Ding provided an automatic bidirectional translation of the intensional programs between their textual and graphical representations by employing lefty’s GUI (Graphviz’s) and dot’s languages [9], [16], [17].

Mokhov proposed an idea of one such “3D editor” within RIPE [18] to visualize, control communication patterns and load balancing in GIPSY. The editor’s idea is to render graphs in a 3D space to allow its users to redistribute demands visually in case of imbalance of workload among the workers. It can be thought of as a virtual 3D remote control accompanied by a miniature expert system which can trigger the planning, caching and load balancing algorithms to learn and perform efficiently every time a related GIPSY application is run.

Similarly, several authors put forward their works on visualizing the configuration, formal systems and load balancing with corresponding graph systems [19]–[23].

These works defined key concepts that are consistent with GIPSY [9] visual mechanisms especially, the General Manager Tier (GMT) [24]. Rabah provided the initial configuration management and PoC visualization for GIPSY nodes and tiers via the above mentioned GMT [25].

In 2012, Tao et al. proposed another interesting work of relevance on the visual representation of event sequences, reasoning, and visualization of EHR data [26]. Wang et al. put forward a temporal search algorithm for event visualization of personal history [27]. Monroe et al. noted the challenges of specifying intervals and absences in temporal queries and approach those with the use of a graphical language [28]. This could be of particular use for no-observations [1] in FORENSIC LUCID case. A recent novel HCI concept of documentary knowledge visual representation and gesture-and speech-based interaction in the Illimitable Space System (ISS) was put forward by Song [29] in 2012. A multimodal case management interaction system was proposed for the German police called Vispol Tangible Interface: An Interactive Scenario Visualization 1.

Building upon the above-mentioned works, we propose to illustrate nested evidence, crime scene and the reconstructed event flow after revaluation in the form of a 2D or 3D DFG. The direct impact is to aid the forensic investigators by providing a scalable visualization, management of evidence modeling, encoding by FORENSIC LUCID [1], [30]–[32] and subsequently its evaluation by GIPSY [9].

A. Conceptual Visualization Design

Deriving from the related research work in context to visualization of LUCID programs, a conceptual example of a 2D DFG that corresponds to a simple LUCID program produced by Paquet [13]. Presently, the rendering of the same is by Ding in 2004 [15] within the GIPSY environment [9].

In Figure 2, page 2 is the conceptual model of hierarchical nesting of the evidential observation sequences os, their individual observations o (consisting of the properties being observed (F, min, max, w, t), details of which are discussed in the referenced related works and in [1, Chapter 7]). These 2D conceptual visualizations are proposed to be renderable at least in 2D or in 3D via an interactive interface to allow modeling complex crime scenes and multidimensional evidence on demand. The end result is envisioned to look like either expanding or “cutting out” nodes or complex-type results as exemplified in Figure 1

1http://www.youtube.com/watch?v=_2DyswIPNDQ

cutout image credit is that of Europa found on Wikipedia http://en.wikipedia.org/wiki/File:PIA01130_Interior_of_Europa.jpg from NASA

![Fig. 1. Modified conceptual example of a 2D DFG with 3D elements](http://www.youtube.com/watch?v=_2DyswIPNDQ)

![Fig. 2. Conceptual example of linked 3D observation nodes](http://en.wikipedia.org/wiki/File:PIA01130_Interior_of_Europa.jpg)
III. MULTIMODAL VISUAL ENCODING OF FORENSIC LUCID-BASED EVIDENCE

Data visualization, not only in the context of cybercrime investigation with FORENSIC LUCID, but in almost every other domain as well provides numerous advantages in terms of deducing inferences, spotting anomalies and recognizing patterns. However, specifically in case of FORENSIC LUCID and investigating cybercrimes, it provides additional usability enhancements to aid investigators to illustrate and define semantic links among the related evidence.

Furthermore, the need to visualize forensic cases, digital evidence, and related specification components revolve around providing usability enhancements to aid the investigators. Additionally, putting the program (specification) in 3 dimensions, especially in the modern and affordable augmented and virtual reality spaces (AR/VR) will help in structuring the program along with the case well arranged in a virtual environment with the digital evidence enclosed within 3D spheres. Moreover, navigable in depth to whatever levels of detail possibly via one of the multimodal interactions, although in the given example via clicking, issuing voice commands, gazing, or gesturing [9].

In case of event reconstruction, in particular, the illustrations and comprehension of operational semantics and demand-driven models are much better along keeping in mind their depth and complexity. Ding’s work provides navigational capabilities from a graph to subsequent subgraphs via extending complex nodes to their definitions as in whenever (whenever) or advances upon (upon), their reverse operators, forensic operators, and others [9] found in [1, Chapter 7].

A. Augmented System Requirements

In order to realize the envisage of DFG visualization of FORENSIC LUCID programs and their evaluation by GIPSY some immediate considerations are discussed below [9]:

- Hierarchical evidential statements with deeply nested contexts should be visualized [9].
- Intentional-imperative hybrid nodes need to be placed in DFGs combining fragments of LUCID and JAVA programs [9]. Previous research works by GIPSY R&D did not address the aspects to augment the DFGAnalyzer and DFGGenerator from Ding’s work in some fashion to provide support for hybrid GIPSY programs. However, to address this one can think to add an “unexpandable” imperative DFG node to the graph, but depth-wise it won’t be just enough to click their way through. Thus, considering possibilities to make it usable hence expandable recent enhancements in Graphviz and GIPSY can be leveraged to generate FORENSIC LUCID code from the DFG and vice versa [9].
- Rabah’s work on visualizing load balancing and communication control patterns for tasks in Euclidean space may as well be leveraged via the GGMT [25].
- The ability to switch among views such as DFG, evidence, and control, etc., is required as well.
- In flat-screen, touch-screen, augmented reality or projected environments, 3D DFG interactions are to support click and touch or voice-based call-outs as well as gestures to link or assemble some of the evidence [4].
- In the virtual reality environment VR controllers and gaze-controlled interactions are essential in addition to the voice and gesture recognition support [34].

B. Survey of the Visualization Languages and Tools

This work focuses on one of the goals of this research, which is to find the optimal technique with its formal specifications along with being feasible to implement using currently available HCI technologies and a usable one [9].

1) Graphviz: Ding’s [15] basic bidirectional translation between GIPL and DFG within GIPSY is already a part of the project and exists for GIPL and INDEXICAL LUCID, the two LUCID dialect antecedents. Moreover, Graphviz modern version now supports integration with Eclipse [35], thus GIPSY’s IDE—RPIE (Run-time Interactive Programming Environment)—can be an Eclipse-based plug-in as well [9].

2) PureData: The PureData [36] language by Puckette along with its commercial divisions namely (Jitter/Max/MSP [37]) apply a DFG-lie programming by graphically placing inlets and outlets of any data type connected in the form of so-called “patches”. These inlets may have external implementations and sub-graphs in procedural languages. Originally, Puckett’s work used signal processing to process electronic music and videos in order to produce interactive artistic and performative processes and was extended beyond that domain. The notion of external plug-ins in PureData allows deep visualization of media in OpenGL which in turn enhances the overall aspect of the process. PureData does draw influence from LUCID as a data flow language as well [9].

3) BPEL: OpenESB IDE provides visual design capabilities to visually illustrate or create a BPEL (Business Process Execution Language) process along with composite applications in the context of Service Oriented Architectures and Web Services. [38]–[40]. These BPEL specifications are translatable to an executable web service composition code in JAVA language. Not only it provides capabilities in terms of designing flows between structures, parallel, asynchronous, sequential processes and fault realization, but more importantly, BPEL notations have a backing formalism modeled upon based on Petri nets. BPEL specifications’ composite applications actually translate to executable JAVA web services composition code [9].

4) Illimitable Space System (ISS): Original ISS’s research-creation practices focused primarily on interactive multimodal installations and productions with the collaboration of local artistic troupes. It helped mobilizing traditional artists and makes them aware of the new technology in order to express themselves in the new form of artistic approach. It started off as a new HCI in the theatre concept and interactive documentaries and moved to performing arts and alternate realities. Various versions of Illimitable Space System exist for motion capture, signal processing, computer vision, projection mapping including LED control, real-time reaction and control for stage and beyond [29], [34], [41]–[43].
ISS and its open-source backend core OpenISS [44] rely on computer vision techniques and machine learning provided by OpenCV and MARF: motion capture libraries for Kinect depth cameras and others, sound control, input from voice and music, and augmented and virtual reality components to co-create either augmented performance or have an installation or film, or use as an education tool for artists [43] or children.

a) “Projected Reality”: We explore an idea of a scalable management, visualization, and evaluation of digital evidence in the context for cybercrime investigation with extensions to the interactive 3D documentary subsystem of the Illimitable Space System (ISSv1) [29]. These modifications would enable investigators to represent and create semantic links among digital evidence within an easy to use interface powered by multimodal interactions including but not limited to eye-gaze, gestures and navigational hardware. That work may scale when properly re-engineered and enhanced to act as an interactive “3D window into the evidential knowledge base grouped into the semantically linked “bubbles” visually representing the documented evidence. By moving such a contextual window, or rather, navigating within the theoretically illimitable space an investigator can sort out and reorganize the knowledge items as needed prior launching the reasoning computation. The interaction design aspect would be of a particular usefulness to open up the documented case knowledge and link the relevant witness accounts and group the related knowledge together. This is a proposed solution to the large-scale visualization problem of large volumes of “scrollable” evidence that does not need to be all visualized at once but behave like a snapshot of a storage deposit [1].

As an example, stills from the actual ISSv1 installation hosting multimedia data (documentary videos) users can call out by voice or gestures to examine the contents as in Figure 3. We propose to reorganize the latter into more structured spaces so that the investigators can create semantic links to group the relevant evidences together and for subsequent evaluation by the distributed GIPSY’s backend engine [1]. As exemplified in ISSv1 the interactions here are projected on a wall/screen or appear on a monitor. Currently, the viewable scene/window is sequentially loaded and unloaded from the viewing device (PCs, laptops, or VR headsets) to prevent memory overload. The access is on demand by the device and the design is similar to RAM swapping by an operating system to support virtual memory and particularly in this case a distributed storage with evidential data. Available gesture-based interactions using Kinect and similar depth cameras with OpenCV are the enabling HCI aspects for the investigator to link the evidential items in the 3D space. The gesture-based interactions provide optional assistance by the voice-based controls for speech processing and the corresponding commands to view the evidence in detail. Modern availability of VR headsets and VR phone applications make this process even more accessible, although the storage, space and bandwidth requirements have higher constraints, to begin with. In our general approach, we propose an architecture to enable interactive visual windowing into the digital evidence processing as an investigator aid tool. Thus, the preferred method of interaction during analysis and human insight phases prior to or after distributed processing of the evidence and event reconstruction algorithms.

b) Virtual and Augmented Reality: Virtual Reality (VR) [45]–[47] is defined in [48] as “a three-dimensional simulation of the real world or an imaginary world allowing the user to have a sense of physical presence and to manipulate 3D objects, in real-time, inside three-dimensional computer-generated environments.” In [49], authors point out the possibility of exhibiting concepts that a user might not be able to view otherwise and the immersive nature of VR can aid in education thus, can be inferred for investigators as well.

Augmented Reality (AR) has to do with overlaying virtual objects on top of the real ones and a possibility with gesture or gaze based interaction with these objects while maintaining a grasp on the real world without complete immersion (avoiding nausea and other related VR issues).

The real benefit is when both techniques can be combined. ISSv3 in our case was being developing incorporating augmented and virtual reality techniques [7], [34]. We experimented in doing both mobile and desktop version of the mixed reality documentary and recording functionality in Unity some of which is visualized in Figure 4 and in Figure 5.

IV. CONCLUDING REMARKS

Moving towards our goal to have a visual 3D DFG-based tool that can model FORENSIC LUCID case specification, and above discussed choices that provide the abilities to do the same in their own ways we attempt to build upon related research work in this area. However, we do consider the potential of the recent work in virtual reality and augmented reality along with different multimodal interaction techniques that seem most consistent with our aim. So far, Ding’s work on Graphviz, Puckette’s PureData, BPEL and the ISS have drawn our interest and all of them are sound and formally backed standards with some exposure in the industry. While the others may require additional work to specify the credibility and correctness of the bidirectional translation between 3D DFG visualization and FORENSIC LUCID [9].

The drawbacks of PureData and Graphvizs dot are that their languages lack formal semantics specifications with a few semantic notes along with lexical and grammar related structures [17]. Thus, employing any or all of these will require us to provide translation rules and their equivalent semantics to FORENSIC LUCID as in Jarraya work that provides translations between the UML2.0/SysML state/activity diagrams and probabilities in [50] when translating to PRISM [9]. ISS is the most scalable approach that can aggregate all the others, but requires significant number of modifications. Given recent advancements in ISSv2 and ISSv3 referenced above including both AR/VR interactions, ISS makes this approach even more appealing and feasible than previously stated [9].

3http://vimeo.com/51329588
REFERENCES


Fig. 3. Interactive documentary using Illimitable Space System (ISSv1) visualization and management [29]
Fig. 4. ISSv3 Examples of the VR environment and some digital content


Fig. 5. Interactive documentary using Illimitable Space System (ISSv3) VR/AR [34]


Improving Multiple Object Tracking with Optical Flow and Edge Preprocessing

David-Alexandre Beaupré, Guillaume-Alexandre Bilodeau
LITIV lab., Dept. of Computer & Software Eng.
Polytechnique Montréal
Montréal, Canada
{david-alexandre.beaupre, gabilodeau}@polymtl.ca

Nicolas Saunier
Dept. of Civil, Geo and Mining Eng.
Polytechnique Montréal
Montréal, Canada
nicolas.saunier@polymtl.ca

Abstract—In this paper, we present a new method for detecting road users in an urban environment which leads to an improvement in multiple object tracking. Our method takes as an input a foreground image and improves the object detection and segmentation. This new image can be used as an input to trackers that use foreground blobs from background subtraction. The first step is to create foreground images for all the frames in an urban video. Then, starting from the original blobs of the foreground image, we merge the blobs that are close to one another and that have similar optical flow. The next step is extracting the edges of the different objects to detect multiple objects that might be very close (and be merged in the same blob) and to adjust the size of the original blobs. At the same time, we use the optical flow to detect occlusion of objects that are moving in opposite directions. Finally, we make a decision on which information we keep in order to construct a new foreground image with blobs that can be used for tracking. The system is validated on four videos of an urban traffic dataset. Our method improves the recall and precision metrics for the object detection task compared to the vanilla background subtraction method and improves the CLEAR MOT metrics in the tracking tasks for most videos.

Index Terms—object detection, object tracking, edges, optical flow, urban scenes

I. INTRODUCTION

Object detection is a fundamental task in the field of computer vision. It is a necessary step in traffic surveillance in order to collect traffic data and analyze road user behavior. It is used to extract image regions that correspond to the objects of interest. Improving the detection of the cars, cyclists and pedestrians can help to improve another important task, which is multiple object tracking (MOT). Many trackers, for instance Urban Tracker (UT) [1], [2] and Multiple Kernelized Correlation Filter Tracker (MKCF) [3], use foreground blobs $B_i$ from background subtraction as an input to track the objects in the video because these detections are generic and do not assume any prior classes. UT is a more complex tracker using feature points and a state machine to keep track of the different objects while MKCF is a fast tracker with simpler data association for tracking multiple objects. Yet, both depend on the quality of background subtraction.

There are many problems with the images produced by background subtraction methods in an urban environment. The first one is foreground blob merging, which occurs when two road users occlude each other or are close to each other as in figure 1. Even if trackers have ways of dealing with these occlusions, we found that it is advantageous to explicitly detect the different occluding objects prior to tracking. Another problem of background subtraction is the case of fragmentation where a unique object is separated in multiple smaller blobs. Once again, we found that it is easier to explicitly merge the fragmented blobs than to let the tracker decide if the multiple blobs were part of the same object or not. Another problem of background subtraction is caused by

Fig. 1. Example of segmentation from our method and ViBe method. a) Original image of an occlusion between cars from the Sherbrooke video sequence, b) original image of an occlusion between pedestrians from the Rouen video sequence, c) and d) respective segmentations with ViBe, and e) and f) respective segmentations from our method. Note that since the object masks that we output are combinations of bounding boxes, objects are just segmented coarsely.
shadows, mainly by pedestrians’ shadows. In fact, foreground blobs will often include the shadows of the objects, which can lead to a tracking box that is much larger and not as precise as the one without shadows, or to merging different objects in the same blob. Our method is able to eliminate most of the unwanted shadows which lead to a more precise detection. One last problem is that the background subtraction blobs are generally much bigger than the real size of the objects that we want to detect. Our proposed method was able to effectively adjust the size of the proposed blobs which results in better recall in the detection metrics.

Our method uses background subtraction [4], optical flow [5] and edge processing in order to create a new binary image of foreground blobs. Background subtraction is used to locate the regions of interest (RoIs), which are the location of the foreground blobs in the current frame. The dense optical flow, with a patch size of 8, is then computed for each blob $B_i$ in the given frame. The motion vectors of the different regions and the relative distance between the regions are compared to merge the blobs that are very likely of being fragmented regions of the same object. The optical flow computed at each frame (with the previous one) is also used to separate objects merged in the same blob that are moving in opposite direction. We use the Canny edge detector [6] on both the blobs of the source frame and the scene background image (see section III-A) to obtain the edges of the foreground objects (we want to eliminate background edges that might be in a foreground blob i.e. road markings near pedestrians). This last step allows adjusting the size of the objects, separating close objects that appeared as one blob in background subtraction and eliminating noise. With all this information, our novel method generates a new binary image with processing steps that handle fragmentation, merging and remove noise while giving a more precise segmentation.

The organization of the paper is the following: in section II, we discuss related work. In section III, we present our new method consisting of the foreground image, merging of similar optical flow regions, separation of opposite flow regions, edge processing and creation of the new binary image. In section IV, we present our results and finally, in section V, we conclude this paper.

II. RELATED WORK

Many methods can be used in order to extract the object RoIs in a given frame. Objects proposal methods like [7], [8] can get good recall results given a large number of proposals. Also, these methods do not require the input to be a video since they propose boxes based on their “objectness”. The downside of object proposal methods is to filter the thousands of initial proposals to extract the real objects in our frame, which is often less than twenty in tracking tasks, and to make sure that every object only has one bounding box. The challenge of keeping the best box around each object while keeping high recall is difficult to achieve for the purpose of a tracking.

Another method to extract RoIs is optical flow as in [5], [9]. Optical flow is the process of computing the motion of every pixel between two consecutive frames. By grouping pixels with similar motion, this results in blobs of pixels for each object with different motion. Thus, these methods are very good at detecting moving objects, but segmenting individual objects from a group can be more difficult, especially if they are moving in the same direction. In fact, two objects very close to one another will be considered in the same motion flow blob since their flow vectors will be very similar. In addition, these methods cannot detect still objects. However, two close objects going in opposite directions are very easy to separate with optical flow methods as stated earlier.

Recently, deep learning methods have achieved great results in object detection as seen in [10], [11] while being able to make those detections almost in real time. However, these neural networks must be trained on every class we want them to detect, which can take up a lot of time and resources. They cannot detect objects from unexpected classes.

Finally, another traditional approach to obtain object RoIs is background subtraction, like with ViBe [4] and SubSENSE [12]. In this case, RoIs are the results of the differences between the current frame and a background frame model. These methods can detect objects from any class. However, they can be sensitive to camera motion and shadows. Also, they cannot resolve merging caused by occlusion or proximity. However, they are very appealing for tracking in urban scenes because of the unknown variety of objects of interest these scenes may contain.

As mentioned above, we chose ViBe [4] to provide us with the initial RoIs. ViBe is a background subtraction method that keeps track of the values of each pixels in the past to determine if a pixel in the current frame is in the foreground or the background. For a given frame, every blob $B_i$ produced by ViBe will be fed into our algorithm in order to improve the detection of objects.

III. METHODOLOGY

This section presents the different steps of our method as shown in figure 2. These steps are simple operations, using optical flow and edge analysis.

A. Background image

The first operation is to accumulate a color background image from the video sequence. This will become useful in the edge processing step (see section III-D) because we will be able to filter out most of the background edges that may be included in foreground edges e.g. road marking that are not of interest [13]. The background image $A_i$ is given by

$$A_i = \alpha * I + (1 - \alpha) * A_{i-1},$$

where $\alpha$ is an accumulation rate and $I$ is an input image. In the experiments, $\alpha = 0.01$, which means that each new frame has a weight of 0.01 in the running average and the mean image has a weight of 0.99.
B. Merging foreground blobs

The first step to process a frame is to check if we can merge any foreground blobs that satisfy the three following conditions (C1), (C2) and (C3) presented below. If this is the case, a new blob \( B_k \) is formed from the union of the blobs \( B_i \) and \( B_j \). The union operation in our case takes the smallest box that frames both blobs \( B_i \) and \( B_j \).

The first condition (C1) is given by

\[
d(B_i, B_j) \leq T_M,
\]

where \( d(B_i, B_j) \) is the minimum distance between the pixels of two blobs \( B_i \) and \( B_j \). This distance must be smaller than a threshold \( T_M \) that can be modified, but we found experimentally that a distance of 7 pixels is a good compromise because we need to merge objects of various sizes (car and pedestrian dimensions vary in different datasets).

The second condition is based on intervals of the magnitude \( \text{mag()} \) of the optical flow of blobs, built as the mean magnitude \( \text{mag}(B_i) \) plus or minus one standard deviation \( \text{std}(\text{mag}(B_i)) \). The lower and upper bounds for the blobs \( B_i \) and \( B_j \) can be written as

\[
\begin{align*}
\min_{B_i} &= \frac{\text{mag}(B_i) - \text{std}(\text{mag}(B_i))}{B_i} \\
\max_{B_i} &= \frac{\text{mag}(B_i) + \text{std}(\text{mag}(B_i))}{B_i} \\
\min_{B_j} &= \frac{\text{mag}(B_j) - \text{std}(\text{mag}(B_j))}{B_j} \\
\max_{B_j} &= \frac{\text{mag}(B_j) + \text{std}(\text{mag}(B_j))}{B_j}.
\end{align*}
\]

We can now define the domain of possible values for each blob as

\[
\begin{align*}
\text{dom}_{B_i} &= [\min_{B_i}, \max_{B_i}] \\
\text{dom}_{B_j} &= [\min_{B_j}, \max_{B_j}].
\end{align*}
\]

The second condition (C2) is then given by

\[
\text{dom}_{B_i} \cap \text{dom}_{B_j} \neq \emptyset.
\]

This condition verifies that both domains \( \text{dom}_{B_i} \) and \( \text{dom}_{B_j} \) have at least one value in common. The third condition (C3) is

\[
|\text{ang}(B_i) - \text{ang}(B_j)| \leq A_T.
\]

This condition checks if the angles \( \text{ang()} \) of the optical flow for the blobs \( B_i \) and \( B_j \) are approximatively in the same direction. We take the angle of the optical flow at the center point of each blob for the comparison with the threshold \( A_T \). The value of \( A_T \) is \( \frac{\pi}{2} \). This means that we sometimes merge (i.e. union operation) two foreground blobs that should not have been merged, but we prefer to err on the side of over-merging because it is possible to separate objects at a later stage of our method. At this step, we also save which RoIs were modified and store them in a map that will be used in section III-E. These new foreground blobs will be the new RoIs for the next steps.

C. Flow separation

The purpose of this step is to separate foreground blobs that contain two objects going in opposite directions. For each foreground blob \( B_i \) in the image, we apply the k-means clustering algorithm to the optical flow vectors. We chose \( k = 3 \) because when there are two objects moving in opposite direction, the segmentation of these objects results in one cluster for the background, and the other two clusters as the distinct objects. We fit bounding boxes \( r_1 \), \( r_2 \) and \( r_3 \) around each of the three clusters. We can then compute the ratio, \( \text{ratio}_{\text{int}}(i,j) \), of the intersection of the boxes over the smallest area \( \text{min}_\text{area} \) of the two boxes as

\[
\text{ratio}_{\text{int}}(i,j) = \frac{r_i \cap r_j}{\text{min}_\text{area}(r_i, r_j)} \quad \text{for every } i \neq j.
\]

Since there are only three boxes, this gives us three \( \text{ratio}_{\text{int}}(i,j) \) for all pairs \( (r_i, r_j) \). We compare each \( \text{ratio}_{\text{int}}(i,j) \) against a threshold \( T_{\text{int}} \) of 0.40 and if the ratio is smaller, we check if the boxes \( r_i \) and \( r_j \) are going in opposite directions using the negation of the third condition expressed by equation 7. For a given blob \( B_k \), we will keep two bounding boxes \( (r_i \text{ and } r_j) \) if both conditions were met. If not, we simply fit a bounding box around the original foreground blob \( B_k \) for this particular region and ignore \( r_1 \), \( r_2 \) and \( r_3 \). During this process, we save every RoI that has been split in two in a map that will be used in section III-E.

D. Edge processing

This is where we use the background image \( A_t \) the pixels included in the blob \( B_i \) followed by an edge detection to
form a representation \( E_A \) of the same size as \( B_i \). We do the same thing for the current image \( I \), forming \( E_I \) of the same size. Edges of \( E_A \) and \( E_I \) are obtained using the Canny edge detector [6] using for threshold the values of \( T_H \) and \( T_L \) given by

\[
T_H = (1 + \sigma) \times \text{median}(I) \tag{9}
\]

\[
T_L = (1 - \sigma) \times \text{median}(I) \tag{10}
\]

The value of \( \sigma \) was determined experimentally and set to \( \frac{1}{2} \). The value of \( \text{median}(I) \) is computed for each image \( I \) and corresponds to the median pixel value in the grayscale image.

With the two edges representations for \( E_A \) and \( E_I \), we can make a logical xor operation for each pixel of the regions. This will eliminate the background edges from the foreground blob edges, leaving us with only edges of the foreground \( E_F \). Moreover, this enables us to eliminate foreground blobs that were in fact background objects. This increases the precision of our method and adjusts better the size of the detection boxes to the objects.

Also, the edge processing step can separate two objects or more that are in the same foreground blob \( B_i \). This operation will also separate blobs that should not have been merged previously. When we obtain our edge representation image \( E_F \), we form groups with the pixels based on distance in order to detect if there are more than one object in the current blob \( B_i \). To do this, we choose a random edge pixel \( e_i \) and find every connected edge pixel with a Manhattan distance of less than 3 pixels to form an edge group \( G_i \). We repeat this process until every edge pixel is a member of an edge group. When this is the case, we find the biggest bounding box \( r_i \) for every edge group \( G_i \). The number of bounding boxes corresponds to the number of distinct objects in a foreground blob \( B_i \). Once again, we store each RoI that has been split into multiple regions in a map for the next step.

E. Decision algorithm

At this point, the information we have is three maps: one from the merging of regions by optical flow (see section III-B), another from the separation of regions (see section III-C) and the last one from the edge analysis (see section III-D). This means that we have, in the best case, two box proposals for each RoI (from the separation map and edge analysis map), but there can be more than that if any of the processing returned more than one box. We present now the algorithm to make our decision regarding which boxes to keep for the final foreground image.

The first thing we check is if an RoI has been modified by the flow separation step (see section III-C). If this is the case, we keep the two boxes returned by the optical flow because we are sure that the two objects were moving in opposite direction. The second verification is if an RoI has been modified both by the flow merging (see section III-B) and the edges (see section III-D), we keep the boxes from the edges processing, because if the merging was successful, the edges will give us one box around the object and if not, the edges will give us multiple boxes depending on the number of objects. The third check that we make is on the number of boxes that the edge processing step returned. If this number is greater or equal than four, we ignore them and simply keep the one box proposed by the optical flow. This is because it is more likely that the edges over-separated one single object into multiple ones and will lead to a bad detection. The fourth verification covers the situation where the edges proposed two boxes, \( e_i \) and \( e_k \), and the optical flow only one, \( f_k \). This is the hardest case because we do not know if there are truly two objects in the RoI or if, for instance, the edge processing separated the shoes of a pedestrian from the rest of its body. We compute the area ratio, \( \text{ratio}_{area} \), from both processes in order to make our decision:

\[
\text{ratio}_{area} = \frac{\text{area}(e_i \cup e_j)}{\text{area}(f_k)}. \tag{11}
\]

We keep the the single box \( f_k \) from the optical flow processing.

---

Fig. 3. Some examples of our method (first line) and the ViBe algorithm (second line) with both trackers (a), (b), (e) and (f) is MKCF while (c), (d), (g) and (h) is UT. First column is an example of how our method is able to separate close objects in the Rene video. Second column shows how objects moving in opposite directions are easier to separate with our method on the Rouen video. Third column demonstrates how our method is able to keep whole objects in the case of occlusion by other static objects in the Sherbrooke video. Fourth column is an example of how our method gives smaller boxes around the pedestrians in the St-Marc video.
if the ratio is smaller or equal to 0.65, a parameter determined experimentally. Otherwise, we keep both boxes $e_i$ and $e_j$ from the edges processing step. Finally, in all the other situations, we simply favor the edge boxes over the ones from the optical flow because they tend to be smaller that their counterpart.

F. New final foreground image

The process to create the new binary image is quite simple. We start by creating an image made only of zero valued pixels. After that, we do a xor operation between the image and the box proposals, which are represented by white pixels. This means that when two objects share an intersection, the pixels at the intersection become black and this leads to better detection inputs for the trackers as the objects are separated. The last operation is to increase the size of those intersections by one pixel in every direction (dilation operation) since it facilitates the segmentation for the trackers. Note that since the resulting object masks are combinations of bounding boxes, objects are just segmented coarsely.

IV. RESULTS

In order to evaluate our proposed method, we used the publicly available UT dataset [1] containing four video sequences of urban mixed traffic. The videos contain pedestrians, cyclists and cars. There were multiple frames that were annotated in each sequence so we could test our method. The evaluation of our method was made in two steps. First, we compared the object detection performance of our method versus the original background subtraction method. Second, we showed how our method can improve the MKCF tracker [3], a tracker with a simple data association scheme, and the Urban Tracker (UT) [1], a tracker with a more complex data association scheme, when given the new foreground images compared to the ones produced by the ViBe method [4]. Our method improves object detection in all videos, and tracking results for most videos.

The code for our method can be downloaded from https://github.com/beaupreda.

A. Evaluation methodology

For the evaluation of our method for the object detection task, we used the Intersection over Union (IoU) metric between the detected bounding boxes and the ground-truth bounding boxes. Then to evaluate our method for the tracking task, we used the tools provided with the Urban Tracker dataset [1]. These tools compute the CLEAR MOT [14] metrics. The multi-object tracking accuracy (MOTA) takes into account the false positives, the ID changes and the misses. The multi-object tracking precision (MOTP) measures the average precision of object matches at each instant. We evaluated our method with an IoU of 30 %. We decided not to use the classical IoU of 50 % because when evaluating with the trackers, most of the CLEAR MOT metrics were negatives as the videos are difficult. Also, when looking at the MKCF and UT papers, we found that they were using distances between the centroid of the boxes, and that the values of these distances were quite permissive. For instance, in the Rouen video, the distance threshold was of 164 px, which is 20.5 % of the width and 27.3 % of height of the video frame. This distance is generous in a way that objects moderately far away can still be considered matched and tracked. Also, the absolute distance does not consider the size of the objects. For example, there are cars and pedestrians in the Rouen video, so a distance of 164 px might be reasonable for cars that are bigger than pedestrians generally, but not for pedestrians. By using an IoU of 30 %, we remain flexible for the tracking accuracy while considering the relative size of the different objects that are tracked. We ran the code of both trackers to obtain the results since we changed the evaluation metric. Results are thus different from the ones reported in their respective papers.

For the detection, we also used an IoU of 30 % because we wanted to remain consistent between our two evaluations. Even when we tested with an IoU of 50 %, our method had better recall and precision than the original background subtraction.

B. Experimental results

For the detection task, the results can be found in table I. Our method shows improved results for both the precision and recall across all four videos of the Urban Tracker dataset. The most significant improvement is for the Rouen video, followed by Sherbrooke. This can be explained by the fact that a lot of objects are traveling in opposite directions in both of these videos. We are thus able to better separate objects.

The quantitative results for the MKCF tracker and UT are presented in table II. For the Sherbrooke video sequence, we see that our method is able to improve both the MOTA and the MOTP for both trackers. The MOTA is increased significantly while the MOTP has a more modest improvement. This is due to the fact that the difficulty of this video sequence comes from the large number of cars moving in opposite directions. Our method is able to separate those objects with the optical flow and give an image segmented with each car individually while the original background subtraction merges cars going in opposite direction in the same blob.

The Rene-Levesque video sequence contains a large number of cars and the camera is far from the scene, which means that the objects of interest are all very small. We increase the MOTA of UT by 5 % and the MOTP by 12 %. Our method is able to improve UT because it is able to separate adjacent cars and because the edge processing reduces the size of the boxes. This leads to a more precise tracking. We were also able to improve both metrics with the MKCF tracker, but to do this, we had to reduce the minimum blob size parameter (100 pixels) in the tracker algorithm because, as mentioned earlier, our method reduces the size of the boxes and many proposed boxes were smaller than the threshold originally used by the algorithm. Thus, there were no tracker on some objects which led to poor results. Using the new parameters, we were
TABLE I
OBJECT DETECTION RESULTS OF ViBe AND OUR PROPOSED METHOD ON THE UT DATASET. PRECISION AND RECALL SHOULD BE HIGH. BOLDFACE: BEST RESULTS

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Recall (ViBe)</th>
<th>Recall (Ours)</th>
<th>Precision (ViBe)</th>
<th>Precision (Ours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sherbrooke</td>
<td>0.606</td>
<td>0.752</td>
<td>0.681</td>
<td>0.739</td>
</tr>
<tr>
<td>Rene-Levesque</td>
<td>0.812</td>
<td>0.855</td>
<td>0.612</td>
<td>0.654</td>
</tr>
<tr>
<td>Rouen</td>
<td>0.734</td>
<td>0.834</td>
<td>0.724</td>
<td>0.823</td>
</tr>
<tr>
<td>St-Marc</td>
<td>0.684</td>
<td>0.754</td>
<td>0.415</td>
<td>0.458</td>
</tr>
</tbody>
</table>

TABLE II
TRACKING RESULTS OF MKCF AND UT USING ViBe AND OUR DETECTIONS ON THE UT DATASET. MOTA AND MOTP SHOULD BE HIGH. BOLDFACE: BEST RESULTS

<table>
<thead>
<tr>
<th>Dataset</th>
<th>MOTA</th>
<th>MOTP</th>
<th>MOTA</th>
<th>MOTP</th>
</tr>
</thead>
<tbody>
<tr>
<td>MKCF</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ViBe</td>
<td>0.317</td>
<td>0.523</td>
<td>0.535</td>
<td>0.576</td>
</tr>
<tr>
<td>Ours</td>
<td>0.334</td>
<td>0.424</td>
<td>0.5309</td>
<td>0.660</td>
</tr>
<tr>
<td>UT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ViBe</td>
<td>0.404</td>
<td>0.690</td>
<td>0.576</td>
<td>0.590</td>
</tr>
<tr>
<td>Ours</td>
<td>0.565</td>
<td>0.613</td>
<td>0.582</td>
<td>0.705</td>
</tr>
</tbody>
</table>

For the Rouen video, we improve the results of the MKCF tracker in terms of both MOTA and MOTP, while we only improve the MOTP for UT. The difficulty of this dataset is coming from the number of pedestrians crossing the street. Once again, these pedestrians are going in opposite direction but there are also some who walk at the same speed and close to one another. These pedestrians are the hardest to detect individually. Our method, which can segment pedestrians going in opposite directions performs well with the simpler MKCF tracker because it helps the tracker during occlusions. UT remains better with the original background subtraction because our method will sometimes merge two pedestrians going in the same direction.

For the St-Marc video, we also had to change the minimum blob size parameter (700 pixels) in the MKCF tracker for the same reasons as stated above. The MOTA was improved while producing a slightly lower MOTP. The same logic can be transferred for UT where the MOTA was slightly improved and the MOTP decreased. We address the issue of occlusion from a traditional background subtraction method to obtain our RoI and then, with the help of the optical flow and edge preprocessing, we are able to deal with the fragmentation caused by the background subtraction and effectively separate objects that are either too close to one another or objects that are going in opposite directions. This method improves both the recall and the precision in the object detection task when compared to the original foreground image. It also improves the CLEAR MOT metrics for both trackers for most of the tested videos.

REFERENCES

Cubes3D: Neural Network based Optical Flow in Omnidirectional Image Scenes

1st André Apitzsch  
Dept. of Information Technology  
Chemnitz University of Technology  
Chemnitz, Germany  
andre.apitzsch@etit.tu-chemnitz.de

2nd Roman Seidel  
Dept. of Information Technology  
Chemnitz University of Technology  
Chemnitz, Germany  
roman.seidel@etit.tu-chemnitz.de

3rd Gangolf Hirtz  
Dept. of Information Technology  
Chemnitz University of Technology  
Chemnitz, Germany  
g.hirtz@etit.tu-chemnitz.de

Abstract—Optical flow estimation with convolutional neural networks (CNNs) has recently solved various tasks of computer vision successfully. In this paper we adapt a state-of-the-art approach for optical flow estimation to omnidirectional images. We investigate CNN architectures to determine high motion variations caused by the geometry of fish-eye images. Further we determine the qualitative influence of texture on the non-rigid object to the motion vectors. For evaluation of the results we create ground truth motion fields synthetically. The ground truth contains cubes with static background. We test variations of pre-trained FlowNet 2.0 architectures by indicating common error metrics. We generate competitive results for the motion of the foreground with inhomogeneous texture on the moving object.

Index Terms—Optical Flow, Omnidirectional Camera, Convolutional Neural Networks, Deep Learning

I. INTRODUCTION

Even for large displacements the estimation of optical flow is still a challenge in computer vision. Common benchmark datasets like Middlebury Flow [1], KITTI Flow [2] or Sintel [3] address the environmental conditions in appearance change, large motions and homogeneous image regions. In this paper we investigate two central aspects: first, the generation of ground truth for optical flow. Second, the exploration of different architectures of correlating images using convolutional neural networks. As a result, we apply various combinations of architectures of FlowNet 2.0 [4] to estimate the pixel displacements depending on the regions of an image with omnidirectional camera geometry. A typical application for obtaining optical flow from image sequences is motion segmentation in indoor scenes with a fixed camera position.

The remainder of this paper is structured as follows: Section II presents previous research activities in estimation of the optical flow using deep neural networks. Section III illustrates the applied network architectures of the neural network as well as the characteristics of images from a fish-eye camera. Section IV explains how our ground truth is generated. Section V describes our experiments for computation of the optical flow and compares the results to our generated synthetic ground truth based on common error metrics. Section VI summarizes the paper’s content, concludes our observations and gives ideas for future work. The source code of our work is available at https://gitlab.com/auxilia/cubes3d.

II. RELATED WORK

While optical flow estimation on images from omnidirectional cameras is not well explored, a phase based method for optical flow estimation on perspective image data was provided by [5]. Reference [6] applied spatial filters (here: Gabor filters) to estimate valid component velocities with a recurrent neural network through the direct measurement of the phase linearity, which is a good indicator of the motion velocity. Beside the variational methods to compute optical flow from [7] which produce a dense flow field, the work of [8] concentrates on the optimization of the variational objective. The motivation for using global optimization methods is the limitation of the variational objective to small places to compute optical flow for large pixel displacements. Adapting this approach to omnidirectional images helps to determine the correct motion fields.

In [9] a deep neural network with 3-D convolutions was trained end-to-end to perform prediction at voxel-level.
The architecture leads to competitive results in semantic segmentation, video coloring and optical flow estimation.

As an alternative approach, which we follow, the optical flow was interpreted as a learning problem called FlowNet [10] and FlowNet 2.0 [4] based on neural networks. The considerable improvements in quality and speed are caused by three major aspects: first, accurate training datasets (e.g. FlyingThings3D [11] and FlyingChairs [10]) are responsible for the success of supervised learning. Second, the stacked network architecture with image warping of the second image with intermediate optical flow. Third, the consideration of small pixel displacements by introducing a subnetwork.

III. OPTICAL FLOW ON OMNIDIRECTIONAL IMAGES

A. Omnidirectional Camera Model

The camera model describes the transformation of a 3-D scene point to the coordinates of a 2-D image point. We use the omnidirectional camera model with an equidistant projection, described in [12]. In general the radial projection of an incoming light ray $\Phi = (\theta, \phi)^T$ onto a virtual image plane with a distance of $f = 1$ to the projection center can be modelled with the radial projection function:

$$X_{\text{norm}} = F(\Phi) = \rho(\theta)u_r(\varphi)$$

where $\rho(\theta)$ is the radius of the projected image point with respect to the center of the image, $\theta$ is the angle between the incoming ray and the optical axis and $u_r(\varphi) = (\cos \varphi, \sin \varphi)^T$ is the unit vector in radial direction. The projection of the pinhole camera model has a singularity for $\theta = \pi/2$ leading to a limitation of a field of view smaller than 180°. Fish-eye lenses are designed to cover the full hemisphere in front of the camera. One of the projection functions is the equidistant projection:

$$\rho(\theta) = \theta$$

where the distance of the mapped image point to the center of the image is linearly proportional to the angle $\theta$ of the incoming ray.

B. Convolutional Neural Networks for Optical Flow

Computing optical flow in 2-D image space is the estimation of the motion vector of two consecutive frames. The idea is to learn input-output relations from a sufficient amount of labeled data with a convolutional neural network in the proposed FlowNet [10] and FlowNet 2.0 [4], respectively. The general working principle of the network architecture of FlowNet2, based on FlowNetCorr is shown in Fig. 2.

Finding correspondences by using correlation layer determines the differences between the resulted feature maps. The approach is similar to convolving an image patch with the difference to convolve the first with the second patch of the feature map. To consider the easiest step for two patches with a square patch of size $K := 2k + 1$, the correlation of two patches at $x_1$ in the first map and $x_2$ in the second feature map is shown as:

$$c(x_1, x_2) = \sum_{o \in [-k,k]} f_1(x_1 + o), f_2(x_2 + o)$$

where $f_1, f_2$ are the multi-channel feature maps. Instead of the convolution of an image with a filter, [3] describes the convolution between two feature maps.

The refinement step of the architecture is shown in Fig. 3, which is part of the FlowNet2-CSS-ft-sd architecture. The range of the movement is limited to avoid the computation of all patch combinations for efficient forward and backward passes.

One of the main differences between FlowNet and FlowNet 2.0 is the idea of stacking networks for different quantity of motions in combination with warping the second image.

C. Architectures of FlowNet 2.0

As our starting point we apply three different architectures of a deep neural network, namely FlowNet 2.0. The original FlowNet2, the FlowNet2-SD and the FlowNet2-CSS-ft-sd. While FlowNet2 leads to good results at flow on Sintel and Middlebury (average angular error on test) and a high average precision at Motion Segmentation, the FlowNet2-SD output comes up with smoother results with less noise at smaller displacements even without refinement [4]. We choose these architectures for the following reasons:

a) FlowNet2: We pick up the idea of stacked networks of the FlowNet2 network architecture to handle the pronounced high variations of displacement in omnidirectional images.

b) FlowNet2-SD: To handle small displacements, which are common by increased radius from the center of the fish-eye image, we compare the results of FlowNet2-SD with our ground truth.

c) FlowNet2-CSS-ft-sd: Here we consider the idea of training a network on a combination of Things3D and Chairs3DHom (Chairs Small Displacements Homogeneous, see [4]). The architecture decreases the impact of large displacements by a non-linearity of the endpoint error compared to the FlowNet2-SD architecture.

D. Optical Flow Color Coding

For visualization of the computed flow field, we use the color wheel of [3], which is shown in Fig. 4.

IV. GROUND TRUTH DATA

Creating a real-world database for pixel-accurate optical flow ground truth is difficult [4]. To control various conditions in obtaining ground truth, namely scene illumination, pixel accuracy, non-rigid movements of the scene we generate synthetic ground truth data as shown in Fig. 16 using Blender. The synthetic data are image sequences of a moving cube with homogeneous and inhomogeneous
FlowNetCorr

Figure 2: Correlated FlowNet Architecture (FlowNetCorr) by [10]. Creating two parallel processing streams to correlate the feature-maps on pixel level and combine them on a higher level. Finding correspondences is realized through a correlation layer by comparing patches of two feature maps.

Figure 3: In order to avoid pooling layers leading to a reduced resolution of the feature maps, a refinement of FlowNet architecture was done by [10]. The idea behind the refinement is to replace the pooling step with upconvolutional layers, as previously presented in [13]–[15]. The refinement is done with the help of a priori information from the previous layer with coarse optical flow, if available.

Figure 4: Flow field coding used in this paper, following [3]. The color indicates the direction of the motion vector. The saturation enhances with increasing radius, which leads to strong movement at fields with high saturation.

colored surfaces captured at 24 frames per second by a virtual omnidirectional camera with equidistant projection (see section [11]). The camera’s fish-eye lens has a field of view of 180° and an azimuth of 360°. For the ground truth flow we use the speed vector Blender calculates for each shader data. The cube moves along three different paths while staying in a fixed z-plane which is parallel to the image plane. One path goes from left to right through the center of the image (linec), another one parallel to the first one but shifted to a different y plane (line); and the last path is a spiral non-uniform rational basis spline (NURBS) curve that starts in the center and ends at the border of the image (spiral). For each path we record three ground truth sequences with different uniform velocities (1, 2 and 4 unit of speed).

V. Experimental Results

To capture image data we use a fixed camera with known intrinsic and extrinsic calibration [17]. The sensor was mounted on a height of \(z = 2.5\) m and the cube has the dimension of \(2\) m \(\times\) \(2\) m \(\times\) \(2\) m. The center of the cube is always at \(z = 0\) m.

We compare the results of three network architectures of FlowNet 2.0 to our own generated ground truth. We pick up the idea of stacking networks from FlowNet 2.0. Estimating the optical flow on images from omnidirectional camera geometry means to deal with peculiarity of different quantity of motions depending on the location of the object in the image. The closer the object is located to the center of the image \((x/2,y/2)\) the bigger the apparent movement.

A. Qualitative Results

We formulate the selection of examples of the results in a way to determine different artifacts in our test data. The artifacts are induced by different characteristics, namely results of various architectures of FlowNet 2.0, the way we synthetically generate ground truth and the camera geometry. A well-working example is shown in Fig. [4]. Our investigations of different architectures based on FlowNet 2.0 induced by different experiments are shown in Fig. [5].
Figure 5: Qualitative results on our ground truth. First column constitutes the semi-transparent superimposed input images. Second column shows the ground truth. Columns 3 to 5 contain various architectures, namely FlowNet2 (stacked networks), FlowNet2-SD for small displacements and FlowNet2-CSS-sd-ft (Refinement of network architecture). See text for details. We choose the examples of different architectures and various experiments to discover their limits intentionally.

B. Quantitative Evaluation of Results

We evaluate the optical flow motion vector with respect to the synthetically generated ground truth by common error metrics, the average angle error (AAE) and average end point error (AEPE) [1]. The angle error (AE) was first introduced in [18] and describes the angle between a flow vector \((u, v)\) and a ground truth flow \((\tilde{u}, \tilde{v})\), in 3D space between \((u, v, 1)\) and \((\tilde{u}, \tilde{v}, 1)\). The AE can be computed by the inverse cosine of the dot product of the vectors divided through their magnitudes. Averaged over all pixels of an
image we get:

\[
AE = \frac{1}{N} \sum_{i=1}^{N} \cos^{-1} \left( \frac{1 + u_i \cdot \tilde{u}_i + v_i \cdot \tilde{v}_i}{\sqrt{1 + u_i^2 + v_i^2} \sqrt{1 + \tilde{u}_i^2 + \tilde{v}_i^2}} \right)
\]

The relative measure of performance ensures to avoid the division by zero for zero flows. By means of normalization, the AE penalizes errors in large flows less than errors in small flows. Current state-of-the-art flow algorithms present AAEs smaller than 2.0 degrees (equivalent to around 0.1 pixels) at Yosemite without clouds [19], with effectively no outliers.

An absolute error for quantitative comparison of the results is the AEPE from [20]. The endpoint error (EPE) is described as the absolute magnitude of difference vectors of the flow vector and the corresponding ground truth vector. Averaged over all pixels of an image we get:

\[
EPE = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(u_i - \tilde{u}_i)^2 + (v_i - \tilde{v}_i)^2}
\]

Evaluated results on AAE, AEPE and Flow errors are presented in Table I. For AAE and AEPE we compute the values of the optical flow from the variations of FlowNet 2.0 with respect to our generated synthetic motion vectors. The mean values of the errors are calculated per experiment (variable number of images) and per architecture. The best (i.e. lowest) values are highlighted in bold.

Taking into account the error metrics relative to the sum of pixel movement in terms of foreground and background, we follow the standard protocol from KITTI 2015 test set [1, 2, 4, 21]. Due to this protocol we report percentage values for movements above 3 pixels of Flow-background (Fl-bg), Flow-foreground (Fl-fg) and Flow-all (Fl-all) to characterize the outliers of static background, dynamic foreground and the whole image. Results are shown in Table I in column 5. On KITTI 2015 test dataset FlowNet 2.0 yields the best result in Fl-fg with 8.75%. The comparison to our own ground truth yields results of Fl-fg 22.85% at FlowNet2 architecture (per experiment values). FlowNet2-SD architecture produces Fl-fg 4.75%. Flownet2-CSS-fis-
sd leads to Fl-fg 5.07%. As long as our sequences contain a moving object with static background the Fl-bg values were close to zero while there is no motion in the background. The Fl-fg errors of 4.75% and 22.85% are more representative and allows the comparison to the error at common benchmark datasets.

**We observe the following.** (1) The average endpoint error of the sequence linec decreases with lower velocity. A plausible explanation is the higher proportion of foreground which is not evenly distributed because of the equidistant projection model of the sensor. (2) We observe noisy effects in Fig. 3 of results with the FN2-CSS-fis-
sd architecture (column (e)) on the bottom-right location of the fish-eye image in all our experiments. (3) Adding texture to the surface of the cube decreases the error of the foreground by about 10%. Our reference is the homogeneous (non-textured) surface at linear motion through the center of the image (see Fig. 3, rows 3 and 4). (4) As we expect, through the characteristics of the FlowNet2-SD architecture the determination of large displacements are difficult and leads to the highest values in Fl-fg at especially spiral-1 (Table I). (5) At large displacements the stacked architecture of FlowNet2 generates a noisy smearing effect. This is shown in Fig. 5 at column and row 3 and can be reduced by adding texture to the object.

**VI. CONCLUSION**

Beside the intensity of the motion the texture of the object plays a significant role for large displacements. In this paper we investigate the computation of optical flow by a state-of-the-art end-to-end learning approach, namely FlowNet 2.0 on omnidirectional image data. For the evaluation of the results we synthetically generate cubes with a homogeneous and inhomogeneous surface and transform the motion vectors of the objects into 2-D image space using the generic camera model. To step into the investigation of the determination of both, large and small displacements, either due to the characteristics of our camera model, or caused by an indeed high magnitude of the motion vector, we apply the initial FlowNet2 architecture. The results have shown, that flow errors in foreground are higher than 20% and lead to noisy blurring effects even for large displacements. To make the neural network more applicable for a wide range of displacements we see some prospective work on the training process. Is training on our simple geometric primitives qualitatively and quantitatively sufficient to achieve competitive results on testing of common benchmark datasets? Another application-driven work will be the motion segmentation based on FlowNet2’s motion vectors on real-world data.

Finally, we note the improvement of the cubes3D dataset. Modelling the synthetic environment closer to the real world helps us to distinguish between foreground and background robustly.

**References**


Table I: Quantitative results of AAE, AEPE and Flow errors with different FlowNet 2.0 architectures.

<table>
<thead>
<tr>
<th>Exp.</th>
<th>Architecture</th>
<th>AAE [°]</th>
<th>AEPE [px]</th>
<th>Fl-bg [%]</th>
<th>Fl-fg [%]</th>
<th>Fl-all [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>linec-4</td>
<td>FN2</td>
<td>0.034</td>
<td>2.73</td>
<td>0.91</td>
<td>55.42</td>
<td>3.38</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.030</td>
<td>2.73</td>
<td>0.06</td>
<td>44.66</td>
<td>2.53</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.079</td>
<td>2.13</td>
<td>1.22</td>
<td>37.59</td>
<td>3.43</td>
</tr>
<tr>
<td>linec-2</td>
<td>FN2</td>
<td>0.025</td>
<td>1.07</td>
<td>0.06</td>
<td>44.66</td>
<td>2.53</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.023</td>
<td>1.23</td>
<td>0.05</td>
<td>31.96</td>
<td>2.23</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.070</td>
<td>1.07</td>
<td>0.28</td>
<td>26.21</td>
<td>2.76</td>
</tr>
<tr>
<td>linec-1</td>
<td>FN2</td>
<td>0.019</td>
<td>0.43</td>
<td>0.11</td>
<td>22.85</td>
<td>1.96</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.014</td>
<td>0.48</td>
<td>0.04</td>
<td>22.83</td>
<td>1.90</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.063</td>
<td>0.47</td>
<td>0.52</td>
<td>20.46</td>
<td>2.25</td>
</tr>
<tr>
<td>line-4</td>
<td>FN2</td>
<td>0.011</td>
<td>0.08</td>
<td>0.12</td>
<td>40.86</td>
<td>0.34</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.003</td>
<td>0.58</td>
<td>0.05</td>
<td>59.03</td>
<td>0.42</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.053</td>
<td>0.57</td>
<td>0.29</td>
<td>33.40</td>
<td>2.76</td>
</tr>
<tr>
<td>line-2</td>
<td>FN2</td>
<td>0.011</td>
<td>0.04</td>
<td>0.11</td>
<td>28.97</td>
<td>0.26</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.002</td>
<td>0.02</td>
<td>0.03</td>
<td>26.93</td>
<td>0.20</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.052</td>
<td>0.08</td>
<td>0.28</td>
<td>10.04</td>
<td>0.33</td>
</tr>
<tr>
<td>line-1</td>
<td>FN2</td>
<td>0.011</td>
<td>0.03</td>
<td>0.09</td>
<td>24.94</td>
<td>0.22</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.002</td>
<td>0.01</td>
<td>0.02</td>
<td>4.75</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.052</td>
<td>0.07</td>
<td>0.26</td>
<td>5.09</td>
<td>0.29</td>
</tr>
<tr>
<td>spiral-4</td>
<td>FN2</td>
<td>0.025</td>
<td>1.71</td>
<td>0.49</td>
<td>81.46</td>
<td>2.05</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.016</td>
<td>2.14</td>
<td>0.10</td>
<td>94.29</td>
<td>1.81</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.069</td>
<td>1.67</td>
<td>0.75</td>
<td>78.14</td>
<td>2.29</td>
</tr>
<tr>
<td>spiral-2</td>
<td>FN2</td>
<td>0.018</td>
<td>0.54</td>
<td>0.24</td>
<td>60.99</td>
<td>1.55</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.013</td>
<td>0.89</td>
<td>0.10</td>
<td>76.79</td>
<td>1.64</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.060</td>
<td>0.57</td>
<td>0.50</td>
<td>48.43</td>
<td>1.71</td>
</tr>
<tr>
<td>spiral-1</td>
<td>FN2</td>
<td>0.015</td>
<td>0.24</td>
<td>0.14</td>
<td>37.77</td>
<td>1.05</td>
</tr>
<tr>
<td></td>
<td>FN2-SD</td>
<td>0.009</td>
<td>0.34</td>
<td>0.07</td>
<td>41.12</td>
<td>1.14</td>
</tr>
<tr>
<td></td>
<td>FN2-CSS-ft-sd</td>
<td>0.056</td>
<td>0.29</td>
<td>0.36</td>
<td>23.80</td>
<td>1.15</td>
</tr>
</tbody>
</table>

Evaluation of trackers for Pan-Tilt-Zoom Scenarios

Yucao Tang
University of Electronic Science and Technology of China
Chengdu, China
tang_yucao@qq.com

Guillaume-Alexandre Bilodeau
LITIV Lab.
Polytechnique Montréal
Montréal, Canada
gabilodeau@polymtl.ca

Abstract—Tracking with a Pan-Tilt-Zoom (PTZ) camera has been a research topic in computer vision for many years. Compared to tracking with a still camera, the images captured with a PTZ camera are highly dynamic in nature because the camera can perform large motion resulting in quickly changing capture conditions. Furthermore, tracking with a PTZ camera involves camera control to position the camera on the target. For successful tracking and camera control, the tracker must be fast enough, or has to be able to predict accurately the next position of the target. Therefore, standard benchmarks do not allow to assess properly the quality of a tracker for the PTZ scenario. In this work, we use a virtual PTZ framework to evaluate different tracking algorithms and compare their performances. We also extend the framework to add target position prediction for the next frame, accounting for camera motion and processing delays. By doing this we can assess if prediction can help to make long-term tracking more robust and help slower algorithm for keeping the target in the field of view of the camera.

II. EVALUATION FRAMEWORK AND METRICS

Chen et al. [2] proposed a C++ framework to evaluate in a reproducible way trackers in PTZ scenarios. Because of the online nature of this scenario, the authors proposed the use of a PTZ camera simulator that pans, tilts and zooms based on a spherical video captured offline. The simulator also includes relevant delays that result in drop frames if tracking takes too long and if the target has a large motion in the image plane. These delays are categorized into execution delay, motion delay and communication delay.

A. Simulator Configuration

We use the evaluation framework as indicated by Chen et al. [2]. However, since some tracker codes are in C++ and others in MATLAB, we adjusted how the delays are calculated to ensure fairness in execution time evaluation. We used the chronometer functions based on execution time in C++ 11 to calculate the time elapsed for processing frames by the trackers instead of the real-time clock. Some of the program running time is spent to read the image from disk drive and it was not included in the execution time. The motion delay is calculated by the time it takes for the simulated camera to tilt and/or pan. We decided that there would be no communication delays by supposing that the camera is not networked.

Some codes are written originally in MATLAB and we had to use the MATLAB engine to call the MATLAB function, or in other word, tracker interfaces. Minor changes have been made to the MATLAB source codes such as eliminating the display and drawing functions since those will affect the speed of processing but are totally unnecessary and are thus taken as irrelevant delays. Besides, after practical experiments, it turns out that calling MATLAB engine from C++ is actually in a time scale of milliseconds and this overhead can be actually in a time scale of milliseconds and this overhead can be neglected since the time to process frame by trackers is much longer than that delay.

B. Performance Evaluation

Chen et al. defined four performance metrics to evaluate the trackers [2]. These metrics are calculated in the image plane...
for the current camera viewpoint (i.e. the viewed subregion on the image sphere projected on the camera image plane). Let $C_{GT}$ and $C_{PT}$ be the ground-truth target center and the predicted target center, and $A_{GT}$ and $A_{PT}$ be the ground-truth target bounding box and the predicted bounding box, respectively. $C_{FOV}$ is the center of the camera image plane, or in other words, field of view (FOV). $TPE^i$ (Target Point Error) and $BOR^i$ (Box Overlapping Ratio) evaluate the quality of target localization and are defined as

$$TPE = |C_{GT} - C_{PT}| \tag{1}$$

and

$$BOR^i = \frac{A_{GT} \cap A_{PT}}{A_{GT} \cup A_{PT}} \tag{2}$$

$TPO^i$ (Target Point Offset) and $TF^i$ (Track Fragmentation) evaluate the quality of the camera control and are defined as

$$TPO^i = |C_{FOV} - C_{GT}| \tag{3}$$

and

$$TF^i = \begin{cases} 
1, & \text{if } TPE^i \text{ is invalid} \\
0, & \text{otherwise}
\end{cases} \tag{4}$$

$TF^i$ indicates whether the target is inside the camera FOV. $TPE^i$ and $TPO^i$ are invalid and assigned -1 if the target is outside the FOV. The overall metrics $TPE, TPO, BOR$ are the average metrics of the valid tracked frames. $TF$ is the sum of $TF^i$ divided by the number of processed frames. In the experiments, we report only $TF$ and $BOR$ as they are the most significant metrics. Besides $TPE$ and $BOR$ have similar purpose, and so are $TPO$ and $TF$.

III. TARGET POSITION PREDICTION

In [2], the authors made the remark that for a tracker to be successful, it should be either very fast, or should use some kind of target position prediction to keep the target close to the FOV of the camera. To assess the practicality of predicting a target position based on previous track information, we implemented three target motion models. Since there will be a delay between frames, it is necessary to predict the object position on the image sphere and move the camera accordingly so that the target appears in its FOV. Therefore, the prediction must account for the motion of the target and the motion of the camera. For calculation, let a target in the first frame appear at point $P_0$ (on the spherical image). At next frame, the target moves to $P_1$ (again on the spherical image). We can calculate its speed as $V_0 = P_1 - P_0/t_1 - t_0$. Then in the third frame if the target moved from $P_1$ to $P_2$ its speed would then be $V_1 = P_2 - P_1/t_2 - t_1$. Thus, a basic classical mechanics model can be used to estimate the next position in the fourth frame.

The position prediction should locate the target near the image center as much as possible. By knowing the motion of the target, it is possible to predict where it will be later after a delay $\Delta_t$. $\Delta_t$ should account for the processing time of the current frame in addition to the time it takes for the camera to move. We experimented with three motion models to obtain the target motion ($\Delta_d$) between two instants:

- **Model 1:** Object is moving at a constant speed and uses the velocity of last instant

$$\Delta_d = V_1 \times \Delta_t \tag{5}$$

- **Model 2:** Object is moving at a constant speed and uses mean velocity in last two instant.

$$\Delta_d = \frac{(V_1 + V_0)}{2} \times \Delta_t \tag{6}$$

- **Model 3:** Object can accelerate

$$\Delta_d = \frac{A \times \Delta t^2}{2} + V_1 \times \Delta t \tag{7}$$

where $A = V_1 - V_0/t_1 - t_0$ is the acceleration.

IV. TESTED TRACKERS

Among the 19 trackers we tested, six trackers are variations of correlation filters: KCF, SRDCF, SWCF, DSST, DFST and sKCF. Two trackers combine correlation filter outputs with color: STAPLE and STAPLE+. One is based on structured SVM: STRUCK. Two trackers are based purely on color: DAT and ASMS. One tracker is based on normalized cross-correlation: NCC. Two trackers are based on boosting: MIL and BOOSTING. One tracker is based on optical flow (ME-DIANFLOW) and one tracker includes a detector (TLD). Two trackers can be categorized as part-based: DPCF and CTSE. Another one combines many trackers in an ensemble: KF-EBT. Below, we briefly describe the trackers. More details can be found in the original papers describing each of them.

1) Kernelized Correlation Filter tracker (KCF) [3] KCF is operating on HOG features. It localizes target with the equivalent of a kernel ridge regression trained with sample patches around the object at different translations. This version of the KCF tracker includes multi-scale support, sub-cell peak estimation and a model update by linear interpolation.

2) Spatially Regularized Discriminative Correlation Filter Tracker (SRDCF) [4] This tracker is derived from KCF. It introduces a spatial regularization function that penalizes filter coefficients residing outside the target area, thus solving the problems arising from assumptions of periodicity in learning the correlation filters. The size of the training and detection samples can be increased without affecting the effective filter size. By selecting the spatial regularization function to have a sparse discrete Fourier spectrum, the filter is optimized directly in the Fourier domain. SRDCF employs also Color Names and greyscale features.

3) Spatial Windowing for Correlation Filter-Based Visual Tracking (SWCF) [5] This tracker is derived from KCF. It predicts a spatial window for the observation of the object so that the correlation output of the correlation filter as well as the windowed observation are improved.
Moreover, the estimated spatial window of the object patch indicates the object regions that are useful for correlation.

4) Discriminative Scale Space Tracker (DSST) [6] DSST extends the Minimum Output Sum of Squared Errors (MOSSE) [7] tracker with robust scale estimation. DSST also learns a one-dimensional discriminative scale filter which is used to predict the target size. The intensity features used in MOSSE [7] tracker are combined with a pixel-dense representation of HOG features.

5) Dynamic Feature Selection Tracker (DFST) [8] DFST is a visual tracking algorithm based on the real-time selection of locally and temporally discriminative features. DFST provides a significant gain in accuracy and precision with respect to KCF by the use of a dynamic set of features. A further improvement is given by making micro-shifts at the predicted position according the best template matching.

6) Scalable Kernel Correlation Filter with Sparse Feature Integration (sKCF) [9] This tracker is derived from KCF. It introduces an adjustable Gaussian window function and a keypoint-based model for scale estimation. It deals with the fixed window size limitation in KCF.

7) Sum of Template And Pixel-wise LEarners (STAPLE) [10] STAPLE combines two image patch representations that are sensitive to complementary factors to learn a model that is robust to both color changes and deformations. It combines the scores of two models in a dense window translation search. The scores of the two models are indicative of their reliability.

8) An improved STAPLE tracker with multiple feature integration (STAPLE+) STAPLE+ is based on the STAPLE tracker and improves it by integrating multiple features. It extracts HOG features from color probability maps to exploit color information better. The final response map is thus a fusion of scores obtained with different features.

9) STRUCTured output tracking with Kernels (STRUCK) [11] This is a framework for adaptive visual object tracking. It applies a support vector machine which is learned online. It introduces a budgeting mechanism that prevents the unbounded growth in the number of support vectors that would otherwise occur during tracking.

10) Distractor Aware Tracker (DAT) [12] This is a tracking-by-detection approach based on appearance. To distinguish the object from the surrounding areas, a discriminative model using color histograms is applied. It adapts the object representation beforehand so that distractors are suppressed and the risk of drifting is reduced.

11) Scale Adaptive Mean Shift (ASMS) [13] This is a mean-shift tracker [14] optimizing the Hellinger distance between a template histogram and the target in the image. The optimization is done by a gradient descent. ASMS addresses the problem of scale adaptation and scale estimation. It also introduces two improvements over the original mean-shift [14] to make the scale estimation more robust in the presence of background clutter: 1) a histogram color weighting and 2) a forward-backward consistency check.

12) Normalized Cross-Correlation (NCC) [15] NCC follows the basic idea of tracking by searching for the best match between a static grayscale template and the image using normalized cross-correlation.

13) Multiple Instance Learning tracker (MIL) [16] MIL uses a tracking-by-detection approach with multiple instance learning instead of traditional supervised learning methods. It shows improved robustness to inaccuracies of the tracker and to incorrectly labeled training samples.

14) BOOSTING [17] It is based on MIL [16]. This is a real-time object tracker based on a novel on-line version of the AdaBoost algorithm. The classifier uses the surrounding background as negative examples in the update step to avoid the drifting problem.

15) MEDIANFLOW [18] This tracker uses optical flow to match points between frames. The tracking is performed forward and backward in time and the discrepancies between these two trajectories are measured. The proposed error enables reliable detection of tracking failures and selection of reliable trajectories in video sequences.

16) Tracking-learning-detection (TLD) [19] It combines both a tracker and a detector. The tracker follows the object from frame to frame using MEDIANFLOW [18]. The detector localizes the target using all appearances that have been observed so far and corrects the tracker if necessary. The learning estimates the detector errors and updates it to avoid these errors in the future.

17) Deformable Part-based Tracking by Coupled Global and Local Correlation Filters (DPCF) [20] This tracker that is derived from KCF relies on joint interactions between a global filter and local part filters. The local filters provide an initial estimate which is used by the global filter as a reference to determine the final result. The global filter provides a feedback to the part filters. In this way, it handles both partial occlusion (with the part filters) but also scale changes (with the global filter).

18) Contextual Object Tracker with Structural Encoding (CTSE) [21] This tracker uses contextual and structural information (that is specific to a target object) into the appearance model. This is first achieved by including features from complementary region having correlated motion with the target object. Second, a local structure that represents the spatial constraints between features within the target object is included. SIFT keypoints are used as features to encode this information.

19) Kalman filter ensemble-based tracker (KF-EBT) This tracker combines the result of two other trackers: ASMS [13] using a color histogram and KCF. Using a Kalman filter, the tracker works in cycles of prediction and correction. Firstly, a motion model predicts the target next position. Secondly, the trackers results are fused with the predicted position and the model is updated.
V. RESULTS AND ANALYSIS

In this section, we report our results on the 36 video sequences of [2]. The video sequences that consist in tracking persons, faces and objects include difficulties such as motion blur, scale change, out-of-plane rotation, fast motion, cluttered background, illumination variation, low resolution, occlusion, presence of distractors and articulated objects. Results are reported for the whole dataset.

A. Ranking method

During testing, we discovered that since different trackers have different tracking speed, using only the four metrics in section II-B is not enough. For example, some trackers have \(TPE\), \(TPO\) and \(BOR\) metrics that are good because they track slowly in real-time simulation, which means they just track a few frames correctly and all other frames are invalid and are ignored for the calculation of the metrics. Under this circumstance, the tracker succeeded in tracking every processed frame (which are only the first frames). Only \(TF\) can capture to some extent this lack of performance as it verifies if the target is in the FOV or not. Thus, we consider another essential metric: processed frame ratio (\(PR\)):

\[
PR = \frac{F_{NP}}{F_{TO}},
\]

where \(F_{NP}\) is the number of processed frames and \(F_{TO}\) is the total number of frames.

\(TF\) contains part of \(PR\) information since it stands for whether the object is inside the FOV in the processed frames. If \(PR\) is low, \(TF\) will be high since the tracker will not be able to track the object correctly in the processed frames because of the long interval between them. However, a high \(TF\) can be caused also by poor robustness.

After considering the PTZ camera nature and the tracker test results, we formulated a ranking formula. The formula stands for the Euclidean distance between the point defined by the pair \((BOR, TF)\) obtained by a tracker and the ideal tracker (top-right point in figure 1). The score is thus:

\[
\text{Score} = \sqrt{(1-BOR)^2 + TF^2}.
\]

We selected \(TF\) and \(BOR\) because we consider that \(TPE\) conveys similar information as \(BOR\) and \(TF\) conveys similar information as \(TPO\).

B. Results without processing delays

We have first set the execution ratio to zero in order to compare different trackers for their performances for in-plane rotations, out-of-plane rotations and drastic scene changes caused by the camera motion. We are looking for the most robust trackers, neglecting their processing times (they are set to all perform at the same speed). Note that in this experiment, the camera motion delays are considered as they reflect the robustness of the trackers. If a tracker performs poorly, this will cause unnecessary camera motion that will result in drop frames.

Table I and Figure 1(a) give the results for the 19 trackers based on their ranking. In the PTZ camera scenario, the difficulties with in-plane and out-of-plane rotations will be amplified because of the application dynamic nature. Surprisingly, trackers which adopted a scale adaptation function such as SRDCF do not necessarily perform better than other trackers due in great part to their slow execution speed. ASMS and DPCF are the best performers in this experiment. In the VOT 2016 benchmark [1] there is no drop frames caused by delays and less viewpoint changes caused by camera motion. Thus, it is reasonable that there is difference between the ranking of our framework and that of the VOT ranking. However, our ranking is still quite similar to that of VOT 2016 benchmark. Trackers like STAPLE, STAPLE+, KCF-EBT and DAT are good both in VOT benchmark and our benchmark. However, in our benchmark the performance of ASMS is surprisingly the best when it just ranked in the middle in VOT. And some trackers like SRDCF do not behave well in our PTZ framework probably because they do not output bounding boxes when the tracking fails and as a result, the PTZ camera is not controlled correctly. The VOT system will check every five frames to verify whether the tracker has failed. If it has failed, it is reinitialized. In our framework, we do not intervene in the tracking process at all. Early failures are thus more penalized.

C. Results with processing delays

We then set the execution ratio to 1 to track objects. In the real world the objects will keep moving while the tracker is processing a frame. As a result, the task of tracking is harder in this case since the intervals between processed frames are caused both by the time for processing the frame and the camera motion delay. \(PR\) should decline and the trackers...
should lose targets more easily. Table II and Figure 1(b) give the results for the 19 trackers.

Compared to the previous case, the ranking of trackers changes. ASMS still ranks first, but DPCF degrades because it is very slow and its $PR$ value declines to 0.08 where it was previously at 0.8. The other trackers relative rankings do not change much, but the average score of trackers is higher, which means that their performances are worse because of the execution delay in the tracking process. Still, compared to the VOT 2016 benchmark [1], the performance of ASMS is still surprisingly the best. This means that this tracker is particularly good for handling viewpoint changes. Finally, performance in VOT are better than for our task because we are testing online tracking and camera control. The PTZ scenario requires tracking people with different illumination variation, different scale and from rapidly changing viewpoints. All of those reasons make our results unique compared to other benchmarks.

### D. Target Position Prediction

Finally, we tested target position prediction to investigate if it can help trackers to perform better. Table III gives results for two trackers. Results are similar for all the other trackers. The proposed models (see section III) for predicting the next position of the target are not improving results. This is due to the fact that the speed of the target is difficult to estimate because it moves in 3D, but we estimate its motion in 2D. Therefore, the predicted speed is not very accurate. After calculating the speed, the framework will use this speed to predict the target position in the next frame. It may cause unnecessary large motion by the camera. For example, the predicted target motion may be too large so the camera, by rotating to this wrongly predicted position, will add delays in the tracking process. This adds to the possibility that the tracker will lose the target. If the target cannot be tracked, its speed will not be updated leading to an even worse situation where the camera just rotates more or less randomly. The high-speed trackers are affected the most by wrong prediction. Their process ratio declines from above 0.8 to below 0.2.

Therefore, we can conclude from this experiment that although appealing in theory, compensating slow tracking by a position prediction is not easy to apply in practice. It may work for objects that are far away and that mostly move in a plane, but it cannot work for target that are closer and that are moving toward or away from the camera. In such cases, the motion of the target cannot be predicted in 2D. For best results, it is thus preferable to design a fast tracker.

### VI. Conclusion

This paper presented a benchmark of recent trackers for the PTZ tracking scenario. Surprisingly, high-speed trackers, such as MEDIANFLOW and NCC, do not necessarily behave better than others. However, since predicting the target position was shown to be difficult, slow trackers should be avoided for the PTZ tracking task. The results of our test indicate that the top performing tracker for the PTZ scenario is the ASMS tracker. This tracker performed very well in accuracy as well as robustness in our tests. It is impossible to conclusively determine whether the performance of ASMS over other trackers come from its image features or its approach. Nevertheless, results of top trackers show that features play a significant role in the final performance.

### REFERENCES


Enhancing Human Action Recognition through Temporal Saliency

Vida Adeli
Department of Computer Engineering
Ferdowsi University of Mashhad
Mashhad, Iran
vida.adeli@mail.um.ac.ir

Ehsan Fazl-Ersi
Department of Computer Engineering
Ferdowsi University of Mashhad
Mashhad, Iran
fazlersi@um.ac.ir

Ahad Harati
Department of Computer Engineering
Ferdowsi University of Mashhad
Mashhad, Iran
aharati@um.ac.ir

Abstract—Images and videos have become ubiquitous in every aspects of life due to the growing digital recording devices. It has encouraged the development of algorithms that can analyze video content and perform human action recognition. This paper investigates the challenging problem of action recognition by outlining a new approach to represent a video sequence. A novel framework is developed to produce informative features for action labeling in a weakly-supervised learning (WSL) approach both during training and testing. Using appearance and motion information, the goal is to identify frame regions that are likely to contain actions. A three-stream convolutional neural network is adopted and improved by proposing a method based on extracting actionness regions. This results in less computation as it is processing only some parts of an RGB frame and also interpret less non-activity related regions, which can mislead the recognition system. We exploit UCF sports dataset as our evaluation benchmark, which is a dataset of realistic sports videos. We will show that our proposed approach could outperform other existing state-of-the art methods.

I. INTRODUCTION

Human action recognition is among the most fundamental topics in computer vision research community [1, 2]. The capability to automatically understand human actions in real world videos provides variety of applications ranging from intelligent surveillance and interactive systems to animation synthesis and content based retrieval. Despite decades of research, human action recognition remains a challenging problem in realistic videos.

More specifically, action recognition can be defined as the problem of determining what type of action is being performed by analyzing the content of an unknown video sequence and assigning it to a category among a set of predefined class labels. The complexity of motion patterns embedded in a video sequence makes it more complicated in comparison with recognition problems in still images. Spatial domain and visual cues capture the static appearance information about the scene. Whilst these attributes help to distinguish between certain action classes, they might result in confusion and error. For example, the motion pattern for “walking” and “running” classes is the only cue that can be used to distinguish the class labels, and without exploiting temporal extent of the action, they may be easily confused with one another. Therefore, we can say the visual cues of appearance and motion are two complementary elements and their fusion at the early stage of the algorithm can help to estimate high potential actionness regions.

Efficient modeling of actions is critical for recognizing human actions. In order to obtain reliable features, which are truly representative of actions, one may require to discard the background context and non-activity regions. However, in some cases background information can help to certainly prune a number of irrelevant classes from decision. For example, the observation of white snowy background can be effective in recognizing the class “Ice skating”. Most existing localization approaches inclusively remove the effects of background cues and try to propose bounding boxes of regions containing the principal action [3-5]. To this end, this paper seeks to extract high potential regions of video frames, which are more likely to entail a generic action instance by the means of motion patterns. At the same time, it proposes a representation that can utilize sufficient and informative amount of background context along with foreground information.

Additionally, it is important how the motion information is used in the recognition process. Inspired by the promising performance of Convolutional Neural Networks (CNN), the majority of existing frameworks make use of a two-stream CNN approach for spatial and temporal domain [1, 3, 6, 7]. They process motion cues separately from appearance information to produce a representation for video. Conversely, the key insight in this paper lies in how motion cues can be exploited to obtain main components of an action for fitting a better spatial representation into description methods such as CNN. Our contributions are two-fold: First, a novel method is proposed to estimate actionness regions of video frames incorporating motion information. This method moves beyond proposing a single bounding box for an action. Instead, it tries to obtain all action components. This results in a better recognition outcome, especially for actions that are composed of multiple human interaction or interaction of human and objects. Second, a representation is proposed that makes use of background context information in a general manner with less prominence compared to foreground. We will show that our approach could outperform many existing methods.
The rest of this paper is organized as follows. In section II a review of the related work on action recognition is provided. Section III describes the proposed action recognition model. Section IV introduces the dataset and discusses the experiment results; and finally in section V the paper is concluded by summarizing the proposed method and obtained results, and outlining potential directions for future work.

II. RELATED WORK

There is a considerable amount of literature that aims at human action recognition. In this section, we try to provide a general overview of existing methods and position our work with respect to the current state of the literature.

Action recognition approaches can be typically divided into two categories: handcrafted feature representations, which mostly make use of local features, and deep neural network models.

A. Handcrafted feature models

There are two main points of view in using handcrafted features for action recognition. One may use a holistic representation, which is a global representation of human body or movements, while some others make use of local features. Space-Time Volume proposed in [8] falls into the holistic category, where a 3D space-time volume is created by stacking silhouette of a person along temporal dimension. Some holistic methods represent each activity with a two-dimensional template image. Motion Energy Image (MEI) and Motion History Image (MHI) [9] are examples of these template images that are created from a sequence of foreground patterns and indicate history of motion occurrences. However, it is obvious that holistic approaches learn a general schematic estimation of an action and cannot capture finer details of it. This shortcoming pushed the investigation towards local models.

Many studies have addressed human action recognition problem by extracting space-time local features [10-15]. The primary work of [15] on Spatio-Temporal Interest Points (STIPs) detects regions of an image that have significant changes in both appearance and motion. Histogram of Oriented Gradients (HOG) and Histogram of Optical Flow (HOF) are exploited in [11] to describe spatio-temporal features. It then applies a bag of features model (BoF) to encode video descriptors. Following by this approach as the representation stage, a SVM classifier is used to make action predictions. In later works, cube local video features are proposed as the 3D version of existing methods using temporal information as their third dimension. Examples of these methods are HOG3D [14], Harris3D [16], Cuboids [17].

Despite nearly acceptable achievement of handcrafted features, there exist many shortcomings, like being too rigid in capturing possible variations of actions. This pushed the research towards deep models.

B. Deep Models

Recently, Convolutional Neural Networks have achieved significant success in various fields such as image classification. Inspired by this remarkable progress, researchers have attempted to utilize these networks into video analysis applications such as action recognition. There have been considerable efforts to incorporate temporal information of videos into CNNs [6, 18-20]. We review these approaches in two categories: Two-stream and spatio-temporal networks.

A prominent class of deep neural networks is the two-stream approach, introduced in [6]. The structure of this network is composed of two parallel channels, one for spatial components to explicitly capture information about appearance, scene and individuals, and the other for temporal components, indicating motion between consecutive frames. Eventually, the final prediction is obtained by the fusion of these two streams’ output. The majority of works in action recognition adopt a two-stream technique, in order to engage temporal information into the process [1, 3, 4, 7, 21].

Another approach in using CNNs for action recognition is to train a single network, by providing structures that use temporal information. These networks are called spatio-temporal networks. 3D convolutional networks are examples of these methods, introduced by Ji et al. [18]. Obviously, these networks accept predefined numbers of frames. They can only process short video clips and can encode local, short-term motion patterns. Moreover, it is unclear if they can capture adequate amount of motion information from videos. To exploit temporal cues for classifying video sequences, some studies use another example of spatio-temporal networks such as recurrent neural networks (RNNs) [22] or their extension, called long short term memory (LSTM) [23].

In comparison with the two-stream method, both of previously mentioned approaches involve complicated processing and share a large number of parameters that should be tuned. This entails the need for a very large training data that may be costly to produce. Therefore, it is essential to build up frameworks that can learn temporal information of videos without the cost of enormous training data. As these networks do not discriminate between foreground and background information, we try to feed these networks with informative regions of video frames using motion cues.

III. OVERVIEW OF PROPOSED METHOD

This paper moves beyond just recognizing video frames and tries to propose a method that can estimate regions of interest in each frame using motion and appearance information. A method is proposed that disposes the need for manual user provided annotations by estimating per frame ROI patches, containing eminent components of an action. It is obvious that these informative proposals as the input of a CNN, provides superior representation of an action and improves classification accuracy.

Our approach consists of four main steps: The first step utilizes both appearance and motion to estimate actionness potentials in which a probability value is assigned to each pixel, indicating their confidence of being foreground. In the second step, high potential ROI patches are selected from each frame and the second representation is constructed in order to incorporate adequate amount of background information. Then CNN features are extracted using the three streams of information, two for appearance models and the third for optical
flow. Subsequently, in the last step, class labels are assigned to each video.

A. Actionness map estimation

To estimate a measurement for each pixel, indicating the probability of belonging to the foreground, we follow a temporal saliency method proposed by Papazoglou et al. [24] for video object segmentation. First, the algorithm computes optical flow between two consecutive frames and based on motion information of each pixel and its neighbors, motion boundaries are estimated. Then it determines for each pixel, whether or not they are within that boundary. Motion boundaries are those pixels of frames that have abrupt changes in their optical flow field.

Two conditions are contemplated to capture boundaries. First, the gradient magnitude of optical flow in a pixel can indicate the probability of being motion boundary. If the optical flow is larger than a threshold, then the pixel is likely to be a motion boundary. Due to the inaccuracy of computing optical flow, a second factor is considered that removes the effect of moving camera motions. It uses the gradient of frame, inspecting the difference between moving direction of a pixel and its neighbors. If a pixel moves differently from all its neighbors, then the pixel is likely to be a motion boundary. Considering $b^m_p$ and $b^0_p$, respectively, as the first and second factor for each pixel $p$, the final constraint ($b_p$) is the combination of these two terms:

$$b_p = \begin{cases} b^m_p & \text{if } b^m_p > T \\ b^m_p \cdot b^0_p & \text{if } b^m_p \leq T \end{cases}$$

Where $T$ can be estimated as the threshold in which the value of $b^m_p$ is considered acceptable. Finally, applying a threshold with value of 0.5 on $b_p$, provides a binary motion history image. However, there is still a need for appearance information to compute foreground probability of each pixel, on account of the fact that some parts of the person's body may be immovable and motionless in some frames of the video. Consequently, an appearance model is considered by fitting a GMM over RGB values and exploring whether a pixel is inside the produced motion boundaries using point-in-polygon (PIP) problem. This generates an inside-outside map for each frames of the video. Eventually, combing these two models, using the summation of their logarithmic probabilities, presents the likelihood of each pixel belonging to the foreground.

B. Selecting actionness patches

The main idea behind the proposed method is to use motion information embedded in video with the purpose of providing informative inputs to the network both during train and test stages. The algorithm mainly explores those regions where actions are happening. The goal is to decide which windows in frames are appropriate for better describing the video. Using the foreground probability map described in previous section, the algorithm tries to select high potential areas of frames resembling the problem of sampling over spatial dimension.

For this purpose, a methodology similar to hysteresis thresholding is proposed. Two different thresholds are considered and are applied to the foreground probabilities. This leads to creation of three classes of pixels: weak, strong and in-between pixels. Pixels below the lower threshold are weak pixels, which can be safely discarded due to the low probability of being a part of an action. Pixels above high threshold are strong pixels that should be retained and added to the set of pixels which have high potential of containing an action. Finally, for in-between pixels, which are pixels between low and high threshold, they are classified as strong if they are connected to a strong pixel directly, or by less stringent indirectly, by the means of other previously connected in-between pixels. They can be considered as continuations of strong areas and can be added to the set of selected pixels.

Correspondingly, these three types of pixels form three classes of weak, strong and in-between regions in frame. In-between regions are those parts of human body or object that may not have an abrupt motion but have relatively good potential of being foreground due to their almost high probability and also closeness to a strong region. Using two thresholds benefits from two principal advantages: first, it insures selecting only high potential pixels. Those pixels that do not have a significant value and are not nearly a strong region are assumed as noisy pixels and are discarded. Second, not only strong regions are selected but also those in-between regions that are probable to be a part of an action are chosen. This provides a better generalization of algorithm to find high potential regions. Fig. 1 shows the improvement clearly where at first small parts of human body are selected and then the detection is improved using the second threshold. In Fig. 2 the detected high potential pixels for three different actions are illustrated.

After specifying high probable pixels, the algorithm detects connected regions and calculates the average of foreground probabilities within each region. Then the maximum value of the mean probabilities is selected and a fixed size window is considered over that pixels to produce a candidate ROI for that frame. For the last step, all pixels inside the chosen ROI are discarded from further processing at this stage. This procedure is repeated until a specific number of ROIs is acquired. The point to notice is that those ROI patches that have more than seventy percentage overlap with previously selected ROIs are eliminated. Fig. 3 presents extracted patches for three different actions. It can be seen that the proposed algorithm could not only detect the person performing action but also it could identify all components of actions including objects and interaction of other individuals.

C. Incorporating background information

The method proposed in the previous section, is capable of identifying important components of an action. As stated in the introduction, in many cases, background can provide rather useful information about the activity. There are two ordinary trends in exploiting background information. Some methods try to learn the original frames. These methods do not discriminate between ROI and background information. In contrast, in the second category of methods the ROI is typically extracted by leveraging background subtraction, tracking or pose estimation.
approaches, where the background is completely ignored. However completely discarding background is not recommended, while it can be helpful of recognizing actions.

Therefore, this paper suggests another representation that can take advantage of background information in decision-making to complement the previously described patch-based representation. In this representation, those pixels with a probability \( P_r \) higher than a specific value are extracted and then the smallest rectangular window \( W_f \) bounding them is generated. In the next step, a new representation is built where pixels inside the window remain intact in the new representation and the remaining pixels become blurred using Gaussian filters with standard deviation parameters \( \sigma \) proportional to the inverse of \( P_r \) for each pixel \( P_i \) and the distance (Dist) to the selected window \( W_f \):

\[
\sigma \sim \text{Dist}(P_i, W_f) \sim \frac{1}{P_i}
\]  

(4)

This results in a representation in which background information has less impact and power to respond to the CNN filters since the strong structures in the background have been suppressed. Fig. 4 illustrates the proposed weighted representation.

**D. Descriptor computation and classification step**

This paper utilizes a convolutional neural network to perform the task of automatic feature learning. Following the two-stream convolutional networks used in [3], we make use of a same architecture but using it as a three-stream network, one temporal stream for the optical flow fields and two spatial streams for the two proposed appearance representations. Eventually, a nonlinear SVM classifier is used as the last step of proposed framework for the final classification.

**IV. EXPERIMENTS AND RESULTS**

We conduct experiments on the real-world UCF sports dataset to demonstrate the effectiveness of our action recognition model in real scenarios and compare our approach against a number of state-of-the-art methods. Details about the dataset and the experiments performed are given below.

**A. UCF Sports Dataset**

UCF sports dataset contains 150 videos categorized into 10 action classes. Videos are collected from real sports broadcasts. The recognition task in this dataset is challenging because of a wide range of scenes, viewpoints and camera movements. We use the standard training and test split that is used by many researches and suggested in [25].

**B. Implementation details**

We implement our deep learning tasks based on the Caffe open source toolbox. The optimal value of the two thresholds used in section III for selecting actionness patches, have been specified through experiments on a validation set. For the lower threshold the value of 0.5 and for the higher threshold the value of 0.7 is selected.

To capture CNN features, a three-stream CNN model is employed. We make use of AlexNet architecture pre-trained on UCF Sports dataset, for the two special streams. The inputs of one of this spatial networks are our ROI patches and the other are the proposed Blurred representation. To capture motion patterns, our temporal stream network bears a close
The features are then extracted from fc7 layer [3] of three CNNs, representing action specific regions and motion cues. Afterwards, a temporal statistical pooling is applied in order to obtain fixed length feature vectors per video. Precisely explaining, video features are constructed by aggregating all frame descriptors of each motion and Blurred representation, and all patch descriptors separately using average, max, median and variance aggregations. This results in three descriptors with \( n=4\times4096 \) dimensions.

Eventually, the three learnt spatial and temporal features are fused to produce a final classifier for action recognition. We use an RBF kernel function to predict action category labels of each video. Features of three representations are fused in the kernel space, using their similarity measures.

C. Experimental results

We evaluate our method on UCF Sports Action dataset. We choose the mean per-class recognition accuracy as our evaluation criteria. In order to comprehensively evaluate the performance of our method, we try to compare it with state-of-the-arts baselines. TABLE 1 summarizes the result of action recognition accuracies in different works across this dataset. From the results, it is clear that our proposed method improves the accuracy of action recognition in UCF dataset and achieves excellent performance in comparison with other competing state-of-the-art results. The confusion matrix for UCF sports dataset of our method is shown in Fig. 5.

V. EXPERIMENTAL DISCUSSION

The experimental results highlighted the impact of motion cues in order to estimate regions of action interests in videos. We attribute the improvement made by the proposed method to three main reasons. Firstly, it uses the motion embedded in the video to estimate high potential areas of actions, under the strong assumption that appearance cues are not the only information that can be used to distinguish an action. Indisputably, motion information can also provide constructive evidences in determining where an action may have occurred. Using motion in early stage of the algorithm, contributes in a better estimation of actionness regions. Secondly, the proposed method moves beyond finding a single bounding box for each action and tries to find almost all probable components of an action. This exert less constraint on recognizing different types of actions, especially those actions that consist of multiple human interactions or the involvement of some object in the action. Finally, we do not suppress the effect of background information in decision making. We propose a framework that can exploit the gist of background cues holistically, while still giving foreground information significantly more weight.

<table>
<thead>
<tr>
<th>TABLE 1. COMPARISON OF AVERAGE RECOGNITION ACCURACY WITH STATE-OF-THE-ART METHODS ON UCF SPORTS DATASET.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recognition accuracy (%)</td>
</tr>
<tr>
<td>---------------------</td>
</tr>
<tr>
<td>UCF-Sports</td>
</tr>
</tbody>
</table>
representation of a video, we suggested an approach that utilizes motion information. We proposed two different representations. In the first one, high potential patches of video containing components of an action are extracted. In the second representation, we tried to include background information. This provides a gist of background structure as the input of the recognition system. These two representations are passed through a three-stream CNN, where the third stream is the optical flow fields. From the experimental results, we illustrate that our approach outperforms the state-of-the-art methods for action recognition on the UCF Sports dataset. We also presented an analysis of our experimental results.

As a potential direction for future work, we intend to investigate a novel mechanism to efficiently aggregate sets of frame level features into discriminative and fixed-size video descriptors.

REFERENCES


Oral Session G

Classification

Tuesday May 15, 2018, 2:50 PM

William Brendel and Luis Marujo
  Support Vector Machine As Graph Theory Problems

Rafael Menelau Oliviera E Cruz, Robert Sabourin and George D. C. Cavalcanti
  On dynamic ensemble selection and data preprocessing for multi-class imbalance learning

Yuanlie He, Sudhir Mudur and Charalambos Poullis
  Multi-label Pixelwise Classification for Reconstruction of Large-scale Urban Areas

Luca Mossina and Emmanuel Rachelson
  Naive Bayes Classification for Subset Selection in a Multi-label Setting

Xiao Bai, Yuan Tian and Jun Zhou
  Semi-supervised product quantization for approximate nearest neighbor search
Abstract— Quadratic programming (QP) problem reformulation has been a research problem for nearly two decades, but is seldom linked to Graph Theory. In fact, typical reformulations convexify a non-convex QP problem. This is accomplished by making the objective function differentiable, optimizing in the continuous domain while ensuring the final solution is binary, or adding regularizers and Lagrangian coefficients to optimize the dual problem. In this research, we demonstrate that QP problems can also be reformulated using the same mechanism as P/NP problem reduction, overcoming speed and memory footprint limitations from other type of reformulation. We use SVM to make the demonstration. In the demonstration, we show that SVM is comparable to a soft weighted edge maximum independent set problem where the amount of support vectors per class is balanced. As a result, SVM can also be reformulated as a maximum clique problem with the same class balancing constraint. After transforming the sequential minimal optimization (SMO) algorithm to our new maximum clique formulation, we demonstrate that such reformulation leads to improved training performance, reaching 36 times faster training time, and a sparser solution for less than one percent accuracy degradation in some datasets.

Index Terms— Quadratic Optimization, Classification algorithms, Support Vector Machines, Graph Theory, Independent Set, Maximum Weighted Clique.

I. INTRODUCTION

In the fields of Machine Learning, Natural Language Processing, Computer Graphics, and Computer Vision, there are multiple problems that can be solved using a quadratic programming (QP) solution. They are usually are formulated as graph theory problems, where typical objective functions to be optimized contain unary potentials related to nodes and binary potentials related to edges. There is a long 50-years history where graph theory has been at the core of an ocean of computer science applications [1], [2]. For instance, image segmentation has been formulated as minimum cut [3], [4], maximum weight independent set [5], maximum weight clique [6], [7] and minimum spanning tree [8] problems [9]. Multi-object tracking has been modeled as maximum weight independent set [10] and generalized minimum and maximum clique [11], [12] problems. Nevertheless, not all QP problems are directly related to graph theory as demonstrated by [13]–[17]. Since graph problems can be reduced to one another, QP problems can also be reformulated into problems that can be solved more efficiently. Several cut problems are reformulated as spectral clustering problems that can be optimized via weighted kernel k-means algorithms [4], [18], achieving real-time computation performances. In the work of Tsang et al. [19], SVM and Support Vector Clustering (SVC) are formulated as minimum enclosing ball problems, obtaining approximate optimal solutions in linear time, with a space complexity independent of the problem size. This work focuses on how SVM formulation [20] can be viewed as an independent set problem [19], [21], and thus be reduced to other graph problems.

More precisely, we connect the SVM dual QP formulation Eqn. (1) to a maximum independent set formulation, which gives a new interpretation on the support vector selection process. Then, the same way we reduce an independent set problem to a maximum clique problem by taking the complement of a graph, we reformulate the SVM dual QP formulation as a dominant set QP [6], [7], the latter being used to define the maximum clique (MC) problem [21], [22]. We show that our MC formulation involves Mercer distance kernels, instead of Mercer similarity kernels, and demonstrate how to construct such kernels, leading to new families of kernels. Finally, we show that our MC formulation has computational advantages while preserving comparable accuracy compared to the standard LIBSVM implementation of the SVM dual QP formulation.

The remainder of this paper is organized as follows: Section II gives a review of the SVM Dual Formulation and SMO, Section III introduces the Problem and how we reformulated it. Section IV includes the experiments and result, Section V presents the conclusions and future work.

II. SVM DUAL FORMULATION AND SMO

SVM aims at learning boundaries between feature vectors \( \{x_k\}_{k=1}^{\gamma} \) of different classes \( \{y_k\}_{k=1}^{\gamma} \). When classes are binary, SVM dual formulation has the following form [20], [23]:

\[
\begin{align*}
\alpha^* & \leftarrow \arg \max_{\alpha \in [0, C]^n} F(\alpha) : 1^T \alpha - \frac{1}{2} \alpha^T H \alpha \\
\text{s.t.} & \quad y^T \alpha = 0 \quad \text{with} \quad y \in \{-1, 1\}^n \\
& \quad H_{ij} = y_i y_j K(x_i, x_j) \quad \text{and} \quad x_k \in \mathbb{R}^d
\end{align*}
\] (1)

where \( \alpha_k > 0 \) means that \( x_k \) is a support vector for class \( y_k \), i.e. the boundary is defined as a linear combination of \( \{x_k | \alpha_k > 0\} \) in a vector space where the dot product is induced by the kernel function \( K \). SMO was originally designed\(^1\) MC = Maximum Clique, not to be confused with Monte Carlo.
to train a support vector machine that requires the solution of the very large SVM dual QP optimization problem [24], defined above. SMO belongs to the family of SQP algorithms [25]. It breaks a large QP problem into a series of smaller QP problems, each of which optimizes a quadratic model of the objective subject to a linearization of the constraints. The method is equivalent to applying Newton’s method to the Karush-Kuhn-Tucker (KKT) conditions of the QP problem. In SMO, the small QP problems involve only two variables. These small QP problems are solved analytically, thus avoiding the use of a time-consuming numerical QP optimization at each iteration. The amount of memory required for SMO is linear in the training set size $n$, which allows SMO to handle very large input sets.

III. PROBLEM SETUP AND REFORMULATION

Typical mathematical formulations of graph theory problems involve optimizing objective functions that contain unary potentials related to nodes, and binary potentials related to edges. Unary potentials are embedded in the form of a vector, and binary potentials in the form of an adjacency matrix. The latter is symmetric if the graph is undirected and semi-definite if the graph theory problem to be solved is P-complete. In the following, we outline how SVM can be interpreted as a special maximum independent set (MIS), and how it can be reformulated as a balanced maximum clique (MC). The two steps reformulation (SVM $\rightarrow$ MIS $\rightarrow$ MC) offers several advantages: it provides another interpretation on the SVM maximum margin formulation, which helps in designing new algorithms, and it also gives us the opportunity to take advantage of 50 years of research and algorithm development of about the MC problem, like the replicator dynamics approach from [26]. The MC formulation is particularly interesting as it provides a natural control over the sparsity of the solution in terms of number of support vectors.

A. SVM As a Balanced Maximum Independent Set Problem

In graph theory, an independent set (IS) is a set of vertices in a graph $G(V, E)$, no two of which are adjacent. That is, it is a set $S \subseteq V$ of vertices such that for every two vertices $i, j$ in $S$, there is no edge connecting the two: $\forall i, j \in S, (i, j) \notin E$. Equivalently, each edge in the graph has at most, one endpoint in $S$. The size $|S|$ of an independent set is the number of vertices it contains. A maximum independent set (MIS) is an independent set of largest possible size for a given graph $G$. When each vertex is assigned a weight, the maximum weighted independent set (MWIS) is the subset of vertices whose weights sum to the maximum possible value without any two vertices being adjacent to one another. Both MIS and MWIS problem are NP-hard [27]. We illustrate the concept of IS and MIS in Fig. (1). The MIS problem is mathematically formulated as follows. Let’s assume that a graph $G$ contains $n$ vertices. We define $\alpha \in \{0, 1\}^n$ as the support of $S$, i.e. $\forall i \in V, i \in S \Leftrightarrow \alpha_i = 1$, otherwise $\alpha_i = 0$. We represent the edge set $E$ by the adjacency matrix $A$, i.e. $(i, j) \in E \Leftrightarrow A_{ij} = 1$. [21] demonstrates that $\frac{1}{2} \alpha^T A \alpha$ is the number of edges between vertices in $S$ and that $1^T \alpha = |S|$. Hence, $S$ is an independent set if and only if $\alpha^T A \alpha = 0$, and the maximum independent set problem can be formulated as:

$$\alpha^* \leftarrow \arg \max_{\alpha \in \{0, 1\}^n} 1^T \alpha \quad \text{s.t.} \quad \alpha^T A \alpha = 0 \tag{2}$$

In addition, [21] shows that the latter equation can be reformulated as:

$$\alpha^* \leftarrow \arg \max_{\alpha \in \{0, 1\}^n} 1^T \alpha - \frac{1}{2} \alpha^T A \alpha \tag{3}$$

and that we can loosen the binary constraint $\alpha \in \{0, 1\}^n$ to the continuous domain $\alpha \in [0, 1]^n$. We can further add a regularization diagonal matrix $D$: $A \leftarrow A + D$, such that the final solution entries $\alpha^+_n$ are pushed to the domain boundaries [22], leading to the final MIS problem formulation:

$$\alpha^* \leftarrow \arg \max_{\alpha \in [0, 1]^n} 1^T \alpha - \frac{1}{2} \alpha^T (A - D) \alpha \tag{4}$$

We can clearly see that Eqn. (1) resembles the independent set QP formulation Eqn. (3), with three differences. First, $\alpha \in [0, 1]^n$ instead of $\alpha \in \{0, 1\}^n$ in Eqn. (1), second $H \in \mathbb{R}^{n \times n}$ whereas $A \in \{0, 1\}^{n \times n}$, and third Eqn. (1) has an additional constraint $y^T \alpha = 0$. Nevertheless, we can reformulate both Eqn. (1) and Eqn. (3) to establish clearer correspondences.

In Eqn. (1), we remap $\alpha \leftarrow \frac{1}{C} \alpha$ so that $\alpha \in [0, 1]^n$. This is equivalent to multiplying the kernel $K$ by $C$. Without a loss of generality, we can further normalize $K$ so that $H \in [-1, 1]^{n \times n}$. Note that the new normalized kernel is still a valid SVM Mercer kernel, since the normalization is linear, hence it preserves the convexity of $H$.

In Eqn. (3), we relax $A_{ij} \in [0, 1]$ to represent an edge probability rather than an hard binary edge connectivity. Then we remap $A \leftarrow 2A - 1^T 1$ so that $A_{ij} \in [-1, 1]$. This can be interpreted as mapping the edge probabilities to a correlation metric. Moreover, this does not affect the solution of the MIS formulation as it just adds an additional maximizer $(1^T x)^2$ to the objective function in Eqn. (2) and (3). Indeed, the function $f: z \rightarrow z + z^2$ is strictly monotonically increasing on $\mathbb{R}_+$, hence maximizing $f(1^T x)$ is the same as maximizing $1^T x$. In the next section we also see that this additional maximizer
\[(1^\top x)^2\] vanishes in the MC formulation Eqn. (5) due to the MC sparsity constraint.

As illustrated Fig. (2), we see now that support vectors can be viewed as nodes in \(G\), connected to each other with a soft correlation weight \(H_{ij}\) on edges (instead of an edge probability), where \(H_{ij} > 0\) means that support vectors \(i\) and \(j\) are likely connected on the graph, and \(H_{ij} < 0\) means that they are likely disconnected on the graph. \(H_{ij} = 0\) would be equivalent to having an edge probability of 0.5.

\[\begin{align*}
\text{(a) 2-classes dataset} & \quad \text{(b) Graph from } H \\
\text{(c) The MIS}
\end{align*}\]

Fig. 2. Independent set formulation of SVM. Edges with high \(H_{ij}\) weight are plain, edges with low \(H_{ij}\) weight are dashed. Green nodes represent the support vectors as a MIS. Best viewed in color.

Finally, the additional constraint \(y^\top \alpha = 0\) imposes the solution to be balanced: the amount of support vectors for each class should be the same.

**B. SVM Interpretation**

Relating SVM to a maximum independent set problem induces the following interpretation: SVM selects support vectors (i) that are either not from the same class and highly correlated, or that are from the same class and as dissimilar as possible, (ii) such that the amount of support vectors for each class is balanced. It means that SVM aims to maximize support vectors with small margin (as the original max-margin SVM formulation [28]) between classes that summarize locally the decision boundary intra classes. The next section further emphasizes even more the notion of boundary summarization with the additional sparsity constraint on the solution when Eqn. (1) is reformulated as a maximum weighted clique problem.

Tsang et al. [19] formulates SVM and SVC as approximations to the minimum enclosing ball problem, allowing them to train support vectors on very large datasets. Their algorithm iteratively alternates between refining the center of the enclosing ball and finding the next support vector as far away from it as possible. The center is defined implicitly: only the similarity from a point to the center is defined explicitly as the linear combination of the support vector kernels. Our interpretation fully supports their approach as we showed that support vectors are far away from each other intra-class and as close as possible to each other extra-class, which is the essence of the max-margin formulation. It also means that we can extend the strategy of [19] to initialize the solution by greedily preselecting a group of pairs of data points as initial support vectors such that points in each pair are from different classes, and such that the distance intra-class is maximized and the distance extra-class is minimized. This strategy helps reduce the number of iterations and thus the training time as demonstrated by Table II. Next we show that if SVM is viewed as an MIS problem, it can also be reduced to a MC problem.

**C. SVM As a Balanced Maximum Clique Problem**

Given a graph \(\tilde{G}(\mathcal{V}, \mathcal{E})\) with \(n\) vertices, a clique is a subset of vertices \(S \subset \mathcal{V}\), all connected to each other, i.e. \(\forall i, j \in S, (i, j) \in \mathcal{E}\). The size \(|S|\) of a clique is the number of vertices it contains. A maximum clique (MC) is a clique of largest possible size for a given graph \(\tilde{G}\). When each vertex is assigned a weight, the maximum weighted clique (MWC) is the subset of vertices all adjacent to each other and whose weights sum to the maximum possible value. Both MC and MWC problem are NP-hard [27], as well as enumerating all possible clique in a graph. We illustrate the concept of clique and MC in Fig. (3).

\[\begin{align*}
\text{(a) Graph} & \quad \text{(b) A clique} \\
\text{(c) The MC}
\end{align*}\]

Fig. 3. A 5-vertices graph, with a clique of size 2, and the maximum clique of size 3. Vertices in the (maximum) clique are displayed in green. Best viewed in color.

In [29], Motzkin and Straus showed that the maximum clique problem is equivalent to the following continuous quadratic program:

\[
\alpha^* \leftarrow \arg \max_{\alpha \in \mathbb{R}^n_+} \frac{1}{2} \alpha^\top \tilde{A} \alpha \quad \text{s.t. } 1^\top \alpha = 1 \tag{5}
\]

where \(\tilde{A}\) is the adjacency matrix for the graph \(\tilde{G}\), and \(\alpha \in \mathbb{R}^n_+\) is the support of \(S\), i.e. \(\forall i \in \mathcal{V}, i \in S \Leftrightarrow \alpha_i > 0\). More precisely, there exists a solution \(\alpha^*\) to Eqn. (5) such that (i) every nonzero component of \(\alpha^*\) is equal to \(\frac{1}{k}\), where \(k\) is the maximum cardinality of a clique in \(\tilde{G}\); and (ii) the set \(S = \text{supp}(\alpha^*)\) is a clique of size \(k\). Typically \(\tilde{A}_{ij} \in \{0, 1\}\), but having \(\tilde{A}_{ij} \in \{-1, 1\}\) does not change the formulation, as the transformation \(A \leftarrow 2\tilde{A} - 1^\top I\) introduces an additional penalty term \(- (1^\top \alpha)^2\) that vanishes with the constraint \(1^\top \alpha = 1\).

The MC and MIS problems are highly correlated: the solutions of the MIS problem on a graph \(\tilde{G}\) are the same as the solutions of the MC on the complement \(\bar{G}\) of the graph \(\tilde{G}\), as illustrated Fig. (4). Similarly, we go from Eqn. (3) to Eqn. (5) by taking the complement \(\bar{A}\) of the adjacency matrix \(A\), i.e. \(\bar{A} \leftarrow 11^\top - A\). Indeed, the objective function of Eqn. (3) becomes \(1^\top \alpha - \frac{1}{2} \alpha^\top \bar{A} \alpha = 1^\top \alpha (1 - \frac{1}{2} 1^\top \alpha) + \frac{1}{2} \alpha^\top \bar{A} \alpha\). Since a maximizer \(\alpha^*\) of \(\alpha^\top \bar{A} \alpha\) is up to a constant factor, and since the function \(f: z \to z(1 - \frac{1}{2} z)\) has a strict maximum at \(z^* = 1\), we can eliminate \(f(1^\top \alpha) = 1^\top \alpha (1 - \frac{1}{2} 1^\top \alpha)\) from the objective function, add the constraint \(1^\top \alpha = 1\) and relax \(\alpha\) to the continuous domain, resulting the formulation Eqn. (5).
The dual SVM formulation is similar to the MIS formulation [21], with the additional constraint \( y^T \alpha = 0 \) and where the graph edge weights have soft values (i.e. \( H_{ij} \) in Eqn.(1)). Since we can switch from the MIS problem to MC problem by taking the complement of the graph, we can then reformulate the SVM dual formulation in Eqn. (1) as:

\[
\begin{align*}
\alpha^* &\leftarrow \underset{\alpha \in \mathbb{R}^n_*}{\text{arg max}} \quad \mathcal{F}(\alpha) = \frac{1}{2} \alpha^T \mathbf{H} \alpha \\
\text{s.t.} \quad &y^T \alpha = 0 \quad \text{and} \quad 1^T \alpha = \nu \\
&\text{with} \quad H_{ij} = y_i y_j K(x^i, x^j) \\
&\text{and} \quad x^k \in \mathbb{R}^d, \quad y \in \{-1, 1\}^n
\end{align*}
\]

where \( \tilde{K}(x^i, x^j) \) is now a Mercer kernel representing the distance between the two vectors \( x^i \) and \( x^j \) instead of their similarity. We illustrate our new formulation in Fig. (5).

---

D. Mercer Distance Kernels

First of all, we can see that if \( K(x^i, x^j) : \mathbb{R}^n \rightarrow \mathbb{R} \) is a similarity measure, \( \exists \rho, \nu \in \mathbb{R} \) such that we can build a distance metric \( \tilde{K}(x^i, x^j) : \mathbb{R}^n \rightarrow \mathbb{R} \) from a similarity measure using one of the following transformations:

\[
\tilde{K}(x^i, x^j) = \frac{\nu}{v + K(x^i, x^j)} \quad \text{or} \quad \tilde{K}(x^i, x^j) = v - K(x^i, x^j) \quad (7)
\]

Note that other transformation functions like \( x \rightarrow e^{-vx^2} \) can be used as well. Proof and new kernel construction can be found in [30]. Interestingly enough, we can build a similarity measure from a distance metric the same way.

**Lemma 1.** \( \exists \rho, \nu \in \mathbb{R} \) such that Eqn. (7) always produces a valid Mercer kernel.

**Proof sketch.** We base the proof on the Gershgorin circle theorem. We can find \( \nu \) and \( \rho \) such that:

\[
\min \text{eigenvalue} \geq \min_i \left( \tilde{K}(x^i, x^i) - \sum_{j \neq i} K(x^j, x^j) \right) \geq 0 \quad (8)
\]

Since the Gershgorin bound is very loose, solving Eqn. (8) will produce a final matrix where \( [\tilde{K}(x^i, x^j)] \gg [K(x^i, x^j)] \). Additional knowledge on the domain of \( \alpha \) helps to refine the bound. Since \( \alpha \in \mathbb{R}^n_* \), we can see that \( \rho > 0 \) and \( \nu = \max_{i,j} [K(x^i, x^j)] \Rightarrow \tilde{K}(x^i, x^j) \geq 0 \Rightarrow \forall \alpha \in \mathbb{R}^n, \sum_{i,j} \alpha_i \alpha_j \tilde{K}(x^i, x^j) \geq 0. \)

One may want to normalize the newly constructed distance kernel \( \tilde{K} \), or construct a Mercer distance kernel directly from a distance metric. Let \( d(x^i, x^j) \) represent an arbitrary distance metric between the two vectors \( x^i \) and \( x^j \), and let’s define:

\[
\tilde{K}(x^i, x^j) = \frac{\rho + \nu d(x^i, x^j)}{\nu + d(x^i, x^j)} \quad (9)
\]

**Lemma 2.** \( \exists \nu, \nu_2, \rho \in \mathbb{R} \) such that Eqn. (9) produces a valid Mercer kernel and \( K(x^i, x^j) \in [0, 1] \).

**Proof sketch.** We will parametrize \( \rho = \nu_2 \nu_2 - \rho_2 \). Then Eqn. (9) becomes:

\[
\begin{align*}
\tilde{K}(x^i, x^j) &= \frac{\rho + \nu_2 d(x^i, x^j)}{\nu + d(x^i, x^j)} = \frac{\nu_2 \nu - \rho_2 + \nu_2 d(x^i, x^j)}{\nu + d(x^i, x^j)} \\
&= \frac{v_2 (\nu + d(x^i, x^j)) - \rho_2}{\nu + d(x^i, x^j)} = \nu_2 - \frac{\rho_2}{\nu + d(x^i, x^j)} \quad (10)
\end{align*}
\]

We recognize the two transformations defined in Eqn. (7). We find \( \nu \) and \( \rho_2 \) such that \( \frac{\rho_2}{\nu + d(x^i, x^j)} \) is a Mercer similarity kernel, then we find \( \nu_2 \) such that \( \tilde{K}(x^i, x^j) \) is a Mercer distance kernel and we scale \( \rho_2 \) and \( \nu_2 \) such that \( \tilde{K}(x^i, x^j) \in [0, 1] \). \( \square \)

Note that in many cases a simpler version of Eqn. (9) can be used by setting \( \rho = 0 \) and \( \nu_2 = 1 \), allowing a grid search only with respect to \( \nu \) with the kernel \( \tilde{K}(x^i, x^j) = \frac{d(x^i, x^j)}{\nu + d(x^i, x^j)} \).

### IV. Experiments and Results

We evaluated our MWC formulation using the following datasets: “adult” (adult.), “webpage” (web.), “cod-rna” (cod.), and “splice” (spl.) from [31]. The adult dataset is composed of nine partitions: a1a \( \rightarrow \) a9a and the webpage dataset is composed of eight partitions: w1a \( \rightarrow \) w8a. For both datasets, each partition starts with a small training set and a large testing one, and ends with a large training set and a small testing one. We run our experiments on each partition. Implementation-wise,
we used the standard LIBSVM code wrapped in the OpenCv library with the RBF kernel $K_{RBF}(x^i, x^j) = e^{-\gamma \|x^i - x^j\|^2}$, and we used the default parameters ($C = 1$). For our approach we adapted the SMO $\nu$-SVM algorithm of [23] with the MC dynamics from [26] and we used $\bar{K}(x^i, x^j) = 1 - K_{RBF}(x^i, x^j)$, $\varepsilon = 10^{-5}$. It can be shown that even with our formulation we can still compute the bias as $b = \sum_{i,j} \alpha^*_i \alpha^*_j y_i \bar{K}(x^i, x^j)$, and that the decision function is $D(x) = b - \sum_i \alpha^*_i y_i \bar{K}(x^i, x)$. While $\nu > 100$ gives similar results as LIBSVM, we are interested in taking advantage of the sparsity constraint and exploring what is the minimum number of support vectors needed to maintain a comparable accuracy. Hence, we set $\nu = 0.1$. We initialize our system with $\leq 10$ random support vectors. We also used the same grid search ($\gamma \in [0, 1]$, $\gamma_{\text{incr.}} = 10^{-5}$) for both methods for fair comparison. Table I shows the baseline on the aforementioned datasets and Table II includes the results of our algorithm. While our solution provides comparable accuracy, it is much sparser in terms of numbers of support vectors and only requires a few iterations to converge. This leads to a training time an order of magnitude smaller than the standard LIBSVM approach, as illustrated Fig. (7).

### Table I

<table>
<thead>
<tr>
<th># feat.</th>
<th>train. size</th>
<th>test. size</th>
</tr>
</thead>
<tbody>
<tr>
<td>adu.</td>
<td>123</td>
<td>1605 → 32561</td>
</tr>
<tr>
<td>web.</td>
<td>300</td>
<td>2477 → 49749</td>
</tr>
<tr>
<td>cod.</td>
<td>8</td>
<td>59535</td>
</tr>
<tr>
<td>spl.</td>
<td>60</td>
<td>10000</td>
</tr>
</tbody>
</table>

### Table II

<table>
<thead>
<tr>
<th>acc. LIBSVM (%)</th>
<th>adu.</th>
<th>web.</th>
<th>cod.</th>
<th>spl.</th>
</tr>
</thead>
<tbody>
<tr>
<td>79.5 ± 0.5</td>
<td>97.7 ± 0.3</td>
<td>66.7</td>
<td>83.6</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>acc. ours (%)</th>
<th>adu.</th>
<th>web.</th>
<th>cod.</th>
<th>spl.</th>
</tr>
</thead>
<tbody>
<tr>
<td>79.1 ± 1</td>
<td>96.9 ± 0.3</td>
<td>69.8</td>
<td>83.6</td>
<td></td>
</tr>
</tbody>
</table>

| #SV LIBSVM      | 200  | ~200 | 200  | 200  |
| #SV ours        | 13.5 ± 4 | 7.75 ± 4 | 18   | 13.6 |

### Table III

| acc. LIBSVM (%), number of support vectors (#SV) and training time (T.T.) for the best parameter $\gamma$ on SVM binary class datasets for LIBSVM and our approach. |

### V. CONCLUSION AND FUTURE WORK

In this work, we demonstrated that it is possible to reformulate QP problems using directly graph theory problem transformations. Our maximum clique formulation of SVM has similar accuracy as the original formulation. At the same time, it provides a significantly smaller training time and number of support vectors. The main advantage of graph theory QP reformulation is two-fold: (1) it allows the use of any MWC algorithm to train support vectors, and (2) it applies SMO-like algorithms to solve other application formulated as a MWC. This application can also be generalized to other graph problems, including formulating support vector clustering as a maximum flow problem or using weighted k-mean for spectral clustering. This also applies to a formulation to a more general problem, as shown in the previous section. In the future, we plan to tackle other applications including multi-class SVM, image and video segmentation, and document summarization. Next, we will briefly explain how to adapt our model to these new applications.

So far we presented graph theory formulation of SVM for the binary classification case. In the remaining of this section...
we show how we can extend the MWC formulation to the multi-class problem. Following our SVM interpretation, if two support vectors are similar and are from the same class, or dissimilar and from different classes, they are positively correlated. Otherwise, they should repulse each other. We modify the edge weight by replacing the product $y_i y_j$ with $2\delta(y_i, y_j) - 1$, where $\delta(y_i, y_j) = 1$ if $y_i = y_j$, 0 otherwise. We also extend the balancing constraint $y^\top \alpha$ by substituting it with $\sum_{y_i = c} \alpha_i \leq \frac{k}{c}$ for every class $c$, where $k$ is the total number of classes. We lose the equality as some classes need more support vectors, depending on the surrounding vectors of other classes. Indeed, pairs $\{c_l, c_m\}$ of classes still need to be exactly balanced, but not all support vectors from class $c_l$ will contribute to the boundary between class $c_l$ and $c_m$, as they may contribute to the boundary between $c_l$ and an other one than $c_m$. Then final multi-class MWC formulation can be summarized as:

$$\alpha^* \leftarrow \arg \max_{\alpha} \quad J(\alpha) = \frac{1}{2} \alpha^\top \mathbf{H} \alpha$$

s.t. $\mathbf{C} \alpha \leq \nu \mathbf{1}$, $\mathbf{C} \in \mathbb{R}^{k \times n}$ and $\alpha \in [0, 1]^n$

with $\mathbf{H}_{ij} = \left(2\delta(y_i, y_j) - 1\right) \mathbf{K}(x_i, x_j)$

$y \in \{1, 2, ..., k\}^n$ and $x \in \mathbb{R}^d$

where $C_{ij} = \delta(y_i, y_j)$, i.e. $C_{ij} = 1$ if $y_j = i$, 0 otherwise. Note that the extension could have also been applied in the original formulation Eqn.(1), and that the same $\nu$-SVM algorithm of [23] can easily be adapted to the multi-class problem. The initialization strategy described in Section III-B holds as well for the multi-class problem.

VI. ACKNOWLEDGEMENTS

This work would not have been possible without the guidance and support of David Salesin, Joel Brandt, and Bobby Murphy. We would also like to thank Melanie Poblete and Aletta Hiemstra for their help in proofreading the document and the anonymous reviewers for their useful comments.

REFERENCES

On dynamic ensemble selection and data preprocessing for multi-class imbalance learning

Rafael M. O. Cruz  
*Ecole de Technologie Supérieure*  
Montreal, Canada  
rafaelmnelau@gmail.com

Robert Sabourin  
*Ecole de Technologie Supérieure*  
Montreal, Canada  
Robert.Sabourin@etsmtl.ca

George D. C. Cavalcanti  
*Centro de Informática*  
Universidade Federal de Pernambuco  
Recife, Brazil  
gdcc@cin.ufpe.br

Abstract—Class-imbalance refers to classification problems in which many more instances are available for certain classes than for others. Such imbalanced datasets require special attention because traditional classifiers generally favor the majority class which has a large number of instances. Ensemble of classifiers have been reported to yield promising results. However, the majority of ensemble methods applied too imbalanced learning are static ones. Moreover, they only deal with binary imbalanced problems. Hence, this paper presents an empirical analysis of dynamic selection techniques and data preprocessing methods for dealing with multi-class imbalanced problems. We considered five variations of preprocessing methods and four dynamic selection methods. Our experiments conducted on 26 multi-class imbalanced problems show that the dynamic ensemble improves the F-measure and the G-mean as compared to the static ensemble. Moreover, data preprocessing plays an important role in such cases.

Index Terms—Imbalanced learning, multi-class imbalanced, ensemble of classifiers, dynamic classifier selection, data preprocessing

I. INTRODUCTION

Class-imbalance [1] refers to classification problems in which many more instances are available for certain classes than for others. Particularly, in a two-class scenario, one class contains the majority of instances (the *majority class*), while the other (the *minority class*) contains fewer instances. Imbalanced datasets may originate from real life problems including the detection of fraudulent bank account transactions, telephone calls, biomedical diagnosis, image retrieval and so on.

One of the biggest challenges in imbalanced learning is dealing with multi-class imbalanced problems [2]. Multi-class imbalanced classification is not as well developed as the binary case, with only a few papers handling this issue [3, 4, 5]. It is also considered as a more complicated problem, since the relation among the classes is no longer obvious. For instance, one class may be the majority one when compared to some classes, and minority when compared to others. Moreover, we may easily lose performance on one class while trying to gain it on another [4].

One way of dealing with imbalanced distributions is to use ensemble learning. As shown in [6], a diverse ensemble can better cope with imbalanced distribution. In particular, Dynamic selection (DS) techniques is seen as an alternative to deal with multi-class imbalance as it explores the local competence of each base classifier according to each new query sample [7, 2, 8]. Only the base classifiers that attained a certain competence level, in the given local region, are selected to predict the label of the query sample.

A key factor in dynamic selection is the estimation of the classifiers’ competences according to each test sample. Usually the estimation of the classifiers competences are based on a set of labeled samples, called the dynamic selection dataset (DSEL). However, As reported in [9], dynamic selection performance is very sensitive to the distribution of samples in DSEL. If the distribution of DSEL itself becomes imbalanced, then there is a high probability that the region of competence for a test instance will become lopsided. Thus, the dynamic selection algorithms might end up biased towards selecting base classifiers that are experts for the majority class. With this in mind, we propose the use of data preprocessing methods for training a pool of classifiers as well as balancing the class distribution in DSEL for the DS techniques.

Hence, in this paper, we perform a study on the application of dynamic selection techniques and data preprocessing for handling with multi-class imbalance. Five data preprocessing techniques and four DS techniques as well as static ensemble combination are considered in our experimental analysis. Experiments are conducted using 26 multi-class imbalanced datasets with varying degrees of class imbalance. The following research questions are studied in this paper:

1) Does data preprocessing play an important role in the performance of dynamic selection techniques?
2) Which data preprocessing technique is better suited for dynamic and static ensemble combination?
3) Do dynamic ensembles present better performance than static ensembles?

This paper is organized as follows: Section II presents the related works on dynamic selection and describes the DCS and DES methods considered in this analysis. Data preprocessing techniques for imbalance are presented in Section III. Experiments are conducted in Section IV. Conclusion and future works are presented in the last section.

II. DYNAMIC SELECTION

A dynamic selection (DS) enables the selection of one or more base classifiers from a pool, given a test instance. This is based on the assumption that each base classifier is
an expert in a different local region in the feature space. Therefore, the most competent classifiers should be selected in classifying a new instance. The notion of competence is used in DS as a way of selecting, from a pool of classifiers, the best classifiers to classify a given test instance. Usually, the competence of a base classifier is estimated based on a small region in the feature space surrounding a given test instance, called the region of competence. This region is formed using the k-nearest neighbors (KNN) technique, with a set of labeled samples, which can be either the training or validation set. This set is called the Dynamic Selection dataset (DSEL) [7]. To establish the competence, given a test instance and the DSEL, the literature reports a number of measures classified into individual-based and group-based measures [7]. Implementation of several DS techniques can be found on GitHub: https://github.com/Menelau/DESlib [10].

Among the categories, we focus on the individual-based measures, which consider individual base classifier accuracy for the region of competence. However, the competency measures are calculated differently by different methods in this category. For example, we consider methods in which the competency is measured by pure accuracy [11], by ranking of classifiers [12] or using oracle information [13].

Instead of grouping DS strategies by competence measure, we may also group them by selection methodology. Currently, there are two kinds of selection strategies: dynamic classifier selection (DCS) and dynamic ensemble selection (DES). DCS selects a single classifier for a test instance, whereas DES selects an ensemble of classifiers (EoC) to classify a test instance. Both these strategies have been studied in recent years, and some papers are available examining them [14, 15]. In this paper, we evaluate two DCS and two DES strategies:

- **The Modified Classifier Rank (RANK)** [12] is a DCS method that exploits ranks of individual classifiers in the pool for each test instance. The rank of a classifier is based on its local accuracy within a neighborhood of the test instance. More formally, given a test instance assigned to class \( C_i \) by a classifier, the ranking of the classifier is estimated as the number of consecutive nearest neighbors assigned to class \( C_i \) that have been correctly labeled. The most locally accurate classifier has the highest rank and is selected for classification.

- **The Local Class Accuracy (LCA)** [11] estimates the classifier accuracy in a local region around the given test instance and then uses the most locally accurate classifier to classify the test instance. The local accuracy is estimated for each base classifier as the percentage of correct classifications within the local region, but considering only those examples where the classifier predicted the same class as the one it gave for the test instance. The classifier presenting the highest local accuracy is used for the classification of the query sample.

- **The KNORA-Eliminate technique (KNE)** [13] explores the concept of Oracle, which is the upper limit of a DCS technique. Given the region of competence \( \theta_j \), only the classifiers that correctly recognize all samples belonging to the region of competence are selected. In other words, all classifiers that achieved a 100% accuracy in this region (i.e., that are local Oracles) are selected to compose the ensemble of classifiers. Then, the decisions of the selected base classifiers are aggregated using the majority voting rule. If no base classifier is selected, the size of the region of competence is reduced, and the search for the competent classifiers is restarted.

- **The KNORA-Union technique (KNU)** [13] selects all classifiers that are able to correctly recognize at least one sample in the region of competence. This method also considers that a base classifier can participate more than once in the voting scheme when it correctly classifies more than one instance in the region of competence. The number of votes of a given base classifier \( c_i \) is equal to the number of samples in the region of competence, for which it predicted the correct label. For instance, if a given base classifier \( c_i \) predicts the correct label for three samples belonging to the region of competence, it gains three votes for the majority voting scheme. The votes collected by all base classifiers are aggregated to obtain the ensemble decision.

These DS techniques are based on different criterion to estimate the local competence of the base classifiers. For example, while both the RANK and the LCA are DCS strategies, the former measures the competence based on ranking, and the latter based on classifier accuracy. On the other hand, the two DES strategies (KNE and KNU) are based on Oracle information. These techniques were selected as they were among the best performing DS methods according to the experimental analysis conducted in [7].

Nevertheless, a crucial aspect in the performance of the dynamic selection techniques is the distribution of the dynamic selection dataset (DSEL), as the local competence of the base classifiers are estimated based on this set. Hence, preprocessing techniques can really benefit DS techniques as they can be employed to edit the distribution of DSEL, prior to performing dynamic selection.

### III. Data Preprocessing

Changing the distribution of the training data to compensate for poor representativeness of the minority class is an effective solution for imbalanced problems, and a plethora of methods are available in this regards. Branco et al. [16] divided such methods into three categories, namely, stratified sampling, synthesizing new data, and combinations of the two previous methods. While the complete taxonomy is available in [16], we will center our attention on the methods that have been used together with ensemble learning [6].

One important category is under-sampling, which removes instances from the majority class to balance the distribution. Random under-sampling (RUS) [17] is one such method. RUS has been coupled with boosting (RUSBoost) [18] and with Bagging [17]. A major drawback of RUS is that it can discard potentially useful data which can be a problem when using dynamic selection approaches.
The other strategy is the generation of new synthetic data. Synthesizing new instances has several known advantages [19], and a wide number of proposals are available for building new synthetic examples. In this context, a famous method that uses interpolation to generate new instances is SMOTE [19]. SMOTE over-samples the minority class by generating new synthetic data. A number of methods have been developed based on the principle of SMOTE, such as, Borderline-SMOTE [20], ADASYN [21], RAMO [22] and Random balance [23]. Furthermore, Garcia et al. [24] observed that over-sampling consistently outperforms under-sampling for strongly imbalanced datasets.

Hence, in this work we considered three over-sampling techniques. Similar to [3], the class with the highest number of examples is considered the majority class, while all others are considered minority classes. Then, the over-sampling techniques are applied to generate synthetic samples for each minority class.

- **Synthetic Minority Over-sampling Technique (SMOTE)** [19], which creates artificial instances for the minority class. The process works as follows: Let \( x_i \) be an instance from the minority class. To create an artificial instance from \( x_i \), SMOTE first isolates the k-nearest neighbors of \( x_i \), from the minority class. Afterward, it randomly selects one neighbor and randomly generates a synthetic example along the imaginary line connecting \( x_i \) and the selected neighbor.

- **Ranked Minority Over-sampling (RAMO)** [22], which performs a sampling of the minority class according to a probability distribution, followed by the creation of synthetic instances. The RAMO process works as follows: For each instance \( x_i \) in the minority class, its \( k_1 \) nearest neighbors (\( k_1 \) is a user defined neighborhood size) from the whole dataset are isolated. The weight \( r_i \) of \( x_i \) is defined as:

\[
  r_i = \frac{1}{1 + \exp(-\alpha \cdot \delta_i)},
\]

where \( \delta_i \) is the number of majority cases in the k-nearest neighborhood. Evidently, an instance with a large weight indicates that it is surrounded by majority class samples, and thus difficult to classify.

After determining all weights, the minority class is sampled using these weights to get a sampling minority dataset \( G \). The synthetic samples are generated for each instance in \( G \) by using SMOTE on \( k_2 \) nearest neighbors where \( k_2 \) is a user-defined neighborhood size.

- **Random Balance (RB)** [23], which relies on the amount of under-sampling and over-sampling that is problem specific and that has a significant influence on the performance of the classifier concerned. RB maintains the size of the dataset, but varies the proportion of the majority and minority classes, using a random ratio. This includes the case where the minority class is over represented and the imbalance ratio is inverted. Thus, repeated applications of RB produce datasets having a large imbalance ratio variability, which promotes diversity [23]. SMOTE and random under-sampling are used to respectively increase or reduce the size of the classes to achieve the desired ratios.

Given a dataset \( S \), with minority class \( S_P \) and majority class \( S_N \), the RB procedure can be described as follows:

1) The modified size, \( newMajSize \), of the majority class, is defined by a random number generated between 2 and \(|S| - 2 \) (both inclusive). Accordingly, the modified size, \( newMinSize \), of the minority class becomes \(|S| - newMajSize \).

2) If \( newMajSize < |S_N| \), the majority class \( S_P \) is created by RUS the original \( S_N \) so that the final size \(|S_N'| = newMajSize \). Consequently, the new minority class \( S_P' \) is obtained from \( S_P \) using SMOTE to create \( newMinSize - |S_P'| \) artificial instances.

3) Otherwise, \( S_P' \) is the class created by RUS \( S_P \). On the other hand, \( S_N' \) is the class that includes artificial samples generated using SMOTE on \( S_N \). Thus, finally, \(|S_P'| = newMinSize \) and \(|S_N'| = newMajSize \).

IV. EXPERIMENTS

A. Datasets

A total of 26 multi-class imbalanced datasets taken from the Keel repository [25] was used in this analysis. The key features of the datasets are presented in Table I. The IR is computed as the proportion of the number of the majority class examples to the number of minority class examples. In this case, the class with maximum number of examples is the majority class, and the class with the minimum number of examples is the minority one. We grouped the datasets according to their IRs using the group definitions suggested by [26]. Datasets with low IR (\( IR < 3 \)) are highlighted with dark gray, whereas datasets with medium IR (\( 3 < IR < 9 \)) are in light gray.

B. Experimental setup

The Weka 3.8 along with Matlab 8.4.0 was used in the experiments. Results were obtained with a \( 5 \times 2 \) stratified cross-validation. Performance evaluation is conducted using the multi-class generalization of the AUC, F-measure and G-mean, as the standard classification accuracy is not suitable for imbalanced learning [6].

The pool size for all ensemble techniques was set to 100. The classifier used as a base classifier in all ensembles was J48, which is the Java implementation of Quinlan’s C4.5, available in Weka 3.8. Here, C4.5 was used with Laplace smoothing at the leaves, but without pruning and collapsing as recommended in [6].

All preprocessing techniques were combined with Bagging during the pool generation phase. Table II lists such combinations. The preprocessing techniques, RAMO and SMOTE, have user-specified parameters. In the case of RAMO, we used \( k_1 = 10, k_2 = 5 \) and \( \alpha = 0.3 \). For SMOTE and RB, the number of nearest neighbors was 5. These parameter settings...
TABLE I
CHARACTERISTICS OF THE 26 MULTI-CLASS IMBALANCED DATASETS TAKEN FROM THE KEEL REPOSITORY. COLUMN #E SHOWS THE NUMBER OF INSTANCES IN THE DATASET, COLUMN #A THE NUMBER OF ATTRIBUTES (NUMERIC/NOMINAL), #C SHOWS THE NUMBER OF CLASSES IN THE DATASET, AND COLUMN IR THE IMBALANCE RATIO.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#E</th>
<th>#A</th>
<th>#C</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle</td>
<td>846</td>
<td>180</td>
<td>4</td>
<td>1.09</td>
</tr>
<tr>
<td>Wine</td>
<td>178</td>
<td>130</td>
<td>3</td>
<td>1.48</td>
</tr>
<tr>
<td>LedDigit</td>
<td>500</td>
<td>70</td>
<td>10</td>
<td>1.54</td>
</tr>
<tr>
<td>Contraception</td>
<td>1473</td>
<td>90</td>
<td>3</td>
<td>1.89</td>
</tr>
<tr>
<td>Hayes-Roth</td>
<td>160</td>
<td>40</td>
<td>3</td>
<td>2.10</td>
</tr>
<tr>
<td>Column3C</td>
<td>310</td>
<td>60</td>
<td>3</td>
<td>2.50</td>
</tr>
<tr>
<td>Satimage</td>
<td>6835</td>
<td>100</td>
<td>3</td>
<td>2.45</td>
</tr>
<tr>
<td>Lymphography</td>
<td>253</td>
<td>90</td>
<td>3</td>
<td>2.49</td>
</tr>
<tr>
<td>New-thyroid</td>
<td>215</td>
<td>50</td>
<td>3</td>
<td>5.00</td>
</tr>
<tr>
<td>Dermatology</td>
<td>358</td>
<td>230</td>
<td>6</td>
<td>5.55</td>
</tr>
<tr>
<td>Balance</td>
<td>625</td>
<td>40</td>
<td>3</td>
<td>5.88</td>
</tr>
<tr>
<td>Flare</td>
<td>1066</td>
<td>90</td>
<td>6</td>
<td>7.70</td>
</tr>
<tr>
<td>Glass</td>
<td>214</td>
<td>90</td>
<td>6</td>
<td>8.44</td>
</tr>
<tr>
<td>Abalone</td>
<td>4593</td>
<td>3</td>
<td>8</td>
<td>14.05</td>
</tr>
<tr>
<td>Post-Operative</td>
<td>1599</td>
<td>110</td>
<td>6</td>
<td>16.00</td>
</tr>
<tr>
<td>Thyroid</td>
<td>5720</td>
<td>210</td>
<td>3</td>
<td>40.16</td>
</tr>
<tr>
<td>Ecoli</td>
<td>336</td>
<td>70</td>
<td>8</td>
<td>51.50</td>
</tr>
<tr>
<td>Page-blocks</td>
<td>5472</td>
<td>100</td>
<td>5</td>
<td>175.46</td>
</tr>
<tr>
<td>Nursery</td>
<td>12690</td>
<td>10</td>
<td>18</td>
<td>45.93</td>
</tr>
<tr>
<td>B100</td>
<td>4139</td>
<td>3</td>
<td>8</td>
<td>2160.00</td>
</tr>
</tbody>
</table>

were adopted from [6]. Finally, for all the dynamic selection methods, we used 7 nearest neighbors to define the region of competence as in [15, 7].

TABLE II
PREPROCESSING METHODS USED FOR CLASSIFIER POOL GENERATION.

<table>
<thead>
<tr>
<th>Preprocessing Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ba</td>
<td>Bagging</td>
</tr>
<tr>
<td>Ba-RM100</td>
<td>Bagging+RAMO 100%</td>
</tr>
<tr>
<td>Ba-RM</td>
<td>Bagging+RAMO</td>
</tr>
<tr>
<td>Ba-SM100</td>
<td>Bagging+SMOTE 100%</td>
</tr>
<tr>
<td>Ba-SM</td>
<td>Bagging+SMOTE</td>
</tr>
<tr>
<td>Ba-RB</td>
<td>Bagging+RB</td>
</tr>
</tbody>
</table>

The complete framework for a single replication is presented in Figure 1. The original dataset was divided into two equal halves. One of them was set aside for testing, while the other half was used to train the base classifiers and to derive the dynamic selection set. Let us now highlight the process of setting up the DSEL. Here, instead of dividing the training set, we augment it using the data preprocessing, to create DSEL. Moreover, the Bagging method is applied to the training set, generating a bootstrap with 50% of the data. Then, the preprocessing method is applied to each bootstrap, and the resulting dataset is used to generate the pool of classifiers. Since we considered a single training dataset, the DSEL dataset has an overlap with the datasets used during Bagging iterations. However, the randomized nature of the preprocessing methods allows the DSEL not to be exactly the same as the training datasets. Thus, avoiding overfitting issues.

C. Results according to data preprocessing method

In this section, we compare the performance of each preprocessing method with respect to each ensemble technique. Tables III, IV and V show the average rank for the AUC, F-measure and G-mean, respectively. The best average rank is in bold. We can see that the SM and SM100 obtained the best results. Furthermore, the configuration using only Bagging always presented the highest average rank.

The Finner’s [27] step-down procedure was conducted at a 95% significance level to identify all methods that were equivalent to the best ranked one. The analysis demonstrates that considering the F-measure and G-mean, the result obtained using preprocessing techniques is always statistically better when compared to using only Bagging.

Moreover, we conducted a pairwise comparison between each ensemble method using data preprocessing with the same methods using only Bagging (baseline). For the sake of simplicity, only the best data preprocessing for each technique was considered (i.e., the best result of each row of Tables III, IV and V). The pairwise analysis is conducted using the Sign test, calculated on the number of wins, ties, and losses obtained by each method using preprocessing techniques, compared
The results of the Sign test are presented in Figure 2.

The Sign test demonstrated that the data preprocessing significantly improved the results of these techniques according to the F-measure and G-mean. Considering these two measures, all techniques obtained a significant number of wins for a significance level \( \alpha = 0.05 \). Moreover, three out of five techniques presented a significant number of wins for \( \alpha = 0.01 \). Hence, the results obtained demonstrate that data preprocessing techniques indeed play an important role when dealing with multi-class imbalanced problems.

Furthermore, DS techniques are more benefited from the application of data preprocessing (i.e., presented a higher number of wins). This results can be explained by the fact that the data preprocessing techniques are applied in two stages: First, it is used in the ensemble generation stage in order to generate a diverse pool of classifiers. Then, they are also used in order to balance the distribution of the dynamic selection dataset for the estimation of the classifiers’ competences.

### D. Dynamic selection vs static combination

In this experiment we compare the performance of the dynamic selection approaches versus static ones. For each technique, the best performing data preprocessing technique is selected (i.e., best result from each row of Tables III, IV and V). Then, new average ranks are calculated for these methods.

To the baseline. The results of the Sign test is presented in

### Table IV-D presents the average rank of the top techniques according to each metric.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Bagging</th>
<th>RM</th>
<th>RM100</th>
<th>SM</th>
<th>SM100</th>
<th>RB</th>
</tr>
</thead>
<tbody>
<tr>
<td>KNE</td>
<td>3.00</td>
<td>1.35</td>
<td>1.75</td>
<td>2.25</td>
<td>2.66</td>
<td>3.82</td>
</tr>
<tr>
<td>KNU</td>
<td>4.42</td>
<td>3.08</td>
<td>3.42</td>
<td>3.35</td>
<td>3.35</td>
<td>3.54</td>
</tr>
<tr>
<td>LCA</td>
<td>3.92</td>
<td>3.42</td>
<td>3.62</td>
<td>3.00</td>
<td>2.38</td>
<td>4.65</td>
</tr>
<tr>
<td>RANK</td>
<td>4.19</td>
<td>3.46</td>
<td>3.69</td>
<td>3.38</td>
<td>2.54</td>
<td>3.73</td>
</tr>
<tr>
<td>STATIC</td>
<td>4.00</td>
<td>3.54</td>
<td>3.15</td>
<td>3.46</td>
<td>2.81</td>
<td>4.04</td>
</tr>
</tbody>
</table>

Based on the average ranks, we can see that the DES techniques present a lower average rank when compared to that of the static combination for the three performance measures. Hence, DES techniques are suitable for dealing with multi-class imbalance. The performance of DES techniques (KNE and KNU) and the static combination were statistically equivalent considering the F-measure and G-mean, while the performance of the KNU was significantly better considering the AUC. On the other hand, the DCS techniques (LCA and RANK) presented a higher average rank when compared to the static ensemble, and may not be suitable to handle multi-class imbalanced problems.

### V. Conclusion

In this work, we conducted a study on dynamic ensemble selection and data preprocessing for solving the multi-class imbalanced problems. A total of four dynamic selection techniques and five preprocessing techniques were evaluated in this experimental study.

Results obtained over 26 multi-class imbalanced problems demonstrate that the dynamic ensemble selection techniques studied (KNE and KNU) obtained a better result than static ensembles based on AUC, F-measure and G-mean. Moreover, the use of data preprocessing significantly improves the performance of DS and static ensembles. In particular, the SMOTE technique presented the best results. Furthermore, DS techniques seems to benefit more of data preprocessing methods since they are applied not only to generate the pool of classifiers but also to edit the distribution of the dynamic selection dataset.
Future works would involve the definition of new preprocessing techniques specific to deal with multi-class imbalance as well as the definition of cost-sensitive dynamic selection techniques to handle multi-class imbalanced problems.

REFERENCES

Multi-label Pixelwise Classification for Reconstruction of Large-scale Urban Areas

Yuanlie He
School of Computer Science and Technology
Guangdong University of Technology
University-Town, Guangzhou
P. R. China (510006)
Email: heyuanlie@gdut.edu.cn

Sudhir Mudur
Immersive and Creative Technologies Lab
Concordia University, CA
Email: sudhir.mudur@concordia.ca

Charalambos Poullis
Immersive and Creative Technologies Lab
Concordia University, CA
Email: charalambos@poullis.org

Abstract—Object classification is one of the many holy grails in computer vision and as such has resulted in a very large number of algorithms being proposed already. Specifically in recent years there has been considerable progress in this area primarily due to the increased efficiency and accessibility of deep learning techniques. In fact, for single-label object classification [i.e. only one object present in the image] the state-of-the-art techniques employ deep neural networks and are reporting very close to human-like performance.

There are specialized applications in which single-label object-level classification will not suffice; for example in cases where the image contains multiple intertwined objects of different labels. In this paper, we address the complex problem of multi-label pixelwise classification. We present our distinct solution based on a convolutional neural network (CNN) for performing multi-label pixelwise classification and its application to large-scale urban reconstruction. A supervised learning approach is followed for training a 13-layer CNN using both LiDAR and satellite images. An empirical study has been conducted to determine the hyperparameters which result in the optimal performance of the CNN. Scale invariance is introduced by training the network on five different scales of the input and labeled data. This results in six pixelwise classifications for each different scale. An SVM is then trained to map the six pixelwise classifications into a single-label. Lastly, we refine boundary pixel labels using graph-cuts for maximum a-posteriori (MAP) estimation with Markov Random Field (MRF) priors. The resulting pixelwise classification is then used to accurately extract and reconstruct the buildings in large-scale urban areas. The proposed approach has been extensively tested and the results are reported.

Keywords—urban reconstruction, remote sensing processing

I. INTRODUCTION

Recent advances in the efficiency and accessibility of deep learning techniques have had a significant impact on the progress of many important and at the time dormant problems in computer vision. In particular, object recognition has greatly benefited since for many years the state-of-the-art had been almost restricted to minimal and incremental progress whereas currently human-like performances in object recognition [14],[15] are being reported, albeit for images with a single object.

Many successful applications which rely on single-label object recognition using deep neural networks have already been reported. The assumption with single-label object recognition is that a network can be trained to recognize objects from various categories and identify their general location [in the form of a bounding box] provided that the image contains exactly one object. More recently, it has been shown how this learning pipeline can be extended to handle cases where multiple spatially separable objects are present in an image i.e multi-label object recognition [16]. However, in certain applications the images may contain objects which are intertwined. Moreover, rather than the general location, the precise location of the object is required. One such application is the classification of geospatial objects for reconstruction of large-scale urban areas. The data is in the form of geometry captured with LiDAR and satellite RGB images. The geospatial objects present in the data are buildings, roads, trees, artificial ground, natural ground, cars, etc and all these are perfectly intertwined i.e. a building is surrounded by ground, etc. One can think of the data (LiDAR, images) as being perfectly tesselated by many objects from each of these categories.

In this paper we address the problem of multi-label pixelwise classification for large scale urban reconstruction and propose our distinct solution based on a 13-layer convolutional neural network (CNN). The CNN is trained using both LiDAR and satellite RGB images in multi-scale format, capturing large-scale urban areas and produces an output of six pixelwise values which are interpreted as likelihoods of the pixel in being a building, road, tree, car, or ground [artificial, natural]. An SVM linear classifier takes the likelihoods as inputs and maps them to a single label. In the final step, boundary pixel labels are refined.

Our technical contributions are:

• the design, development and supervised training of a 13-layer convolutional network. The CNN is specifically designed for the classification of geospatial objects appearing in remote sensor data and in particular LiDAR and satellite RGB images into the following six classes: buildings, roads, tree, cars, natural ground and artificial ground.
• a method for introducing scale invariance during the training. Due to the multiple scales the CNN produces a likelihood-per-scale per pixel. These are further processed and combined into a single label by training an SVM. The labels are finally refined using graph cuts for maximum a-posteriori (MAP) estimation with Markov Random Field (MRF) priors which also ad-
addresses the problem of boundary pixels not being assigned labels.

- a complete framework for the geospatial object classification and reconstruction of large-scale urban areas. The multi-label pixelwise classification is used to reconstruct the buildings of large-scale urban areas. Generic objects such as cars and trees are replaced by procedurally generated models to yield realistic 3D visualizations.

A. Paper Organization

The paper is organized as follows: Section II presents an overview of the state-of-the-art in the area of object recognition and large-scale urban reconstruction. In Section III we present a technical overview of our proposed technique and in Section IV we provide a brief description of the dataset used. The architecture of the developed network is described in detail in Section VI including the training, refinement, validation and classification results. Section VII presents how these classification results are used in the context of large-scale urban reconstruction. The conclusion and future work are discussed in Section VII.

II. Related Work

Object classification has been an active research topic in computer vision for many years and large-scale urban reconstruction for even more. In fact, object recognition is often employed as the first step in reconstruction for identifying the geospatial objects present in the scene. In this section we provide a brief overview of the state-of-the-art in the areas related to this work in object recognition using neural networks and in large-scale urban reconstruction.

A. Object Classification

The first Convolutional Neural Networks (CNN) was introduced by LeCun [1] for hand writing recognition and could achieve outstanding performance. Yang et al. [2] later extended the CNN with an additional layer for Support Vector Machine (SVM) which could detect and classify traffic signs. They demonstrated excellent performance and reported classification accuracies of 98.24% and 98.77% for the GTSDB and CTSD datasets, respectively. In a different application of object classification, Iijina and Chalavadi [3] proposed a method for recognizing human action. Instead of random initialization of the network they propose computing the initial weights of the CNN using genetic algorithms which minimize the classification error; using this method they can achieve classification accuracies of 99.98% and 96.92% for the UCF50 and MNIST datasets. Hu et al. [4] propose a Single Signal Crowd CNN Model for counting dense crowds, and report outstanding performance for the training on the UCSD dataset and testing on the UCF-CROWD dataset. Liang et al. [5] propose a recurrent CNN (R-CNN) for object recognition by incorporating recurrent connections into each convolutional layer. The R-CNN is shown to outperform the state-of-the-art models on the CIFAR-10, CIFAR-100, MNIST and SVHN datasets.

More recently [20] Fully-Convolutional Networks have been shown to produce the best results for multi-label pixelwise classification by training on overlapping patches, however a significant disadvantage is the fact that the produced output is considerably downsamled compared to the input and further processing is required. When dealing with semantic segmentation of fine structures such as the ones appearing in urban datasets this generates spurious results. Of similar performance but same shortcoming is the CRF-based approach proposed in [21] where again upsampling/interpolation is required on the generated output.

B. 3D reconstruction

The state-of-the-art in urban reconstruction can be better categorized according to the type and scale of the input data. For a comprehensive survey of urban reconstruction of various types and scales we refer the reader to the survey by Musialski et al. [6]. In this section, we provide a brief overview of state-of-the-art in large-scale urban reconstruction from remote sensor data, most relevant to our work.

In [19], Zhou et al. propose an automated system which given the exact bounding volume of a building can simplify the geometry based on dual contouring while retaining important features. Using this technique the authors were able to simplify the original geometry considerably. A different technique was presented in [18] where pointcloud data was converted automatically to polygonal 3D models. This technique was applicable directly on the raw pointcloud data without requiring any user interaction. Later, in [17] the authors extended the work to include a fast boundary refinement algorithm based on graph-cuts which was used to refine the boundaries and [24] for extracting appearance information.

On a similar line of research, Lafarge et al. [10] proposed a method which produces excellent reconstructed models from aerial LiDAR which can also handle the vegetation and complex ground. Following a more interactive approach, Arikam et al. [8] proposed a system for generating polyhedral models from semi-dense unstructured point-clouds. Planar surfaces were first extracted automatically based on prior semantic information, and later refined manually by an operator.

The Achilles’ heel of almost all reported work in this area is the geospatial object classification: if an object is misclassified then subsequent steps will most definitely also go wrong. Furthermore, extracting buildings from LiDAR data often produces jagged boundaries which affects the accuracy and quality of the reconstruction. Hence, it is of imperative importance to have as accurate classification as possible at the pixel-level. The proposed neural network achieves this yielding average accuracy in the high ninety percentile for buildings.

III. System Overview

The training dataset is first converted to the input form expected by the network. Scale invariance is introduced by training the network on composite images containing multiple scales of the original depth map and RGB image captured by airborne LiDAR and satellite imaging. For training data, this also involves creation of multi-scale label data.

The overall size of each of these composite images is very large. Hence random samples of a fixed patch size taken from
the composite images are used for training the CNN. The CNN’s output is six floating-point values per input pixel. These are interpreted as the likelihoods of the pixel to being classified with one of the six labels: building, road, tree, car, artificial ground, or natural ground. These likelihoods are used to train a linear classifier which outputs a single label per pixel. In a final step boundary labels are recovered and all labels are refined using graph-cuts for maximum a-posteriori (MAP) estimation with Markov Random Field (MRF) priors.

The training of the network (CNN, linear classifier) is performed on a large urban dataset described in the following Section IV. Once the network has been trained, data not used during the training is processed and labels are generated. The resulting labels are then used to extract only the buildings, cars and trees which are further processed to produce the 3D models representing the urban area.

IV. Dataset

The data used for the training and testing of the proposed network is provided by the International Society for Photogrammetry and Remote Sensing (ISPRS). The data is available as part of the benchmark on urban object detection and 3D building reconstruction [12] and consists of several datasets. In this work, we have used the Potsdam dataset for 2D semantic labeling [11] because of its higher accuracy. The Potsdam dataset consists of 24 image pairs consisting of three $6K \times 6K$ registered images, namely a depth map captured with airborne LiDAR with a sampling density of 5cms, a color satellite image, and the ground truth label map. The label map shows the ground-truth per-pixel classification into six classes: buildings (blue), trees (green), roads (white), natural ground (cyan), artificial ground (red) and cars (yellow). The 'artificial ground’ label contains all areas on the ground that do not correspond to roads but are covered by materials such as asphalt that are typically used for paving roads. In particular, it contains parking lots, pavements, inner courtyards and driveways (if paved). The 'natural ground’ label contains any areas on the ground covered by vegetation other than trees. In particular, it contains lawn and low bushes. The remaining labels are self-explanatory. Figure 1 shows a sample pair available in the Potsdam dataset.

(a) Depth map; contains values ranging from [0,1].
(b) Label map; contains one of six values corresponding to the geospatial feature classes.
(c) Color image; contains an RGB color where each channel ranges from [0,1].

Fig. 1: The Potsdam dataset consists of 24 image pairs. Each pair consists of a $6K \times 6K$ (a) depth map, (b) label map and (c) color image.

V. Network Architecture

The proposed deep neural network consists of a 13-layer Convolutional Neural Network (CNN) and a linear classifier (SVM). A diagram of the network’s architecture is shown in Figure 2.

The input to the network are RGBD values corresponding to pixels contained in a small patch $N \times N$ of the input image. The optimal size of the patch $N \times N$ is determined empirically by varying the size while assessing the performance. Our experiments [the most relevant of which are shown in Figure 9] have shown that the value resulting in optimal performance is $N = 100$. The CNN’s filter’s kernel size $k$ was also determined in a similar fashion and is set to $k = 5$. An extensive empirical study showing the performance of the CNN with respect to different combinations of 3 patch sizes $[34, 70, 100]$ and 7 kernel sizes $[5, 7, 9, 11, 13, 15, 17]$ was performed. Table I shows the optimal performance achieved by the CNN for which the hyperparameters are set to patch size of $100 \times 100$ and kernel size of $5 \times 5$.

Based on the above, a patch of pixels with size $100 \times 100$ and four channels per pixel (RGBD) becomes the input to the
network for all reported results. Similarly a kernel size of $5 \times 5$ is used for all spatial convolutions.

The first spatial convolution layer maps the input image patch into 6 feature maps [of size $96 \times 96$]. The following sub-sampling layer samples the output image of the previous layer with $3 \times 3$ kernel and average pooling, and generates 6 images [of size $48 \times 48$] output. The next layer of the network applies a spatial convolution and maps the 6 input images to 12 output images [of size $44 \times 44$]. Then these images are sub-sampled with $3 \times 3$ kernel max pooling, and $12 (21 \times 21)$ mapping images are generated. All the pixels of the resulting 12 images are fully connected to a linear layer with 5292 nodes i.e. $12 \times 21 \times 21$. Next, the 5292-node linear layer passes through two fully connected linear layers of 120-nodes and 80-nodes respectively. The final linear layer is fully connected with the previous and consists of 6 nodes corresponding to the six labels. Each convolutional operation is followed by the non-linear operation $ReLU(x) = \max(x, 0)$. Thus, the CNN models the following operation,

$$
\Gamma(\Pi(ReLU(\Psi \Pi_{avg}(ReLU(\Psi \times X)))))) \rightarrow \Phi^p
$$

where $p$ is a pixel and $\Phi^p$ is a 6-tuple of values corresponding to the six labels. In the above equation $X$ denotes the input data, $\Psi$ denotes a convolution kernel, $\Gamma(\cdot)$ maps the input to a fully connected linear layer, $\Pi_{max}(\cdot)$ is the max-pooling operation, $\Pi_{avg}(\cdot)$ is the average-pooling operation, and $ReLU(\cdot)$ is the rectified linear unit function.

As previously mentioned, the output of the CNN network is a 6-tuple $\Phi$ of values for each pixel $p$ contained in the input patch

$$
\Phi^p = < \phi^p_1, ..., \phi^p_6 >
$$

where each component in $\Phi^p$ is interpreted as the unnormalized likelihood of the pixel $p$ to be classified with any one of the six labels. $\phi^p_1, ..., \phi^p_6$ represent the unnormalized probabilities of building, tree, road, artificial ground, natural ground and car respectively. We define $\Lambda^p$ as the 6-tuple of normalized likelihoods given by,

$$
\Lambda^p = \frac{e^{\Phi^p}}{\omega} = < \lambda^p_1, ..., \lambda^p_6 > = < \frac{e^{\Phi^p_1}}{\omega}, ..., \frac{e^{\Phi^p_6}}{\omega} >
$$

where $\omega = \sum_{i=1}^6 e^{\Phi^p_i}$ such that $\sum_{i=0}^6 \lambda^p_i = 1$.

An example of the output is shown in Figure 3. The components of the per-pixel likelihoods $\Lambda^p$ are grouped according to the labels and are shown as six images. The range of values of each individual component of $\Lambda$ is $[0,1]$.

As it is evident from Figure 3, the output at this point is a tuple $\Lambda$ for each pixel in the input composite image. This means that for each pixel in the original [non-composite] image there will be essentially five sets of likelihoods within each composite image; one for each scale. The five sets of likelihoods $\Lambda_i$ with $1 \leq i \leq 5$ corresponding to each pixel are combined into a single tuple $\Lambda$ by first up-scaling the images to the original $6k \times 6k$ resolution and then averaging the per-pixel likelihoods. Pixels lying on the boundaries for which not every scale may output a likelihood are not assigned likelihoods.

The resulting $6k \times 6k$ set of normalized likelihoods $\bar{\Lambda}$ corresponding to each pixel $p$ and the original label map with the same resolution $6k \times 6k$ becomes the input to a linear classifier (SVM). After training, the SVM learns weights $W$ and bias $b$ of the mapping function $f(W \times \bar{\Lambda} + b) \rightarrow l_i$ where $l_i, 1 \leq i \leq 6$ indicates one of the six labels. Figure 4 shows the result of this process on the likelihoods corresponding to the label ‘building’. Figure 5 shows the final output of the SVM and Figure 5 shows the ground truth for the labels. It should be noted that at this point, boundary pixels cannot be assigned a label.

A. Training

The Potsdam dataset consists of 24 pairs of images. The training is performed on 20 randomly selected image pairs and validated against the remaining 4 image pairs. Inspired by DenseNet [13], we incorporate scale invariance into the training by preprocessing the original data to create composite images containing multiple resolutions of the original. These composite image pairs [depth, RGB, labels] become the input to the network. A composite contains the five images $l_i$, where $0 \leq i \leq 4$ each with resolution corresponding to $i \times 16\%$ decrements of the original resolution i.e. $6k \times 6k, 5k \times 5k, 4k \times 4k, 3k \times 3k, 2k \times 2k$. The dataset contains considerable variance in the orientations of the geospatial objects hence rotation invariance is implicitly incorporated in the training.

The CNN was trained for 300 epochs on a single machine with the following specifications: Intel Core i7-6700K CPU @ 4.00GHz 8, 16GB RAM, 12GB NVidia GeForce GTX TITAN X/PCIe/SSE2. The Torch API was used for the development of the CNN and the code will be made available as open source. The duration of the training for 300 epochs was 26 hours however, as it can be seen from Figure 4 after the first few epochs the training error rapidly reduces and almost converges.

Although the available memory on the GPU is 12GB, the Torch API imposes a restriction on the maximum GPU usage to 2GB. Hence, the available training data cannot be used in its entirety. Instead, given the 20 training image pairs we perform random sampling and gather as many training samples [100 × 100 image patches] as the memory can fit. The uniform random sampling from the 20 image pairs includes patches from various resolutions. We ensure that all pixels within each sampled image patch fall entirely within the same scale. Sampled patches falling on boundaries between different scales are rejected. This results in a total of 400,000 training samples.

Finally, the likelihoods generated by the CNN are combined as previously described and fed into multi-class SVM,
which generates the pixel’s classification after using a one-vs-all learning strategy. The SVM learns how to map the 6-tuple $\Delta^p$ corresponding to each pixel $p$ into a single class which classifies the input with the highest margin. The Torch API was again used for the development of the SVM and the code will be made available as open source. The duration of the training was 27 minutes and was performed on the same machine as above.

B. Maximum-a-posteriori Inference with Markov Random Field Priors for Label Refinement

The linear classifier combines the six pixel-wise likelihoods produced by the CNN into a single label. Pixels along the boundaries of the image cannot be assigned a label because...
the convolutional filter falls out of bounds. To overcome this problem, Overfeat [23] first introduced the shift-and-stitch trick where shifted versions of the input were processed and the results interleaved into a full resolution output. However, the computational efficiency of this approach does not scale to the current large-scale urban datasets we are dealing with.

Instead, we propose the use of graph-cuts for maximum-a-posteriori (MAP) estimation with Markov Random Field (MRF) priors. We reformulate the problem as finding an optimal labeling $f_p$ for every pixel $p$ such that $f(p) \rightarrow l$ where $l$ is a new label. In addition to the six labels we include a new label $\text{unknown}$ to account for the boundary pixels which have not been assigned a label. Hence, the set of labels becomes $\{\text{buildings, roads, trees, cars, natural ground, artificial ground, unknown}\}$.

The energy function which is minimized is then given by,

$$E(f) = E_{\text{unary}}(f) + E_{\text{pairwise}}(f)$$ (4)  

The unary energy term $E_{\text{unary}}(f)$ provides a measure of the compatibility of the new label under the labeling $f(p_i)$ to the pixel $p_i$ with label $l_{p_i}$ in the observed data and is given by,

$$E_{\text{unary}}(f) = \sum_{i=0}^{N} \begin{cases} 
10, & \text{if } f(p_i) \neq l_{p_i}, \\
15, & \text{if } f(p_i) = \text{unknown}, \\
0, & \text{if } f(p_i) = l_{p_i}, 
\end{cases}$$ (5)

The pairwise energy term $E_{\text{pairwise}}(f)$ provides a measure of compatibility of the new labels under the labeling $f(p_i), f(p_j)$ for neighbouring pixels $p_i$ and $p_j$ respectively and is given by,

$$E_{\text{pairwise}}(f) = \sum_{i,j=0}^{N} \begin{cases} 
0, & \text{if } l_{p_i} = l_{p_j}, \\
20, & \text{otherwise}. 
\end{cases}$$ (6)

The pairwise measure $V(f_{p_i}, f_{p_j})$ between neighbouring pixels $p_i, p_j$, and $p_k$ can be trivially shown to be metric since the following conditions are true,

$$V(f(p_i), f(p_j)) = 0 \leftrightarrow i = j$$  
$$V(f(p_i), f(p_j)) = V(f(p_j), f(p_i)) > 0$$  
$$V(f(p_i), f(p_j)) \leq V(f(p_i), f(p_k)) + V(f(p_k), f(p_j))$$ (7)

This is a multi-label MRF problem with non-submodular energy potentials and as such can only be approximately solved. The alpha-expansion algorithm is used to break the multi-label problem into a series of binary problems. Experiments have shown that after 5 iterations the energy $E(f)$ is reduced on average 12% and the overall accuracy of the classification results increases by $[0.5 - 1\%]$. Table I shows a comparison between the performance before/without and after/with this process. Image boundary pixels for which no label was generated are now relabeled based on the new labeling resulting from the energy minimization using graph-cuts. Figure 5 shows an example output of this process.

![Fig. 4: Training error of 300 iterations](image)

![Fig. 5: Maximum-a-posteriori Inference with Markov Random Field Priors for Label Refinement. (a) The labels generated by the network. Pixels along the image boundaries cannot be assigned a label. (b) Dense and refined labeling resulting from the proposed method. Pixels along the image boundary are assigned a label.](images)
can be seen the overall accuracy for building classification is almost 95%. Figure 6 shows the evaluation results for one of the 14 test images, namely P4-14. The individual evaluation results for P4-14 are shown in Table III.

**TABLE II:** The overall evaluation of the classification results for the 14 test images for which ground truth was not provided. The network performance statistics were computed by and provided by the ISPRS Working Group II/4 organizers as part of their urban classification benchmark. All shown values are percentages.

<table>
<thead>
<tr>
<th></th>
<th>pred./ref.</th>
<th>roads</th>
<th>bldgs</th>
<th>nat. gnd.</th>
<th>tree</th>
<th>car</th>
<th>artif. gnd.</th>
</tr>
</thead>
<tbody>
<tr>
<td>roads</td>
<td>89.9</td>
<td>1.6</td>
<td>4.9</td>
<td>2.9</td>
<td>0.1</td>
<td>0.6</td>
<td>0.3</td>
</tr>
<tr>
<td>bldgs</td>
<td>2.5</td>
<td>94.40</td>
<td>0.8</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0.3</td>
</tr>
<tr>
<td>nat. gnd</td>
<td>6.8</td>
<td>0.90</td>
<td>82.40</td>
<td>9.1</td>
<td>0</td>
<td>0</td>
<td>0.7</td>
</tr>
<tr>
<td>tree</td>
<td>7.6</td>
<td>1</td>
<td>20.3</td>
<td>70.6</td>
<td>0.4</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>artif. gnd.</td>
<td>17.60</td>
<td>1.6</td>
<td>0.8</td>
<td>1.80</td>
<td>76.7</td>
<td>11</td>
<td>11.1</td>
</tr>
</tbody>
</table>

**TABLE III:** Evaluation results for urban area P4-14. All shown values are percentages.

<table>
<thead>
<tr>
<th></th>
<th>pred./ref.</th>
<th>roads</th>
<th>bldgs</th>
<th>nat. gnd.</th>
<th>tree</th>
<th>car</th>
<th>artif. gnd.</th>
</tr>
</thead>
<tbody>
<tr>
<td>roads</td>
<td>89.44</td>
<td>1.17</td>
<td>6.04</td>
<td>1.82</td>
<td>0.11</td>
<td>1.42</td>
<td>1.42</td>
</tr>
<tr>
<td>bldgs</td>
<td>1.80</td>
<td>96.61</td>
<td>0.38</td>
<td>0.87</td>
<td>0.09</td>
<td>0.24</td>
<td>0.24</td>
</tr>
<tr>
<td>nat. gnd</td>
<td>17.64</td>
<td>1.10</td>
<td>70.3</td>
<td>9.82</td>
<td>0.04</td>
<td>1.11</td>
<td>1.11</td>
</tr>
<tr>
<td>tree</td>
<td>9.72</td>
<td>1.18</td>
<td>16.73</td>
<td>71.73</td>
<td>0.4</td>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>artif. gnd.</td>
<td>18.51</td>
<td>1.7</td>
<td>0.48</td>
<td>1.09</td>
<td>76.02</td>
<td>13.9</td>
<td>13.9</td>
</tr>
<tr>
<td>PRec./Corr.</td>
<td>64.15</td>
<td>6.8</td>
<td>11.96</td>
<td>2.24</td>
<td>0.49</td>
<td>14.36</td>
<td>14.36</td>
</tr>
<tr>
<td>Rec./Compl.</td>
<td>63.5</td>
<td>93.2</td>
<td>72.5</td>
<td>81.5</td>
<td>88.7</td>
<td>66.5</td>
<td>66.5</td>
</tr>
<tr>
<td>F1</td>
<td>74.2</td>
<td>94.90</td>
<td>71.40</td>
<td>76.3</td>
<td>81.90</td>
<td>23.60</td>
<td>23.60</td>
</tr>
</tbody>
</table>

VI. URBAN RECONSTRUCTION

The classification results are further processed. The depth map is used to extract boundary points and extrude 3D models to represent the geospatial objects in the scene. In particular the boundaries for the buildings are extracted and extruded to create polygonal models and generic objects such as cars are replaced by CAD models, and trees are replaced by procedural models. Figure 7(a)-(b) shows the generated labels being projected onto the same 3D models and Figure 7(c)-(d) shows textured models only for the classified buildings. Figure 8a shows a closeup of an urban area in which buildings have been replaced by polygonal models, cars by generic CAD models and trees by procedural models. The same scene with textures and from a different viewpoint is shown in Figure 8b.

**Fig. 6:** The evaluation result for one of the 14 test images. Resolution 6k × 6k. (a) Satellite image of the urban area P4-14. (b) Generated label map. (c) Red/green image, indicating wrongly classified pixels.

**Fig. 7:** (a) The resulting labels being projected onto the 3D models. (b) The satellite image projected onto the scene models. (c) Textured models showing only the classified buildings.

VII. CONCLUSION

We have presented a novel technique for multi-label pixel-wide classification for reconstruction of large-scale urban areas. Unlike existing methods, the proposed method relies on a relatively small CNN to efficiently process large sets of data. An empirical study was performed and presented to determine the parameters for which the network produces optimal results. Scale invariance is incorporated in the processing by training the network on composite images containing multiple scales of the originals. This results in multiple per-pixel classification...
labels which are mapped into a single label using a trained linear classifier. Pixels lying on image boundaries are not assigned any label. We reformulate the problem as a labeling problem and propose the use of graph-cuts for maximum-a-posteriori inference of those labels with Markov Random Field priors. The result is a dense set of labels where all pixels in the image are assigned a label according to the minimized energy function. The proposed technique has been extensively tested on large-scale datasets depicting urban areas for which ground truth is available. The achieved accuracy in the classification ranges in the 90th percentile.

ACKNOWLEDGMENT

The authors would like to thank the International Society for Photogrammetry and Remote Sensing and in particular the ISPRS Working Group II/4 organizers and Dr. Markus Gerke for creating the benchmark and making it publicly available. We also thank Microsoft Research for making processing large datasets possible through its Azure for Research Award. This research is based upon work supported by NSERC DG under Grant no N01670.

REFERENCES


Fig. 9: Gray cells: Kernel vs patch size is too small. The resulting image size after the two convolutional layers is too small and cannot be processed further without changing the network architecture.
Naive Bayes Classification for Subset Selection in a Multi-label Setting

Luca Mossina and Emmanuel Rachelson
Department of Complex Systems Engineering
ISAE-SUPAERO, Université de Toulouse
Toulouse, France
luca.mossina@isae-supraero.fr, emmanuel.rachelson@isae-supraero.fr

Abstract—This article introduces a novel probabilistic formulation of multi-label classification based on the Bayes theorem. Under the naive hypothesis of conditional independence of features given the labels, a pseudo-bayesian inference approach is adopted, known as Naive Bayes. The prediction consists of two steps: the estimation of the size of the target label set and the selection of the elements of this set. This approach is implemented in the NaiBX algorithm, an extension of naive Bayes into the multi-label domain. Its properties are discussed and evaluated on real-world data.

Index Terms—Naive Bayes, multi-label classification, subset selection.

I. INTRODUCTION

Multi-label classification (MLC) aims at predicting a set of one or more labels \( y = \{y^1, \ldots, y^m \mid y^i \in \mathcal{L} \} \), as a function of some input \( x \). Applications cover a diverse range of fields such as text categorization [7], where more than a topic can be associated to a law, or gene function analysis [5], where a gene can be responsible for multiple mechanisms (transcription, cell growth and division, metabolism etc.).

NaiBX, our proposed algorithm, works in two steps. First, it learns to predict the number \( m \) of labels \( y^i \in \mathcal{L} \) to be included in the subset for a given input \( x \). After that, it proceeds predicting the first label \( y^{(1)} \) given \( x \) and \( m \), the second label \( y^{(2)} \) given \( x, m \) and \( y^{(1)} \), and so on until completion. At the end of the procedure, one will have followed the inference sequence \( m \rightarrow y^{(1)} \rightarrow y^{(2)} \rightarrow \ldots \rightarrow y^{(k)} \rightarrow \ldots \rightarrow y^{(m)} \). For this purpose, we construct a cascade of naive Bayes classifiers (NBC), where predictor number \( k \) predicts the \( k \)-th element in the subset. Opting for a NBC means assuming conditional independence of features given the labels. This hypothesis induces a dramatic simplification of the predictors computation’s complexity (both in time and space), and the overall learning task boils down to training the elementary parameters of a single NBC that can be used for prediction of every element in the subset.

We first introduce the state-of-the-art approaches to MLC in Section II. We recall the principle and properties of Naïve Bayes classification in Section III-A. In Section III-B, we introduce a general “cascade of predictors” approach to multi-label classification and propose a Naïve Bayes algorithm as the base classifier. We derive a learning algorithm called NaiBX and discuss its properties in Section III-D. Experimental results and comparisons are presented in Section IV. Finally, we summarize and conclude in Section V.

II. MULTI-LABEL CLASSIFICATION

Our method addresses the problem of MLC [8][20]: given a discrete set \( \mathcal{L} \) of options (“labels”) and an arbitrary feature space \( \mathcal{X} \), one will look for the most appropriate subset of those options via the classifier \( h : \mathcal{X} \rightarrow \mathcal{P}(\mathcal{L}) \).

If the set of available labels is \( \mathcal{L} = \{1, 2, \ldots, L\} \), a target subset of labels \( \mathbf{y} \) can be, for example, \( \mathbf{y} = \{3, 5, 6, 7, 9, \ldots, L\} \equiv [0, 0, 1, 0, 1, 1, 0, 1, \ldots, 1] \in \{0, 1\}^L \). The latter is often referred to as one-hot-encoding.

Existing binary or multi-class classification algorithms have been adapted to the multi-label case, a process known as algorithm adaption. ML-kNN [19] and BPMLL [18] are, for example, ML versions of \( k \)-nearest neighbours and back-propagation. We find, however, that the most efficient approaches are obtained via problem transformation, where the multi-label instance is decomposed into simpler binary of multiclass classification problems; the two main paradigms are the Label Powerset (LP) and the Binary Relevance (BR) [13] methods. The first consists in turning a multi-label problem into a multi-class one, mapping directly an element of the feature space to any of the elements in \( \mathcal{P}(\mathcal{L}) \). As the number of classes grows exponentially with the number of labels, LP cannot deal with big sets of labels. The second, BR, consists in independently training a binary classifier for each admissible label \( l_j \in \mathcal{L} \), obtaining as many classifiers as there are labels.

The \( RA\)andom \( k\)-LaBel \( \mathcal{E} \)lts \( (RA\mathcal{E}L) \) algorithm [14], is a particularly interesting and effective variation of LP, where one trains \( m \) models whose targets are a subset of \( k \) labels from those available. That is, instead of having all the elements in \( \mathcal{P}(\mathcal{L}) \) as targets, a set of arbitrarily sized label clusters are employed. This makes problems of up to a few dozens of labels manageable.

Stemming from the principles of BR, [10] extended the method by taking into account information about label interdependence. Incrementally, at each step, what was previously predicted is taken into account. Their Classifier Chains (CC) meta-algorithm first starts by predicting whether label \( l_1 \) is to be included in the target vector. It then continues predicting the presence of the second label \( l_2 \) given the information contained in the data and whether \( l_1 \) was included
in the target vector \((\hat{Y}_i \in \{0,1\})\). At the \(i\)-th step we have \(\hat{Y}_i = h(X, \hat{Y}_{i-1}, Y_{i-2}, ..., \hat{Y}_1)\). The order of evaluation of the binary labels \(Y_i \in \{0,1\}\) can affect negatively the performance of the algorithm. The Ensemble of Classifier Chains (ECC) [11] extends CC accounting for this limitation: \(m\) different predictors are trained on random permutations of the labels, like for example: \(h^1(Y_1, Y_4, Y_3, Y_2, \ldots), h^2(Y_3, Y_1, Y_2, Y_4, \ldots), \ldots, h^m(Y_4, Y_3, Y_2, Y_1, \ldots)\). Then, one applies a bagging [1] step for the selection, via a threshold function, of the best labels. In [4] this method was refined via the Probabilistic Classifier Chains (PCC). At each level of the chain they compute a joint distribution, which is the cause of higher computational costs. Because of this, recent advances in CC go towards approximate methods based on Monte Carlo sampling [3] and tree search[2].

III. Naive Bayes for Subset Selection

Let \(X = (X_1, \ldots, X_n)\) be the random vector of observations \(x\), taking its values in \(X\), and let \(C\) be the random variable describing the class associated to \(X\) in a classification problem. A probabilistic classifier \(f\) assigns the class \(c\) to a new observation \(x\) if \(c\) maximizes the conditional probability of \(C = c\) given that \(X = x\).

A. Naive Bayes Classification

As estimating a multivariate conditional distribution can be rather challenging, the Naive Bayes simplification is widely adopted [6] consisting in supposing that, \(\forall (i, j) \in [1, n]^2\),

\[
\mathbb{P}(X_i \mid C, X_j) = \mathbb{P}(X_i \mid C).
\]

While yielding poor probability estimations, the classifications inferred are of good quality [16]. A Naive Bayes Classifier is then deduced as

\[
f_{\text{NBC}}(x) = \arg\max_{c \in C} \mathbb{P}(C = c \mid X = x) = \arg\max_{c \in C} \frac{\mathbb{P}(C = c) \mathbb{P}(X = x \mid C = c)}{\mathbb{P}(X = x)} = \arg\max_{c \in C} \mathbb{P}(C = c) \mathbb{P}(X = x \mid C = c) = \arg\max_{c \in C} \left[ \mathbb{P}(C = c) \prod_{i=1}^{n} \mathbb{P}(X_i = x_i \mid C = c) \right],
\]

as the denominator does not depend on \(c \in C\).

B. Cascade of Predictors

In MLC we look for a collection of labels \(Y \subset \mathcal{L}\). An intuitive way to proceed is to consider that selecting a given-size subset consists in choosing a first element in \(\mathcal{L}\), then a second given the first, then a third given the first and second, and so on until one reaches the appropriate subset size. Selecting a subset of \(\mathcal{L}\) can be done by choosing an ordered sequence of values of \(\mathcal{L}\) if our selection function at each step effectively re-creates the correct unordered subset. This approach differs from classifier chains since it does not predict in sequence whether the \(|\mathcal{L}|\) labels belong or not to the target, but rather picks them incrementally. Notably, the cascade architecture does not rely on an a priori ordering of the labels.

Let \(Y\) be the random variable describing the subset of \(\mathcal{L}\) that should be associated to \(x\). The target of the classification algorithm is to learn the correct mapping from \(x\) to realizations of \(Y\). We write \(\bar{y}\) such realizations of \(Y\) to avoid confusion with vectors \(y\) of values of \(\mathcal{L}\). Then the classifier \(f\) we are searching for is

\[
f(x) = \arg\max_{\bar{y} \in \mathcal{P}(\mathcal{L})} \mathbb{P}(Y = \bar{y} \mid X = x).
\]

In order to sequentially select the elements of the optimal \(\bar{y}\), we want to decompose the probability of Equation 2 into elementary probabilities related to each element \(y_i\) of \(\bar{y}\). Such elementary probabilities are related to the random event \(`y_i \in Y\)’. Let \(M\) be the random variable describing the size of \(Y\). Then a subset \(\bar{y}\) is composed of elements \(y_1, \ldots, y_k\), where \(M\) takes the value \(k\). Given \(\bar{y} = \{y_1, \ldots, y_k\} \subset \mathcal{L}\), the following statements hold:

\[
\begin{align*}
“y_1 \in Y” & \land \ldots \land “y_k \in Y” \ni \bar{y} \subset Y \quad (3) \\
“y_1 \in Y” & \land \ldots \land “y_k \in Y” \land “\text{all others } \not\in Y” \ni \bar{y} = Y \quad (4) \\
“y_1 \in Y” & \land \ldots \land “y_k \in Y” \land “M = k” \ni \bar{y} = Y \quad (5)
\end{align*}
\]

Equation 3 expresses the fact that individual properties on the \(y_i\) can help characterize the probability that a given subset \(\bar{y}\) is included in \(Y\). Equation 4 helps expressing that \(Y\) is precisely equal to such a subset \(\bar{y}\). Its formulation is equivalent to the target of CC algorithms. Finally, Equation 5 is of particular interest to us since it states that the subset that is both included in \(Y\) and has the same size as \(Y\) is precisely equal to \(Y\). For any sequence of values \(y_1, \ldots, y_i \subset \mathcal{L}\), we introduce the notation

\[
p(y_i \mid X, M, y_1, \ldots, y_{i-1}) = \mathbb{P}(y_i \in Y \mid X = x, M = m, y_1, \ldots, y_{i-1} \in Y).
\]

We then use Equation 5 to decompose the probability estimate of the probabilistic classifier in Equation 2, using the chain rule

\[
\mathbb{P}(Y = \bar{y} \mid X = x) = \mathbb{P}(M = m \mid X = x) \times \mathbb{P}(\bar{y} \subset Y \mid X = x, M = m) = \mathbb{P}(M = m \mid X = x) \times p(y_1 \mid X, M, y_1) \times p(y_2 \mid X, M, y_1, y_2) \times \cdots \times p(y_k \mid X, M, y_1, \ldots, y_k).
\]

Copyright © 2018 by CENPARMI, Concordia University
So, writing \( s(x) = \max_{y \in P(\mathcal{L})} \mathbb{P}(Y = \hat{y} \mid X = x) \), from Equation 2 we derive

\[
s(x) = \max_{y \in P(\mathcal{L})} \left( \mathbb{P}(M = m \mid X = x) \times \prod_{i=1}^{m} p(y_i \mid x, m, y_1, \ldots, y_{i-1}) \right).
\]

In [4] it is argued that the Bayes optimal classifier solves the maximization problem in Equation 2 to optimality. The CC approach, however, exploits Equation 4 and adopts a greedy search heuristic consisting in incrementally picking the most (marginally) probable labels in a predefined (artificial) order. Our cascade architecture somehow falls in between these two extremes. It adopts a greedy, possibly sub-optimal search method that incrementally picks labels in the label set, but does not rely on any predefined ordering of the labels. The cascade architecture searches for a solution to the maximization problem of Equation 2 by computing the heuristic score function

\[
s(x) = \max_{y \in \mathcal{L}} \left[ p(y_m \mid x, m, y_1, \ldots, y_{m-1}) \times \right.
\]
\[
\max_{y_{m-1} \in \mathcal{L}} \left[ p(y_{m-1} \mid x, m, y_1, \ldots, y_{m-2}) \times \right.
\]
\[
\ldots \times \max_{y_1 \in \mathcal{L}} \left[ p(y_1 \mid x, m) \times \right. \right.
\]
\[
\left. \max_{y \in \mathcal{L}} \mathbb{P}(M = m \mid X = x) \right] \right].
\]

Each of the \( m + 1 \) probability estimators in the product above is a classifier in itself. The feature space of \( p(y_k \mid x, m, y_1, \ldots, y_{k-1}) \) is \( \mathcal{X} \times \mathbb{N} \times \mathcal{L}^{k-1} \). We call such a structure a cascade of predictors. The cascade structure unfolds seamlessly from the application of the chain rule (see Figure 1). In a cascade, one predicts the number of elements in the subset, then the first value of the subset, then the second using the results from the computation of the first, etc.

C. Cascade of NBCs

Any efficient classification algorithm can be used to predict each level in the cascade. This implies storing in memory \(|\mathcal{L}| + 1\) classifiers having increasingly complex feature spaces and predicting values in a class set of size \(|\mathcal{L}|\), which may not scale up to large label sets. Furthermore, the feature spaces of the last predictors in the cascade are complex, requiring powerful learning architectures, lots of data and possibly very long training times. Taking NBCs as base classifiers for each level in the cascade induces a dramatic simplification of both training and storage of the multi-label classifier. Let us suppose that each of these estimators is built upon the Naïve Bayes assumption. Based on the conclusions of [16][17], although the probability estimates of these classifiers are poor, at each step of the cascade the computed \( \arg \max \) remains close to optimal. Eventually, we are left with \(|\mathcal{L}| + 1\) NBCs: one for the subset size prediction and one for each level in the cascade. If we start the numbering at zero, predictor zero estimates \( p(y_1 \mid x, m) \), predictor two estimates \( p(y_2 \mid x, m, y_1) \) and so on.

Let \( f_k \) be the selection function of predictor number \( k \). Since it is a Naïve Bayes classifier, according to Equation 1, its selection function decomposes as

\[
f_k(x, m, y_1, \ldots, y_{k-1}) = \arg \max_{y_k \in \mathcal{L}} \mathbb{P}(y_k \in Y) \times \mathbb{P}(M = m \mid y_k \in Y)
\]
\[
\times \prod_{i=1}^{m} p(X_i = x_i \mid y_k \in Y) \times \prod_{j=1}^{k-1} \mathbb{P}(y_j \in Y \mid y_k \in Y)
\]

Computing the selection function \( f_k \) requires evaluating each factor in Equation 7, which are univariate probability estimators. These are not specific to the \( k \)-th step in the cascade: take two predictors \( f_k \) and \( f_{k'} \), both will make use of the same generic estimators \( \mathbb{P}(y \in Y) \), \( p(X_i \mid y \in Y) \), \( \mathbb{P}(M \mid y \in Y) \) and \( \mathbb{P}(y' \in Y \mid y \in Y) \). The same univariate probability estimators are simply combined in different fashions at the different stages of the cascade.

The cases of \( f_1 \) and \( f_0 \) require different computations. Recall that \( f_1(x, m) \) is the selection function of the first label. Its computation makes use of the same \( \mathbb{P}(y \in Y) \), \( p(X_1 \mid y \in Y) \) and \( \mathbb{P}(M \mid y \in Y) \) probability estimators as the rest of the cascade (simply it does not use the \( \mathbb{P}(y' \in Y \mid y \in Y) \) estimator). Finally, \( f_0(x) \) selects the most probable subset size associated to \( x \) via the relation

\[
\begin{align*}
\mathbb{P}(M \mid X) & \propto \mathbb{P}(M) \times \mathbb{P}(X \mid M),
\end{align*}
\]

yielding another Naïve Bayes Classifier.

Table I summarizes the 6 univariate distributions that are required for the computation of all levels in the cascade, along with the space complexity of their storage (detailed in section...
add_example
add_example
the
a generic updating step of the parameters of distributions in
features [9, Chapter 6], we learn just a probability
κ
experiments, as we assume normality, we store the mean and
for
of the distributions
I). In case of continuous features, we estimate the parameters
predict_subset
predict_subset
D. The NaiBX Algorithm
NaiBX is an online algorithm that combines a training
function add_example and a prediction function predict_subset. The add_example procedure
computes the statistics describing the 6 probability distributions
required by NaiBX for future predictions (presented in Table I).
In case of continuous features, we estimate the parameters
of the distributions \(X_i | M\) and \(X_i | Y\). In our numerical
experiments, as we assume normality, we store the mean and
and variance of each distribution, yielding \(\kappa = 2\). In case of bag-
of-words features [9, Chapter 6], we learn just a probability
parameter, thus \(\kappa = 1\) (see (D2) and (D5) in Table I).
Algorithm 1 presents the incremental learning process of the
add_example function, where update_parameters is a
generic updating step of the parameters of distributions in
Table I. For (D2) and (D5) we update the mean and variance,
for (D1), (D3), (D4) and (D6) we update their univariate
probability estimates. Algorithm 2 presents the operations
performed when a new sample \(x\) requires the prediction
of the associated subset of labels. The predict_subset function
receives a new observation \(x_{\text{new}}\) as an input and
predicts a vector of labels \(y_{\text{pred}}\) in a two-step process.
In the first one, it estimates the size of the target vector via
the predict_size function. In the second step it proceeds by
estimating the elements of the vector through the cascade of
predictors. At each iteration the function predict_label is
called and fed as an input the size \(\hat{m}\) and the labels
estimated so far. Note that NaiBX was thought as the natural
extension of Naive Bayes Classifiers to the multi-label case.
If one trains NaiBX on a data set with targets \(y_{\text{obs}}\) of size
\(m = 1\) with values from a target set of size \(|\mathcal{L}| = 2\), that is
\(f_{\text{classifier}} : \mathcal{X} \rightarrow \mathcal{L} = \{c_1, c_2\}\), then NaiBX will act
as a traditional binary Naive Bayes Classifier. Furthermore,
allowing \(|\mathcal{L}| > 2\) will return a multi-class classifier.

E. Complexity Analysis
Storing the cascade of predictors during the training phase
boils down to storing the parameters of the six probability
distributions presented in Table I. The space complexity of
storing these parameters are recalled in the above tables. The
space requirement for the whole cascade of classifiers is in
III-E). Finally, the overall space requirements for the whole
cascade is \(\mathcal{O}(n\kappa + |\mathcal{L}|)|\mathcal{L}|\), \(\kappa\) being the number of parameters
describing a distribution \(p(X_i | \cdot)\).

\[
\begin{array}{|c|c|}
\hline
\text{Distribution} & \text{Space complexity} \\
\hline
(D1) \ P(M) & \mathcal{O}(|\mathcal{L}|) \\
(D2) \ p(X_i | M) & \mathcal{O}(n\kappa|\mathcal{L}|) \\
(D3) \ P(y \in Y) & \mathcal{O}(|\mathcal{L}|) \\
(D4) \ P(M | y \in Y) & \mathcal{O}(|\mathcal{L}|^2) \\
(D5) \ p(X_i | y \in Y) & \mathcal{O}(n\kappa|\mathcal{L}|) \\
(D6) \ P(y' \in Y | y \in Y) & \mathcal{O}(|\mathcal{L}|^2) \\
\hline
\end{array}
\]

Table I
UNIVARIATE DISTRIBUTIONS IN A CASCADE OF NBCS

Algorithm 1: NaiBX—Generic Learning Step

\[
\begin{align*}
\text{add_example}(x, y) \\
m &= \text{length}(y) \\
\text{update_parameters}(P(M)) \\
&\text{for } x_i \text{ in } x \text{ do} \\
&\quad \text{update_parameters}(p(X_i | M)) \\
&\text{for label } y \text{ in } y \text{ do} \\
&\quad \text{update_parameters}(P(y)) \\
&\quad \text{update_parameters}(P(M | y)) \\
&\text{for } x_i \text{ in } x \text{ do} \\
&\quad \text{update_parameters}(p(X_i | y)) \\
&\text{for each label } y' \in y, y' \neq y \text{ do} \\
&\quad \text{update_parameters}(P(y | y')) \\
&y \leftarrow y \setminus y
\end{align*}
\]

Algorithm 2: NaiBX, Prediction Step

\[
\begin{align*}
predict_subset(x_{\text{new}}): \\
\hat{m} &\leftarrow \text{pred_size}(x_{\text{new}}) \\
y_{\text{pred}} &\leftarrow \emptyset \\
&\text{while length}(y_{\text{pred}}) \leq \hat{m} \text{ do} \\
&\quad y_{\text{pred}} \leftarrow y_{\text{pred}} \cup \text{pred_label}(x_{\text{new}}, \hat{m}, y_{\text{pred}}) \\
&\text{return } y_{\text{pred}} \\
pred_size(x_{\text{new}}): \\
\hat{m} &\leftarrow \arg\max_{d \in \{0, 1, ..., L\}} \mathbb{P}(m_d) \times \prod_{i=1}^{n} \mathbb{P}(X_i | m_d) \\
&\text{return } \hat{m} \\
pred_label(x_{\text{new}}, \hat{m}, \{y_1, y_2, ..., y_i\}): \\
y_{i+1} &\leftarrow \arg\max_{y_{i+1} \in \mathcal{L}} \mathbb{P}(y_{i+1}) \times \mathbb{P}(\hat{m} | y_{i+1}) \\
&\times \prod_{i=1}^{n} \mathbb{P}(x_i | y_{i+1}) \\
&\times \prod_{j=1}^{n} \mathbb{P}(y_j | y_{i+1}) \\
&\text{return } \{y_1, y_2, ..., y_i\} \cup \{y_{i+1}\}
\end{align*}
\]

\(x_{\text{new}}\) is the features vector, \(n\) is the number of features.

The time complexity of the training and prediction phases
unfolds straightforwardly from the presentation in Algorithms
1 and 2. These remarks are summarized in Table II.

\[
\begin{array}{|c|c|c|}
\hline
\text{Training} & \text{Prediction} \\
\text{Time} & \mathcal{O}
\left(|\mathcal{L}|(\kappa n + |\mathcal{L}|)\right) & \mathcal{O}
\left(|\mathcal{L}|^2(n + |\mathcal{L}|)\right) \\
\text{Space} & \mathcal{O}
\left(|\mathcal{L}|\right) & \mathcal{O}
\left(|\mathcal{L}|^2\right) \\
\hline
\end{array}
\]

Table II
NAIBX—TIME AND SPACE COMPLEXITY

Copyright © 2018 by CENPARMI, Concordia University
We also report the Accuracy, Precision and Recall [13] of our sense of the performance of the algorithm. Let us denote by \( \hat{y} \) the predicted target vectors, respectively the observed and predicted target vectors of the algorithms’ complexity on performances. For large target sets, predicting the correct one is a nontrivial task even if some models can perform better than others given a specific metric.

A. Evaluation Metrics

We adopt a fairly standard pool of metrics [13] to get a sense of the performance of the algorithm. Let us denote by \( \hat{y} \) and \( \hat{y} \) respectively the observed and predicted target vectors for the \( i \)-th entry in a collection of data. The label cardinality, given by \( LCard = \sum_{i=1}^{N} |y_i| \), allows to quantify the multi-labelness of data, yielding the average number of labels per target across the dataset. The label density \( LDens = \frac{LCard}{N} \times 100 = \frac{1}{N} \sum_{i=1}^{N} |y_i| \times 100 \), expresses what proportion of the available labels are, on average, associated to a data point \( x \).

Let \( \sum_{i=1}^{N} \sum_{k=1}^{L} \mathbbm{1}(y_i = \hat{y}_k) \) take value one if the two vectors are exactly equivalent and zero otherwise. Then the zero-one loss metric \( L_0 \) and complementary zero-one score \( ZS \) are given by \( L_0 = 1 - ZS = 1 - \sum_{i=1}^{N} \sum_{k=1}^{L} \mathbbm{1}(y_i = \hat{y}_k) \) respectively. For large target vectors, \( ZS \) becomes less meaningful as even a single mistake will invalidate an otherwise good prediction. More forgiving is the Hamming Loss (with its complement, the Hamming score), which will invalidate an otherwise good prediction. More forgiving is the Hamming Loss (with its complement, the Hamming score), which invalidates an otherwise good prediction. More forgiving is the Hamming Loss (with its complement, the Hamming score), which invalidates an otherwise good prediction.

B. Methods

Among the options available, ECC seems to be one of the most efficient variations on BR, while RAkEL is among the most interesting variations on LP. For both cases, we opted for a Support Vector Machine [6] as base classifier as reported in the literature.

We are interested in methods granting running times in the order of seconds or minutes. Nonetheless, for the sake of comparing predictive performances, we allowed running times (including handling data, training and testing) of up to 12 hours with 4 GB of memory reserved to the task. When some method failed to deliver a result, its corresponding line was left blank. For NaiBX we ran our implementation while not using default parameters.

### IV. Experimental Results

The computational experiments were carried out on data from a set of standard MLC data sets commonly adopted in the literature [15][12]. In Table III are the data used in the experiments, including sets of continuous features and sets of binary bag-of-words encoding.

#### A. Evaluation Metrics

<table>
<thead>
<tr>
<th>Data</th>
<th>( N )</th>
<th>( \text{dim}(X) )</th>
<th>Labels</th>
<th>( LCard )</th>
<th>( LDens )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Continuous Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAL500</td>
<td>502</td>
<td>68</td>
<td>174</td>
<td>26,043</td>
<td>15.0 %</td>
</tr>
<tr>
<td>Emotions</td>
<td>593</td>
<td>72</td>
<td>6</td>
<td>1,869</td>
<td>31.1 %</td>
</tr>
<tr>
<td>Mediummill</td>
<td>43902</td>
<td>120</td>
<td>101</td>
<td>4,376</td>
<td>4.3 %</td>
</tr>
<tr>
<td>NUS-WIDE</td>
<td>269648</td>
<td>128</td>
<td>81</td>
<td>1,873</td>
<td>2.3 %</td>
</tr>
<tr>
<td>Scene</td>
<td>2407</td>
<td>294</td>
<td>6</td>
<td>1,074</td>
<td>17.9 %</td>
</tr>
<tr>
<td>Yeast</td>
<td>2417</td>
<td>103</td>
<td>14</td>
<td>4,237</td>
<td>30.2 %</td>
</tr>
<tr>
<td>(2) Bag-of-Words Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bibtex</td>
<td>7935</td>
<td>1836</td>
<td>159</td>
<td>2,402</td>
<td>1.5 %</td>
</tr>
<tr>
<td>Eron</td>
<td>1702</td>
<td>1001</td>
<td>53</td>
<td>3,378</td>
<td>6.4 %</td>
</tr>
<tr>
<td>LLog</td>
<td>1460</td>
<td>1004</td>
<td>75</td>
<td>1,180</td>
<td>1.6 %</td>
</tr>
<tr>
<td>Slashdot</td>
<td>3782</td>
<td>1079</td>
<td>22</td>
<td>1,181</td>
<td>5.4 %</td>
</tr>
<tr>
<td><strong>TABLE III</strong></td>
<td><strong>Experiments Data Sets</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

None of the aforementioned metrics can be considered flawless. In general, measuring the performance in MLC can be a problem in itself, as some models can perform better than others given a specific metric.

#### B. Methods

Among the options available, ECC seems to be one of the most efficient variations on BR, while RAkEL is among the most interesting variations on LP. For both cases, we opted for a Support Vector Machine [6] as base classifier as reported in the literature.

We are interested in methods granting running times in the order of seconds or minutes. Nonetheless, for the sake of comparing predictive performances, we allowed running times (including handling data, training and testing) of up to 12 hours with 4 GB of memory reserved to the task. When some method failed to deliver a result, its corresponding line was left blank. For NaiBX we ran our implementation while not using default parameters.

### TABLE IV

<table>
<thead>
<tr>
<th>Data</th>
<th>ALGO</th>
<th>( \Delta LCard )</th>
<th>Hpx</th>
<th>Zpx</th>
<th>Acc</th>
<th>Pre</th>
<th>Rec</th>
<th>Time [s]</th>
<th>Train</th>
<th>Pred</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Continuous Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAL500</td>
<td>ECC</td>
<td>-12.072</td>
<td>0.000</td>
<td>0.226</td>
<td>0.511</td>
<td>0.277</td>
<td>29.265</td>
<td>3.427</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NaiBX</td>
<td>RAkEL</td>
<td>-16.042</td>
<td>0.863</td>
<td>0.000</td>
<td>0.209</td>
<td>0.610</td>
<td>0.235</td>
<td>24.480</td>
<td>0.142</td>
<td></td>
</tr>
<tr>
<td>Yeast</td>
<td>NaiBX</td>
<td>-0.325</td>
<td>0.817</td>
<td>0.002</td>
<td>0.249</td>
<td>0.379</td>
<td>0.372</td>
<td>0.031</td>
<td>6.235</td>
<td></td>
</tr>
<tr>
<td>(2) Bag-of-Words Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bibtex</td>
<td>ECC</td>
<td>-0.066</td>
<td>0.806</td>
<td>0.320</td>
<td>0.578</td>
<td>0.682</td>
<td>0.786</td>
<td>0.695</td>
<td>0.006</td>
<td></td>
</tr>
<tr>
<td>Eron</td>
<td>RAkEL</td>
<td>0.377</td>
<td>0.795</td>
<td>0.317</td>
<td>0.592</td>
<td>0.642</td>
<td>0.774</td>
<td>0.759</td>
<td>0.003</td>
<td></td>
</tr>
<tr>
<td>NaiBX</td>
<td>RAkEL</td>
<td>-0.079</td>
<td>0.771</td>
<td>0.275</td>
<td>0.528</td>
<td>0.639</td>
<td>0.648</td>
<td>0.600</td>
<td>0.017</td>
<td></td>
</tr>
<tr>
<td>Yeast</td>
<td>NaiBX</td>
<td>-12 hrs</td>
<td>-12 hrs</td>
<td>116.427</td>
<td>2.351</td>
<td>30.826</td>
<td>0.138</td>
<td>12.484</td>
<td>0.020</td>
<td></td>
</tr>
<tr>
<td>(CV 66%)</td>
<td>RAkEL</td>
<td>-0.288</td>
<td>0.799</td>
<td>0.203</td>
<td>0.556</td>
<td>0.680</td>
<td>0.653</td>
<td>0.305</td>
<td>0.629</td>
<td></td>
</tr>
</tbody>
</table>

† \( \Delta LCard = LCard - LCard \), the smaller the value the better.
‡ Most of the predictions were empty sets, losses yield no valuable meaning.

### Metrics—Experimental Results

The computational experiments were carried out on data from a set of standard MLC data sets commonly adopted in the literature [15][12]. In Table III are the data used in the experiments1, including sets of continuous features and sets of binary bag-of-words encoding.

None of the aforementioned metrics can be considered flawless. In general, measuring the performance in MLC can be a problem in itself, as some models can perform better than others given a specific metric.

#### B. Methods

Among the options available, ECC seems to be one of the most efficient variations on BR, while RAkEL is among the most interesting variations on LP. For both cases, we opted for a Support Vector Machine [6] as base classifier as reported in the literature.

We are interested in methods granting running times in the order of seconds or minutes. Nonetheless, for the sake of comparing predictive performances, we allowed running times (including handling data, training and testing) of up to 12 hours with 4 GB of memory reserved to the task. When some method failed to deliver a result, its corresponding line was left blank. For NaiBX we ran our implementation while not using default parameters.
CAL500 $NBX_{trueM}$ -0.325 0.863 0.002 0.249 0.610 0.372
Emotions $NBX_{trueM}$ 0.066 0.806 0.320 0.592 0.682 0.771
Mediamill $NBX_{trueM}$ -0.236 0.968 0.186 0.164 0.223 0.201
Scene $NBX_{trueM}$ -0.061 0.906 0.642 0.694 0.752 0.816
Yeast $NBX_{trueM}$ 0.007 0.799 0.203 0.541 0.680 0.635

<table>
<thead>
<tr>
<th>Data</th>
<th>ALGO $NBX_{trueM}$</th>
<th>$\Delta$Card</th>
<th>$H_s$</th>
<th>$Z_s$</th>
<th>Acc</th>
<th>Pre</th>
<th>Rec</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(1) Continuous Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CAL500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Emotions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mediamill</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scene</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Yeast</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2) Bag-of-Words Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bibex</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Eronn</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LLog</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SlashDot</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$NBX_{trueM}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE V METRICS—$NBX_{trueM}$ VS THE BEST ALTERNATIVE.

C. Results

The good overall performance of NaiBX is very close on average to its best competitor for $H_s$, $Z_s$ and Acc, when it is not the best itself. If time is not a determinant factor, then highly engineered algorithms can be employed, otherwise a compromise has to be made. If a trade-off between rapidity of learning/prediction and quality of estimations has to be taken into account, NaiBX, with its simplicity and agile structure, is an interesting option worth considering. In particular, NaiBX will naturally scale up to large label sets, where methods derived from BR that need training a model for each label might not be exploitable.

D. Prediction of the Target Size

The prediction of the target size $m$ is the peculiar feature of NaiBX and we think that it deserves some attention. We ran an experiment supposing the size of the target was known and reported the results as $NBX_{trueM}$, skipping the size prediction phase. As reported in Table V, on most of the instances $NBX_{trueM}$ is as good as or better than the best performing algorithm on all metrics, prompting great interest for further work on the specific topic of size estimation. Estimating $m$ with more refined methods at the cost of increased computational time is an option worth exploring.

V. Conclusion

The proposed algorithm showed significant advantages in terms of computation costs and proved to be competitive in terms of predictive performance, thus offering a viable alternative for tasks requiring a more agile computational footprint. Our approach allows to see the problem from a different perspective than the current literature, notably thanks to the prediction of the target size ($m$) independently from the prediction of the labels. In future research we will further address this aspect, not excluding the possibility of mixing different prediction paradigms for the two tasks.

Overall, we introduced NaiBX, a computationally light and efficient multi-label classification method, that proved to be both scalable to large and complex data sets and competitive with state-of-the-art algorithms in terms of predictive performance. This opens up new perspectives for its application on large scale, real-world data.

ACKNOWLEDGEMENTS

This research benefited from the support of the “FMJH Program Gaspard Monge in optimization and operation research”, and from the support to this program from EDF.

REFERENCES

SEMISUPERVISED PRODUCT QUANTIZATION FOR APPROXIMATE NEAREST NEIGHBOR SEARCH

Xiao Bai¹, Yuan Tian¹, Jun Zhou²

¹School of Computer Science and Engineering, Beihang University, Beijing, China
²School of Information and Communication Technology, Griffith University, Nathan, Australia

ABSTRACT

Approximate nearest neighbor search via product quantization is a promising way to achieve fast retrieval on large scale datasets. Though exhibiting great flexibility in real-world applications, product quantization has yet to be comprehensively studied especially when prior knowledge is presented. In this paper, we introduce a semi-supervised image retrieval method based on product quantization. In this method, constraint rules based on prior knowledge are defined for codebook generation to guarantee the semantic similarity of data. We iteratively optimize the codebook generation and data decomposition steps by fixing the other. Experimental results on three datasets validate the superiority of our method over several baseline approaches.

1. INTRODUCTION

Nearest Neighbor (NN) search is an important method in information retrieval and computer vision. Given a query, NN search aims at finding its nearest neighbor in the target dataset. However, the retrieval efficiency is always a big concern on NN search as its high complexity makes it difficult to scale up to large dataset. This leads to the development of approximate nearest neighbors (ANN) methods, such as hashing [1, 2, 3, 4] and Vector Quantization (VQ) [5, 6], which find approximate results instead of exact matches.

Vector quantization is a technique to encode vectorized image representation into compact codes using a codebook and data partition which are generated from the original vector space. In order to minimize the mean distortion, the objective of VQ models is minimizing the differences between sample distances in the original space and in the transformed space. Product Quantization (PQ) [6, 7] is an important approach in VQ. Compared with hashing, product quantization has higher retrieval accuracy when samples have the same code length. The idea of PQ is to divide the original space into a Cartesian product of low-dimensional subspaces and then quantize each subspace separately. The distance between a query and a target sample approximates the distance between their encoded form, i.e., as sum of the distances between the sub-vector of the query and the quantized centers in each subspace. A distance lookup table can be constructed for fast distance evaluation.

The retrieval performance of PQ methods has been boosted thanks to a variety of approaches, such as Optimized Product Quantization [7], Cartesian k-means(CKM) [8], Locally Optimized Product Quantization [9] and Composite Quantization (CQ) [10]. Compared with these unsupervised methods, supervised algorithms normally achieve higher accuracy [11, 4, 12]. However, prior knowledge, mostly presented in human labels, is usually limited in many real world applications, which is insufficient to train a good model. On the other hand, there may be a lot of unlabelled data. It is therefore intuitive to take advantage of limited prior knowledge and expand it to the unlabelled data to improve the retrieval performance. Motivated by such needs, we propose a novel semi-supervised product quantization method in this paper.

The goal of the semi-supervised product quantization method is to generate a codebook which better preserves the semantic similarity of data. We treat pairwise relationships of some data points as the prior knowledge and use them to constrain the codebook generation and data decomposition. If a pair of data points are close to each other in original space or are labeled as similar with each other, they shall be assigned into the same cluster. On the contrary, if they are far away from each other or are labeled as dissimilar, they shall be allocated into different clusters. After decomposing the data space, a sub-codebook is built in each sub-space by clustering without breaking these rules. In order to achieve better data decomposition, we iteratively optimize the decomposition step and the codebook generation step by fixing the other. Then the optimized codebook and decomposition are used to encode data points, and to compute the distances between a query and the codewords which correspond to clusters. This provides an approximation to the distances between the query and the data points assigned to these codewords. We summarize the encoding process of our approach in Fig. 1.

The rest of the paper is organized as follows. We introduce related work in Section 2. The proposed method is described in Section 3. The experimental results and performance comparison with several alternative methods are presented in Section 4. The conclusions are drawn in Section 5.
In this section, we briefly summarize related work in quantization based image retrieval and semi-supervised clustering.

**Quantization Based Image Retrieval:** The most effective way for fast retrieval is to construct indices and performing retrieval based on the indices. Vector quantization [5] is such a solution. It allows a vector be mapped to a member (codeword) of a set of fixed vectors (codebook). Then a data sample is represented by the index of its corresponding codeword. To improve the scalability of vector quantization, product quantization (PQ) [6] generates clusters of short codes and then constructs codebook by using Cartesian product of these centers in subspaces for retrieval. This method normally can be implemented in several steps. Firstly, it divides each d-dimensional original vector to m sub-vectors, which is d/m-dimensional. Then the k-means algorithm is performed on m sub-vectors corresponding to the m subspaces. If the cluster number in each subspace is k, the total number of codewords is $k^m$, leading to time complexity of $O(mkd)$. In PQ, the codebook is composed of the Cartesian product of m sub-codebooks. Then a vectorized data point is quantized to the codewords and encoded in a composite code containing the indices of the nearest cluster centroids of all subspaces. Calculating the distance between the query and the quantized data item is a straightforward process.

Cartesian k-means [8] increases the quantization accuracy by finding an optimal rotation in feature space and then applying product quantization over the rotated space, whose idea is similar to optimized product quantization [7]. Additive quantization [13] uses the summation of codewords selected from different codebooks to be the quantizer in order to improve quantization accuracy. Composite quantization [10] improves the vector approximation accuracy by importing orthogonality condition between dictionaries, so as to guarantee the efficiency of approximate distance computation. Other variations of product quantization, including distance-encoded product quantization [14], sparse composite quantization [15] and locally optimized product quantization [9], address different aspects of quantization for performance improvement.

**Semi-Supervised Clustering:** Clustering algorithm is traditionally viewed as unsupervised approach. The purpose of clustering is distributing data to different clusters according to their similarities. In some cases, prior knowledge is also involved in the modelling, such as pairwise constraints between data points (must-link or cannot-link) or class labels for some samples. Note that data labels can be converted into pairwise constraints because the points can be defined as similar if they are in the same class, or dissimilar if they are in different classes. Semi-supervised clustering methods are developed to cope with such situation, aiming at fully exploring the prior knowledge and improving the clustering performance [16, 17, 18].

The pairwise constraints introduced by Wagstaff et al [16] provide prior knowledge on which data points should be grouped or not. An extended work, Pairwise Constrained k-means (PCKMeans) [17], plays an important role in semi-supervised clustering. It uses soft constraints which allows constraints be breached if an operation leads to a more cohesive clustering. The goal of PCKMeans is to minimize a combined objective function, defined as the sum of the total squared distances between the points and their cluster centers, and the cost caused by breaking any pairwise constraint. Let $M$ be a set of must-link pairs, where $(x_i, x_j) \in M$ denotes that $x_i$ and $x_j$ should be grouped in the same cluster, and $C$ be a set of cannot-link pairs where $(x_i, x_j) \in C$ denotes that $x_i$ and $x_j$ should be grouped in different clusters. The combined objective function of PCKMeans is as follows

$$J = \sum_{x_i \in X} \| x_i - \mu_{l_i} \|^2 + \sum_{(x_i, x_j) \in M} w_{ij} \zeta_{l_i \neq l_j} + \sum_{(x_i, x_j) \in C} \tilde{w}_{ij} \zeta_{l_i = l_j}$$

(1)

where point $x_i$ is assigned to center $\mu_{l_i}$, and $\zeta$ is an indicator function $\zeta_{true} = 1$ and $\zeta_{false} = 0$. The pairwise constraint $(x_i, x_j) \in M$ means $x_i$ and $x_j$ should be assigned to the same center, i.e., $l_i = l_j$ is true and $l_i \neq l_j$ is false. For $(x_i, x_j) \in C$, the data points should be assigned to different centers, thus $l_i \neq l_j$ is true and $l_i = l_j$ is false. $W = \{ w_{ij} \}$ and $\tilde{W} = \{ \tilde{w}_{ij} \}$ are defined as the penalty costs for breaking the constraints in $M$ and $C$, respectively.

**3. SEMI-SUPERVISED PRODUCT QUANTIZATION**

In this section, we describe the proposed Semi-Supervised Product Quantization (SSPQ) method. The training stage consists of two steps. The first step is building codebook.
Fig. 2. This figure shows clustering results when \( k = 3 \). (a) is the result of \( k \)-means clustering. (b) is the result of constrained \( k \)-means clustering. "\( \triangle \)" represents the center of a cluster. Solid line between two data points means they are similar pair, dotted line between two data points means they are dissimilar pairs. Green dotted line represents the region of cluster. Those points in the region are assigned to the same cluster. This figure tells that the result of semi-supervised \( k \)-means clustering is closer to real data distribution.

\[
C = C_1 \times C_2 \times \cdots \times C_m, \quad \text{where } C_i \text{ is the sub-codebook, by using two sets of pairwise constraints } Sim_e \text{ (the set of similar pairwise constraints) and } Sim_\neq \text{ (the set of dissimilar pairwise constraints) to learn codewords of SSPQ and quantizing the samples } x_1, x_2, \ldots, x_n \text{ in dataset } X \text{ to the codewords. The second step is updating the codebook } C_i \text{ and decomposition to minimize the distortion } D = \frac{1}{n} \sum_i \|x - q(x)\|^2 \text{ of quantization by iteratively fixing the codebook or orthogonal matrix } R \text{ while optimizing the other. This training process recovers the optimal codebook and decomposition for quantization.}
\]

### 3.1. Semi-Supervised Codebook Generation

Given a dataset \( X = \{x_1, x_2, \ldots, x_n\} \) of \( n \) \( d \)-dimensional points In semi-supervised product quantization, we split the training data \( x \) into \( m \) dimensional sub-vectors \( s_i, 1 \leq i \leq m \). The sub-vectors \( s_i \) is then quantized to \( k \) codewords \( \{c_{i1}, c_{i2}, \ldots, c_{ik}\} \), where \( c_{ij} \) is a cluster center of the \( i \)th subspace. Usually \( k_1 = k_2 = \cdots = k_m = k \), i.e., all subspace has the same number of clusters.

As is shown in Fig. 2, in this method, semi-supervised information is used in order to generate a codebook which maintains higher semantic similarity. The input to codebook generation are training data set \( X \), a set of similar-pair constraints \( Sim_e \), and a set of dissimilar-pairs constraints \( Sim_\neq \). We get codewords \( C_i = \{c_{i1}, c_{i2}, \ldots, c_{ik}\} \) separately in subspaces \( S_i = \{s_i(x_1), s_i(x_2), \ldots, s_i(x_n)\} \), where \( s_i(x_j) \) is the sub-vector of \( x_j \) to the \( i \)th subspace.

We randomly choose \( k \) sub-vectors from subpace \( S_M \) as the initial codewords \( \{c_{M1}, c_{M2}, \ldots, c_{Mk}\} \), and assign each sub-vector \( s_M(x_j), j = 1, \ldots, n \) to their closest codeword \( c_{Ma}, a = 1, \ldots, k \). Then we determine whether the initial assignment result violates the constraints or not, and make adjustment accordingly. If according to \( Sim_e \), the original data point \( x_i \) of assigned sub-vector \( s_M(x_i) \) is similar to \( x_j \), but \( s_M(x_j) \) is assigned to another codeword \( c_{Ma} \), we compare the distance of sub-vectors between their codewords. If \( d(s_M(x_i), c_{Ma}) < d(s_M(x_j), c_{Mb}) \), \( s_M(x_j) \) is assigned to \( c_{Ma} \), otherwise \( s_M(x_j) \) is assigned to \( c_{Mb} \).

On the contrary, if \( x_i \) is dissimilar to \( x_j \), but \( s_M(x_i) \) and \( s_M(x_j) \) are assigned to the same codeword \( c_{Ma} \), we compare the distance of sub-vectors to the codeword. If \( d(s_M(x_i), c_{Ma}) < d(s_M(x_j), c_{Ma}) \), an empty partition for \( s_M(x_j) \) is returned. Otherwise, an empty partition for \( s_M(x_i) \) is returned. After the adjustment, each codeword \( c_{Mi} \) is updated by averaging all sub-vectors \( s_M(x_j) \) that have been assigned to it. This process is repeated until convergence. Using this method, we can get a codebook that characterizes semantic similarity well in each subspace.

We utilize Cartesian product to build the codebook: \( C = C_1 \times C_2 \times \cdots \times C_m = \{(c_{i1}, c_{i2}, \ldots, c_{im})\} \), where \( i_j \in \{1, 2, \ldots, k\}, j = 1, \ldots, m \), \( c_{ij} = (c_{j1}, c_{j2}, \ldots, c_{jk}) \), \( c_{jk} \) is the \( k \)th center in the \( j \)th subspace. Therefore, the total number of codewords is \( k = k^m \). The data point \( x \) is quantized to its nearest codeword on the basis of the Euclidean distance:

\[
q(x) = \arg\min_{c \in C} d(x, c)
\]

This step also minimizes the following objective function

\[
\min_{C; \{b_i\}} \sum_{i=1}^{N} \|x_i - Cb_i\|^2
\]

where \( b_i = (b_{i1}^T, b_{i2}^T, \ldots, b_{im}^T)^T \). The \( d_i \) dimensional sub-vector \( b_{im} \) is a binary vector whose every entry is 0 expect that one element selected from \( k \) centers of the \( M \)th subspace is 1. \( C \) is a \( d \times mk \) matrix

\[
C = \text{diag}(C_1, C_2, \ldots, C_m) = \begin{pmatrix}
C_1 & 0 & \cdots & 0 \\
0 & C_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & C_m
\end{pmatrix}
\]

### 3.2. Iterative Optimization

Quantization approach can achieve higher accuracy when distortion becomes smaller [7]. Therefore, we optimize our algorithm by minimizing the quantization distortion:

\[
D = \frac{1}{n} \sum_x \|x - q(x)\|^2
\]

where \( \|\cdot\| \) is the Euclidean distance, and \( n \) is the total number of data points in the training set.

Let \( x = Rx, q(\hat{x}) = Rq(x) \), where \( R \) is an orthogonal matrix to minimize the distortion. Quantizing \( x \) to \( q(x) \) is equivalent to quantizing \( \hat{x} \) to \( q(\hat{x}) \). So we can use \( \hat{x} \) to represent data for quantization process optimization, such that \( \|x -
\( q(x)^2 = \|\tilde{x} - \tilde{q}(\tilde{x})\|^2 \). Following this analysis, with fixed \( R \) and \( q(x) \in C = \{ e^T R e \in C_1 \times C_2 \times \cdots \times C_m, R^T R = I \} \), the objective function

\[
\begin{align*}
\min_{R, C_1, \ldots, C_m} \sum_{x} \| x - q(x) \|^2
\end{align*}
\]

becomes

\[
\begin{align*}
\min_{C_1, \ldots, C_m} \sum_{x} \| x - \tilde{q}(\tilde{x}) \|^2
\end{align*}
\]

where \( \tilde{q}(\tilde{x}) \in C_1 \times C_2 \times \cdots \times C_m \). Then we can get the optimized codebook in this step by utilizing semi-supervised clustering method in every subspace as described before.

Because of \( \| x - q(x) \|^2 = \| Rx - q(\tilde{x}) \|^2 \), \( R \) can be optimized by fixing codebook \( C_1 \). The objective function becomes

\[
\begin{align*}
\min_{R} \sum_{x} \| Rx - q(\tilde{x}) \|^2 \\
\text{s.t.} \quad R^T R = I
\end{align*}
\]

We use matrix form to represent Eq. (8) and it is written as

\[
\begin{align*}
\min_{R} \| RX - P \|_F^2 \\
\text{s.t.} \quad R^T R = I
\end{align*}
\]

where \( X \) is composed by \( x \), \( P \) is a \( d \times n \) matrix whose columns are composed by \( q(\tilde{x}) \) and \( \| \cdot \|_F \) is the Frobenius norm. Our approach iteratively optimizes \( R \) and \( P \), following the same optimization process in [2].

### 3.3. Searching with Quantization

Given a query vector \( y \), before iterative optimization, we quantize vector \( x \) to \( q(x) \), so the distance \( d(x, y) \) between \( x \) and \( y \) is approximated by:

\[
\tilde{d}(x, y) = d(q(x), y)
\]

According to Section 3.2, quantizing \( x \) to \( q(x) \) is equivalent to quantizing \( \tilde{x} \) to \( \tilde{q}(\tilde{x}) \). Therefore, Eq. (10) becomes

\[
\tilde{d}(x, y) = d(q(\tilde{x}), \tilde{y}) = \sqrt{\sum_{j} d(\tilde{q}(s_j(\tilde{x})), s_j(Ry))^2}
\]

where \( s_j(\cdot) \) is the \( j \)th sub-vector of the original data, and \( d(c_{jk_i}, s_j(Ry))^2 : j = 1, \ldots, m, i = 1, \ldots, k \) of every vector in the target set is computed before searching. The final retrieval can be achieved by finding \( k \) vectors with the smallest distances between the query and target set samples.

### 4. EXPERIMENT

In this section, we present the experimental results of our method.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Dataset Size</th>
<th>Query Set Size</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>70000</td>
<td>10000</td>
<td>784</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>60000</td>
<td>10000</td>
<td>320</td>
</tr>
<tr>
<td>ImageNet</td>
<td>1200000</td>
<td>50000</td>
<td>4096</td>
</tr>
<tr>
<td>SIFT-1M</td>
<td>10000000</td>
<td>10000</td>
<td>128</td>
</tr>
</tbody>
</table>

**Table 1.** Datasets used in the experiments.

**Fig. 3.** The performance of different algorithms on MNIST and CIFAR-10 with 64bits encoding for searching various numbers of ground truth nearest neighbors (N=1,10).

**4.1. Datasets and settings**

We ran experiments on four datasets, including MNIST [19], CIFAR-10 [20], ImageNet [21], and SIFT-1M [6].

- MNIST includes 70k grayscale images of handwritten digits from ‘0’ to ‘9’. Every image sample is represented by a 784 dimensional vector. We randomly chose 10000 data vectors as queries and used the rest as retrieval dataset.

- CIFAR-10 includes 60k tinny color images divided into ten classes with 6k images per class. Every image is represented by a 320 dimensional vector. We randomly chose 1000 data vectors as the queries and then used the remaining as retrieval dataset.

- ImageNet includes over 1.2 million images of 1000 classes. It provides a training dataset which we used as the retrieval dataset. We used 50k provided images as the query set. We present images by 4096 dimensional feature vectors as in [22].
SIFT-1M consists of 1 million 128 dimension SIFT vectors to represent each images [23]. The dataset provides 10k query samples, which formed the query set in our experiments. We randomly chose 1000 pairs of data points to form the pairwise constrains, and used the remaining as retrieval dataset. Table 1 summarizes the specification of each dataset.

4.2. Evaluation Criteria

Following [6], we set $k = 256$ as the codebook size in each subspace, which allows the subindex be stored in 1 byte. The code length of our method was set to $m \log_2 k$, where $m$ was the number of subspaces. We set $m = 2, 4, 8, 16$ and the code length to be 16, 32, 64, 128 bits, respectively. We defined $N$ Euclidean nearest neighbors as the true neighbors and in this work we chose $N = 100$ because the results of $N = 1$ to 1000 are similar according to experiments.

The query performance is measured with recall over $R$, i.e., the proportion of query vectors for which the nearest neighbor is ranked in the top $R$. This measure shows the proportion of queries for which the nearest neighbor is retrieved correctly. In practice, we are often interested in retrieving the $N$ nearest neighbors ($N > 1$) and not only the nearest neighbor. For each query, we computed the proportion of $R$ to $N$, which is the ratio $N$ ground-truth nearest neighbors are found in $R$ retrieved vectors. The average recall score over all the queries was used as measure. In our experiments, results with $N$ being 1 and 10 are reported.

We also adopted the widely used mean average precision (mAP) criterion, defined as $mAP = \frac{1}{Q} \sum_{i=1}^{Q} AP(y_i)$, where $Q$ is the number of queries, and $AP$ is computed as $AP(y) = \frac{1}{L} \sum_{r=1}^{L} P_y(r) \delta(r)$. $L$ is the number of true neighbors for the query $y$ in the $n$ retrieved samples, where $n$ is the size of the dataset except that $n$ is 1500 on the ImageNet dataset for evaluation efficiency. $P_y(r)$ denotes the precision when top $r$ data points are returned, and $\delta(r)$ is an indicator function which is 1 when the $r$th result is a true neighbor and otherwise 0. We treated the 100 nearest neighbors as the ground truth.

4.3. Results Comparison

We compared our semi-supervised product quantization (SSPQ) method with several baseline methods such as product quantization (PQ) [6], Cartesian k-means (CKM) [8], composite quantization (CQ) [10] and iteratively quantization (ITQ) [2]. It is already shown that quantization based methods have better retrieval accuracy than hashing algorithms with the same code length and comparable search efficiency (see Section 4.2). Therefore, we just show one result from ITQ [2] as a representing unsupervised hashing approach.

Figure 3 shows the comparison on MNIST and CIFAR-10 for searching 1 and 10 nearest neighbors using codes of 32 bits. Figure 4 shows the comparison on ImageNet and SIFT-1M for searching 1 and 10 nearest neighbors using codes of 64 bits. It can be observed that our method SSPQ outperforms product quantization and its variation models. This meets our expectation because SSPQ takes advantage of prior knowl-
edge to generate the codebook which attains more accurate approximating vectors to codewords. Our optimization also finds a better decomposition of data space.

The mean average precision (mAP) with the code length of 16, 32, 64, and 128 are shown in Figure 5. Our SSPQ method also achieves the best performance. The improvement on MNIST, CIFAR-10, and ImageNet is significant comparing with the alternative approaches. Our approach achieves relatively small improvement over CQ and CKM on SIFT-1M. This shows that learning with semi-supervised product quantization indeed benefits the search performance.

We report the search efficiency of PQ, CQ and ITQ and the proposed SSPQ on MNIST and SIFT-1M in Table 2. It can be seen that the searching time of these retrieval methods are similar. Our method achieves close searching time as PQ, and is more efficient than CQ by a small margin. Compared with PQ and its variants, our method achieves higher retrieval accuracy while maintaining the search efficiency.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>SSPQ</th>
<th>PQ</th>
<th>CQ</th>
<th>ITQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>0.52(ms)</td>
<td>0.41(ms)</td>
<td>1.53(ms)</td>
<td>0.50(ms)</td>
</tr>
<tr>
<td>SIFT-1M</td>
<td>7.01(ms)</td>
<td>6.89(ms)</td>
<td>7.14(ms)</td>
<td>6.87(ms)</td>
</tr>
</tbody>
</table>

### 5. CONCLUSION

In this paper, we have introduced a semi-supervised product quantization method. It uses prior knowledge to preserve the semantic similarity of data in codebook generation. During the process, pairwise relationship of data points and their labels are treated as the prior knowledge. In addition, we iteratively optimize the codebook generation and data decomposition steps so as to achieve better data decomposition. Experimental results on different datasets show that our method has outperformed several baseline methods.

### 6. REFERENCES


Oral Session H

Biometrics and medical applications

Wednesday May 16, 2018, 10:00 AM

Hai-Duong Nguyen, Soonja Yeom, Il-Seok Oh, Kyoung-Min Kim and Soo-Hyung Kim
Facial Expression Recognition Using a Multi-level Convolutional Neural Network

Nabil Belacel and Miroslava Cuperlovic
Machine learning and feature selection for the analysis of Alzheimer Metabolomics Data

Yong-Jeong Kim, No-Sang Kwak and Seong-Whan Lee
Sequential Backward Spatio-Spectral Filter Optimization for Motor Imagery Classification in Ear-EEG Brain-Computer Interface

Huafeng Qin and Mounim A. El Yacoubi
Finger-vein quality assessment by joint representation learning from grayscale and binary images

Catherine Cheung, Julio Valdes, Richard Salas Chavez and Alejandro Lehman Rubio
Failure modelling and anomaly detection of a propulsion subsystem
Abstract—Even though many breakthroughs have been made in image classification, especially in facial expression recognition, this research area is still challenging in terms of wild sampling environment. In this work, we carry out a study of multi-level features in a convolutional neural network for facial expression recognition. Based on our observations, we introduce a model by combining a hierarchy of features intentionally to improve the classification task. Our model was evaluated on the FER2013 dataset and achieved a comparable performance to the current state-of-the-art methods.

Index Terms—facial expression recognition in the wild, FER2013, hierarchical features, multi-level convolutional neural networks

I. INTRODUCTION

Facial expression recognition is still a challenging issue in human-machine interaction research area. In order to improve user experience, researchers are trying to change the way that we can interact with computers, and making machines understand our feelings. Traditional approaches with handcrafted features achieved competitive results on the data acquired in a controlled environment. On the other hand, facial expression recognition in the wild is more difficult due to various aspects in an unconstrained sampling scenario, e.g., dynamic illumination, occlusion, head poses, etc. Recently, many breakthroughs for image classification have been achieved by using deep convolutional neural networks (CNNs). These architectures consist of two main components: an automatic feature extractor and a classifier. The first component produces low-level, mid-level, and high-level features describing simple, moderate, and complex textures for the object of interest respectively. Generally, a strong classifier learns the target from a large number of high-level features, that is, a huge amount of data should be used to train the network. Thanks to the availability of public large-scale image dataset and powerful hardware, deep learning techniques are widely used in many image classification issues, e.g., ImageNet [1], PASCAL VOC [2], CIFAR [3], Facial Expression Recognition Challenge 2013 (FER2013) [4], etc.

The first successful CNN was developed by LeCun in the 1990s which is famous for handwritten digit recognition [5]. In 2012, Krizhevsky et al. proposed a deep CNN named AlexNet for 1000-class image classification problem [6], their model outperformed significantly the runner-up and led to a breakthrough in the ImageNet challenge. Deeper networks with a stack of convolutional layers continued to be improved by adding layers, changing feature map size, etc. Simonyan et al. presented VGG nets and carried out an analysis of how the depth improves classification accuracy [7]. One of the issues in this research area was choosing the size of the filters, Szegedy et al. introduced the Inception module which includes $1 \times 1$, $3 \times 3$, and $5 \times 5$ filters in the same layer [8]. Given a dataset, their model named GoogLeNet automatically selects appropriate filter sizes and outperformed the others to be the winner of ImageNet challenge 2014. Even though the invention of Inception module leads to significant improvements, researchers figured out that small filter such as $3 \times 3$ produces a good performance for most of image classification problems. Therefore, recent works focus on improving performance by adding layers to the network. However, as increasing the number of layers, it is more difficult for the information to flow, which is known as degradation problem or gradient vanishing. ResNet - the winner model of ImageNet challenge 2015 - outperformed previous ensemble models by using Identity mapping [9]. This connection copies the learned shallower layer directly to the latter. Recently, Huang et al. introduced an improvement of Identity mapping called Dense connectivity in their DenseNet [10], which connects any layers.
to all the subsequent layers and boosts information to flow in the network.

One of the main contributions of this paper is our study of low-level and mid-level features in CNNs for facial expression recognition in the wild. The analysis shows an effective way to use these types of features to improve classification performance. In lieu of feeding either high-level or a combination of three types of features in a CNN into the classifier, we propose variants of connection in the network to select the most informative features for the classification task. In addition, the proposed architecture also eliminates trivial information to improve the overall performance. By combining a hierarchy of features intentionally, our model\(^1\) archived a comparable performance to the current state-of-the-art methods on the FER2013 dataset.

The rest of the paper is organized as follows. In Section II, we review recent works related to facial expression recognition in the wild. Most of them are the state-of-the-art methods in this research area. The details of our architecture for facial expression recognition will be discussed in Section III. Section IV shows our experimental results and the comparison between our model and the state-of-the-art methods. In Section V, we conclude our research.

II. RELATED WORKS

In FER2013 challenge of the ICML 2013 Representation Learning [4], Tang introduced a CNN jointly learned with linear support vector machine (SVM) for facial expression recognition [11]. With a simple CNN and a SVM instead of softmax classifier, the model outperformed the others and won the first place in the challenge. Inspiring by the success of GoogLeNet [8], Mollahosseini et al. proposed an architecture containing four Inception modules [12]. However, their research cannot lead to a better performance on the FER2013 dataset. In 2016, Zhou et al. proposed the multi-scale CNNs [13]. This model consists of three other networks with different input sizes. In addition, they used late fusion technique to get the final classification results. By combining multiple CNNs and modifying the loss function, Yu et al. obtained a higher accuracy compared to the previous approaches [14]. Similarly, Kim et al. introduced a multiple CNNs for facial expression recognition in the wild [15]. Another multi-scale CNN was proposed by Wang et al. [16]. In this work, the authors use the entire feature maps in the network for classification. However, using all generated features without selection may reduce the overall performance due to trivial information in shallow layers of the network. To the best of our knowledge, this work is the current state-of-the-art method on the FER2013 dataset.

III. PROPOSED METHOD

A. Plain Networks

Networks with a stack of convolutional layers are very common in visual classification problems. Due to the accessibility of public data and high-performance systems, some researchers studied the relationship between classification performance and the depth of the network [7]–[9], [17]. As increasing the number of convolutional layers, the networks capture better information for category discrimination, which is called high-level feature. In order to obtain this information, the networks combine low-level and mid-level features acquired from the intermediate convolutional layers as shown in Fig. 1. Generally, the classifier with hidden layers only takes the high-level feature into account. An excellent example for the success of this type of network is VGG net [7]. This model consists of several blocks of convolutional layers and the last block is followed by fully connected layers for classification. This architecture outperformed the others in ImageNet Challenge 2014 and is still popular for transfer learning in many computer vision issues.

In this research, one of the proposed architectures for facial expression recognition is a deep plain CNN inspired by VGG net with 18 weighted layers organized into 5 blocks (Fig. 2a). Each block contains 2, 3, or 4 convolutional layers and followed by max-pooling. The input of the network is a 48 × 48 gray-scale image and the output of the bottleneck layer are 512 1 × 1 filters, i.e., the fully connected layer contains only a 7-way softmax classifier. Note that bottleneck is the layer just before the final output layer of the network. Since we do not use any fully connected layers before softmax classifier, it is reasonable to analyze how a filter contributes to the final classification. Given a network, Selvaraju et al. [19], [20] proposed a technique called Gradient-weighted Class Activation Mapping (Grad-CAM) to visualize the regions of the input that are useful for classification. In other words, their method explains how a CNN places attention on an object of interest at a specific network layer. Fig. 3c shows a Grad-CAM generated by the 24th filter in the third block of our plain architecture. At this level, the network somehow focuses on the eyes, the mouth, which play a significant role in facial expression recognition. On the other hand, at the first block, the network gives its attention to the background and other meaningless regions as shown in Fig. 3b.

In addition, Fig. 4 presents other Grad-CAM visualizations generated from the third block. Even though these filters are very worthwhile to construct higher-level features, it can be seen that all of them do not contribute directly to facial expression recognition. It turns out only a number of filters in the mid-level of the network are important for classification. From this point of view, we propose a multi-level convolutional neural network for facial expression recognition in the wild.

B. Multi-level Convolutional Neural Networks

The backbone of the proposed model is an 18-layer CNN inspired by VGG net [7]. However, our architecture not only uses high-level feature for classification but also takes mid-level features into account as described in Fig. 2b. These feature maps are extracted from the 2nd, the 3rd, and the 4th blocks of the network. We do not use the information from the 1st block directly due to the fact that it includes trivial filters for facial expression recognition. As shown in Fig. 3b,
Fig. 1: Hierarchical features in CNNs [18].

Fig. 2: Plain and multi-level architectures. Multi-level networks consider both mid-level and high-level features in classification. This information can be extracted through middle blocks of the network.

Fig. 3: Face image and Grad-CAM visualizations with our 18-layer plain network.

Fig. 4: Trivial filters from the 3rd block of our 18-layer plain network.

the network at this level only focuses on meaningless regions, e.g., background, hair, etc. This is one of the improvements of our multi-level convolutional neural network (MLCNN) compared to Wang’s approach [16]. In addition, according to the observation on the mid-level features carried out in the previous section, in the same block, there are a number of unimportant filters for facial feature extraction (Fig. 4) even though the number of filters in low- and mid-level layers is higher compared to high-level layer. Therefore, we insert a fully connected layer with 256 units to reserve useful filters as well as eliminate insignificant ones. The network connection is the vector concatenation operator defined as:

\[
(x_1, x_2, ..., x_T) \oplus (x_1, x_2, ..., x_T) = (x_1, x_2, ..., x_1, x_2, ..., x_T) \quad (1)
\]

where \((x_1, x_2, ...)^T\) and \((x_1^2, x_2^2, ...)^T\) are feature vectors from different network levels. In case the input is an output of convolutional layers, it needs to be vectorized before applying this operator.

Another issue that most of deep learning researchers have to deal with is how to choose the depth of the network. Generally, we make the decision based on experiments [9]. In some ways,
our multi-level architecture itself selects important features at each level for classification, i.e., given a base model, the network determines the depth based on filters’ contribution.

IV. EXPERIMENTAL RESULTS

A. FER2013 Dataset

We evaluate our method on a large dataset originally used for the Facial Expression Recognition Challenge [4]. This dataset consists of 35887 48 × 48 gray-scale images: 28709 for training, 3589 for evaluation and the rest for testing. Each of them belongs to one of seven expressions: angry, disgust, fear, happy, sad, surprise, and neutral. One of the challenges on this dataset is data unbalancing. The details are given in TABLE I and Fig. 5. In addition, this dataset contains a number of invalid samples including non-face images, incorrect face cropping, and expression labeling errors as shown in Fig. 6.

Due to a large number of training samples as well as the sampling scenario, this dataset is very common for evaluating facial expression recognition methods, especially deep learning based techniques.

B. Facial Expression Recognition on FER2013

We trained the plain network using Keras [21] with TensorFlow backend [22] and the training data from FER2013 dataset for up to 2000 iterations. The weights were initialized randomly before updating by Adam algorithm [23] with mini batch size of 1024. The learning rate started from $10^{-4}$ and the minimum value was $10^{-7}$. To prevent overfitting, data augmentation was involved for training data. Data preprocessing techniques include normalization, translation, rotation, scaling, and mirroring. In addition, we used dropout [24] after max-pooling layers in order to improve the network generalization capability.

To train MLCNN, we initialized it with the weights of the plain architecture before training for around 100 iterations and the model with the best performance was selected. The loss function that we used to train our models is cross entropy and the activation function is ReLU.

TABLE I: FER2013 Dataset

<table>
<thead>
<tr>
<th>Expression</th>
<th>Training</th>
<th>Validation</th>
<th>Testing</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>angry</td>
<td>3995</td>
<td>467</td>
<td>491</td>
<td>4953</td>
</tr>
<tr>
<td>disgust</td>
<td>436</td>
<td>56</td>
<td>55</td>
<td>547</td>
</tr>
<tr>
<td>fear</td>
<td>4097</td>
<td>496</td>
<td>528</td>
<td>5121</td>
</tr>
<tr>
<td>happy</td>
<td>7215</td>
<td>895</td>
<td>879</td>
<td>8989</td>
</tr>
<tr>
<td>sad</td>
<td>4830</td>
<td>653</td>
<td>594</td>
<td>6077</td>
</tr>
<tr>
<td>surprise</td>
<td>3171</td>
<td>415</td>
<td>416</td>
<td>4002</td>
</tr>
<tr>
<td>neutral</td>
<td>4965</td>
<td>607</td>
<td>626</td>
<td>6198</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>28709</strong></td>
<td><strong>3589</strong></td>
<td><strong>3589</strong></td>
<td><strong>35887</strong></td>
</tr>
</tbody>
</table>

TABLE II: Our performance on FER2013 testing set

<table>
<thead>
<tr>
<th>Model</th>
<th>Architecture</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plain model</td>
<td>18 layers</td>
<td>69.21</td>
</tr>
<tr>
<td>MLCNN</td>
<td>Plain model + multi-level connections</td>
<td>73.03</td>
</tr>
</tbody>
</table>

TABLE III: Performance comparison on FER2013 testing set

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inception [12]</td>
<td>66.40</td>
</tr>
<tr>
<td>Multi-scales CNNs [13]</td>
<td>71.80</td>
</tr>
<tr>
<td>MNL [14]</td>
<td>72.08</td>
</tr>
<tr>
<td>Hierarchical committee of CNNs [15]</td>
<td>72.72</td>
</tr>
<tr>
<td>Multi-scale CNN [16]</td>
<td>72.82</td>
</tr>
<tr>
<td>Our MLCNN</td>
<td>73.03</td>
</tr>
</tbody>
</table>
other single [11], [12], [16] and ensemble [13]–[15] models. Since our MLCNN is more robust to select informative midlevel features, it performs better than Multi-scale CNN [16] which also focus on different levels of feature in the network for classification. Although our model performance is better than the others, one of the drawbacks is that it involves two stages for training. The plain model and the MLCNN should be trained separately since the weights of the former are used in the initialization step for training the latter.

V. CONCLUSION

In this research, we carried out a study about hierarchical features in CNNs for facial expression recognition. The experimental results indicate that in addition to high-level feature, a number of mid-level features also play a significant role in the classification task. However, this information should be handled carefully. In this paper, we proposed an MLCNN that automatically selects important mid-level and high-level features based on their contribution. Our architecture outperforms other single and ensemble models on the FER2013 dataset, which is considered to be a promising method for facial expression recognition in the wild.

ACKNOWLEDGMENT

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2015R1D1A3A01018993), the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2017R1A4A1015559), and the Ministry of Trade, Industry and Energy(MOTIE) through the Regional Innovation Centre Programme.

REFERENCES


Machine learning and feature selection for the analysis of Alzheimer Metabolomics Data

Nabil Belacel  
Digital Technology  
National Research Council  
Ottawa, Canada  
Nabil.Belacel@nrc.gc.ca

Miroslava Cuperlovic-Culf  
Digital Technology  
National Research Council  
Ottawa, Canada  
Miroslava.Cuperlovic-Culf@nrc.gc.ca

Abstract—Metabolomics provides a highly valuable data for disease diagnosis and prediction, due to its accurate, possibly non-invasive and low cost determination of highly physiologically relevant molecular data. At the same time, early and accurate Alzheimer disease diagnosis remains a highly challenging task. Consequently, number of researchers have recently become interested in utilizing metabolomics data for the discovery of biomarker for classification of Alzheimers disease (AD). However, although many methods already exist for the determination of markers for ADs identification from high throughput data, more precise and accurate method for feature selection as well as AD classification are still needed. Various machine learning approaches have achieved successful classification of samples between cognitively healthy and AD using metabolomics data. However, they failed to achieve a good classification rates for differentiation between AD, mild cognitive disorder and cognitively healthy or normal. In this paper, we propose new machine learning approaches to select a subset of features that improve the classification rates between these three classes, thus allowing separation between different levels of cognitive disorders. Our experiment results demonstrate that the performances of several classifiers are improved when using our selected metabolic markers relative to the classification that can be obtained from the complete metabolomics dataset. The obtained results indicate that our algorithms are effective in discovering markers for AD classification from metabolomics data.

Index Terms—Machine learning, Feature selection, Classification, Metabolomics data, Alzheimer disease

I. INTRODUCTION

Metabolite profiling, or metabolomics, has become a powerful approach for better understanding of biological mechanisms involved in human pathology development and identifying early predictive biomarkers [1]. It provides a highly valuable data for disease diagnosis and prediction, due to its accurate, possibly non-invasive and low cost determination of highly physiologically relevant molecular data. While high-throughput metabolomics platforms, such as mass spectrometry, provide tools for simultaneous study profiles of hundreds of metabolites in a single experiment, the number of independent samples is usually substantially smaller. One of the solution to this problem is to apply machine learning with feature selection [1]. The problem of feature selection is well known in machine learning and its goal is to identify a subset of relevant features [2]. In this paper, we propose an algorithm from machine learning to select a subset of features with application to Alzheimer Disease (AD). Early and accurate AD diagnosis remains a highly challenging task. Consequently, number of researchers have recently become interested in utilizing metabolomics data for the discovery of biomarker for diagnosis and classification of AD [3]. However, although many methods already exist for the determination of markers for ADs identification from high throughput data, more precise and accurate method for feature selection as well as AD classification are still needed. Various machine learning approaches have shown classification of cognitively healthy and AD samples using metabolomics data [4]. However, previous efforts have failed to achieve a good classification rates for differentiation between AD, mild cognitive disorder and cognitively healthy or normal patients [5]. In this paper, we propose new machine learning approaches for selection of a subset of features that improve the classification rates between these three classes, thus allowing separation between different levels of cognitive disorders.

II. METHODS AND MATERIALS

A. Alzheimer Diseases

We are interested in the application of our novel machine learning and feature selection methodology to AD. AD is an age-related, brain disease characterized by progressive cognitive impairment and dementia with increasing impairment in multiple cognitive domains including memory, executive functions and language [6]. Pathological hallmarks of AD are neurofibrillary tangles composed of abnormally phosphorylated, conformed and truncated tau, and senile plaques with a core of the altered cleavage of amyloid precursor protein leading to the deposition of b-amyloid. In addition several other anomalies have been observed including mitochondrial malfunction, increased oxidative stress and oxidative and nitrosative damage to nucleic acids, proteins and lipids; energy metabolism changes; altered composition of lipids and lipid rafts; autophagy; neuroinflammation, deregulation of purine metabolism [7] and number of other metabolism pathways [8], dysregulation of unsaturated fatty acid metabolism [9] to name just a few. With expected major increase in the number of AD patients worldwide and associated impact in human cost and suffering as well as economic impact there is a pressing need for better understanding of the
AD etiology, early diagnosis as well as treatment for this disease. Metabolomics analysis provides information about the perturbations in the metabolome reflecting changes in genome, transcriptome and proteome caused by physiological as well as environmental impacts. In this way metabolomics provides molecular data closest to the phenotype. Metabolomics measurements in brain, cerebrospinal fluid (CSF) and other body fluids can provide diagnostic markers as well as information about the disease development and possibly treatment options and follow-up avenues. Trushina et al. [8] have presented a metabolomics measurement of a cohort of cognitive normal (CN), Mild cognitive disorder (MCI) and Alzheimer disease (AD) patients that were enrolled in the longitudinal Mayo Clinic Study of Aging and Mayo Clinic Alzheimer Disease Research Center. CSF metabolomics measurements were performed using non-targeted ultra-performance liquid chromatography coupled to time-of-flight mass spectrometry detecting 351 metabolite peaks with 74 identified metabolites. CSF measurements were performed and provided for 15 AD, 15 CN and 15 MCI subjects. Other details of the cohort are provided in the original publication [8]. Dataset for analysis presented in this publication is obtained from Metabolomics Workbench [10].

B. PROAFTN classification method

In this section we briefly describe fuzzy classification method PROAFTN with more detailed description provided in earlier references [11]–[13]. PROAFTN method belongs to the class of supervised learning and it is used for solving classification problems. To calculate the similarity between samples to assign and the prototype of classes the PROAFTN based on the preference relational systems as described by Roy [14]. It employs a partial comparison between the samples and prototypes of the classes for each feature. Then, it applies a global aggregation using the concordance and non-discordance principle [11], [14]. Therefore it avoids resorting to conventional distance that aggregates the score of all features in the same value unit. Hence, it helps to overcome some difficulties encountered when data is expressed in different units and to find the correct preprocessing and normalization data methods. Therefore, PROAFTN is the appropriate technique for classifying metabolomics profile. Let each sample $s$, which we need to classify, is described by a set of $m$ attributes or features $g_1, g_2, \ldots, g_m$, let $w_1^h, \ldots, w_m^h$ is set of weights of each feature to class $h$ for $h = 1, \ldots, k$ and let $\{C_1, \ldots, C_k\}$ be the set of $k$ classes.

The classification procedure PROAFTN used in this paper to classify the samples is summarized in Algorithm 1.

In the following, we describe our feature selection using feature ranking and greedy algorithm.

C. Proposed feature selection approach

The problem of features selection is well known in machine learning for biomarker discovery. Generally, feature selection algorithms are divided into three categories: filters, wrappers and embedded approaches [16]. Filters methods evaluate quality of selected features, independent from the classifier, while wrapper methods require application of a classifier and it should be trained on a given feature subset to evaluate its quality. Embedded methods perform feature selection during

---

**Algorithm 1** PROAFTN classification procedure.

**Input:** $S$: set of samples; $K$: the number of classes; $w_i^h$ the weight of the feature $i$ of the class $h$. $S$ is divided into training and test sets.

**Output:** $\delta(s, C^h)$: the membership degree of $s$ to $C^h$

Step 1: Building the classification model for PROAFTN: Assign a relative importance weights $w_j^h$, $j = 1, \ldots, m; h = 1, \ldots, k$ to features.

From the training set: Apply the discretization and inductive algorithm to build the prototype of the classes as in [12], [15]. Each prototype $b_i^h$ is defined by $m$ features $g_j$, $j = 1, \ldots, m$ and its score in each feature is defined by two intervals: - pessimistic: $[S^1_j(b_i^h), S^2_j(b_i^h)]$; and - optimistic $[d^1_j(b_i^h), d^2_j(b_i^h)]$ as presented in figure 1.

Step 2: Compute the indiffERENCE relation between the sample $s$ and the prototype $b_i^h$ of the class $h$:

$$I(s, b_i^h) = \sum_{j=1}^{m} w_j^h C_j(s, b_i^h)$$ (1)

$$C_j(s, b_i^h) = \min \{C^1_j(s, b_i^h), C^2_j(s, b_i^h)\}$$ (2)

where

$$C^1_j(s, b_i^h) = \frac{d^1_j(b_i^h) - \min\{S^1_j(b_i^h) - g_j(s), d^2_j(b_i^h)\}}{d^2_j(b_i^h) - \min\{S^1_j(b_i^h) - g_j(s), 0\}}$$

$$C^2_j(s, b_i^h) = \frac{d^2_j(b_i^h) - \min\{g_j(s) - S^2_j(b_i^h), d^1_j(b_i^h)\}}{d^2_j(b_i^h) - \min\{g_j(s) - S^2_j(b_i^h), 0\}}$$

Step 3: Evaluation of the membership degree:

$$\delta(s, C^h) = \max\{I(s, b_1^h), I(s, b_2^h), \ldots, I(s, b_k^h)\}$$ (3)

Step 3: Assign sample $s$ to the class:

$$a \in C^h \Leftrightarrow \delta(s, C^h) = \max\{\delta(s, C^i) / i \in \{1, \ldots, k\}\}$$ (4)

---

![Fig. 1. The concordance index between the sample $s$ and the prototype $b_i^h$](image-url)
learning (for example, determining the feature intervals and features’ weights during PROAFTN model building). Given machine learning classifier such as decision trees, support vector machine, neural network or PROAFTN, it is conceivable to select the best subset of features satisfying a given model selection criterion by exhaustive enumeration of all subsets of features. Exhaustive enumeration is impractical for large number of features (thousands or hundreds of metabolites) due to the combinatorial explosion of the number of subsets [16]. Hence, to perform feature selection for metabolomics data, we combine the greedy algorithm that is heuristic for combinatorial problem with the machine learning classifier PROAFTN. Our heuristics apply the backward elimination starting with the whole features and it eliminates subset after subset by local search. The following steps summarize our feature selection approach:

- *Step 1*: Apply PROAFTN classifier as presented in algorithm 1 using the whole number of features with same weight of importance. Calculate the correct classification $f$. Let the subset of feature $SF$ initially is $m$; $SF := m$.
- *Step 2*: Using Monte Carlo simulation method, randomly generate the weights for the $SF$ features, i.e. generate randomly $|SF|$ numbers between 1 and $|SF|$. Then, Scale the generated number to have minimum 0 and maximum 1 and sum to one on the possible domain of weights.
- *Step 3*: Apply PROAFTN classifier with different weights of features generated in step 2. Then, calculate the correct classification $f^t$. $iff$ $f^t \geq f$ goto step 4; Otherwise goto step 2.
- *Step 4*: Remove all the features that have weight smaller or equal to a given threshold $t_0$. We suppose we will have $SF^t$ subset of features. Go to step 2 with remaining features $SF = SF^t$.
- *Step 5*: Repeat steps 2 to 4 until there is no improvement or after given CPU time.

The pseudo optimal subset of features obtained from the above approach can be ranked using the different values of the features weights. The obtained order of weights provides information about the most important features from the selected subset, i.e. features that have the highest rate of correct classification. Since this algorithm is not exhaustive, the obtained weights from the selected features are referred to pseudo optimal weights of the features subset. It should also be pointed out that the optimal subset of features may not be unique.

### III. Experiments

#### A. Data set

The dataset presented in this publication is obtained from metabolomics Workbench [10]. It includes measurement of 45 subjects divided in three groups: 15 belongs to the class AD, 15 belongs to the class CN (control) and the last 15 cases belong to the intermediate class MCI. After noise filtration of metabolomics data, each subject is described by 686 variables "measurements" after the data pre-processing. During the data pre-processing we have just removed all the features that have at least one variable or measurement is missing. To study the usefulness of the discovered features, we have calculated the performances of several classifiers on different subsets of features based on leave-one-out cross validation (LOOCV). The proposed method PROAFTN with the proposed feature selection were implemented using C++ language. We have compared our classifier method PROAFTN with the proposed feature selection to several classifiers with feature selection. The experiment on the comparative classifiers is conducted using WEKA (Waikato Environment for Knowledge Analysis) implementation of seven classifiers including: Decision tree C4.5 (J48), Multilayer Neural Network (NN(MLP)), Nearest Neighbor(NN), Random Forest, Naive Bayes, Support Vector Machine (SVM) and Bayesian Network [17]. We have also used the Weka for feature selection. We have tested several techniques, however we have only reported one method that gave the best results on data set used in our experiment II.

**Performance metrics**: The metrics used for evaluating the performance of the proposed work are described below:

- *Correctly Classified samples* It is the average percentage of correctly classified samples as AD, CN or MVI.
- *Incorrectly Classified Instances* It is the average percentage of samples that are incorrectly classified into one of the three classes (AD, CN or MVI).
- *Kappa statistic:* It measures the agreement of prediction with the true class. If the kappa value is 1, there is a complete agreement with the true class.
- *Matthews Correlation Coefficient (MCC):* It is a correlation coefficient between the observed and predicted classification results. If MCC is +1, it represents the perfect classifications and -1 indicates the complete disagreement between the observed and predicted classification results.

#### IV. Results and Discussion

Several phases will need to be completed in order to ascertain whether the feature selection methods improve classification. First, we classify the initial full dataset using all measured features as presented in table I. Feature selection will then be carried out using our proposed approach and the search method, obtaining a new reduce dataset for each method. These new subsets of features obtained by each method will be introduced into the different classifiers, producing new accuracy as shown in tables II and table III. We will already be able to observe in this step whether the methods improve or worsen efficiency by themselves [18]. As shown in the table I only the classifier neural network (NN) got the higher accuracy with 51.11% of correct classification and with kappa value of 0.267. All other classifiers have an accuracy below 50%. The tree based classifiers and probabilistic approaches have lower accuracy with negative value for kappa. The PROAFTN method has an accuracy of 44.44% when all features have the same weight. The results of PROAFTN classification reaches more than 60% of correct classification for the whole set of features when including different weights (data not shown).
After we removed the features that have at least one measurement missing, we have started applying the machine learning classifiers on 686 features. Different approaches for feature selection were tested with two approaches reported here. The first one is based on filtering method used in Weka known as Greedy Stepwise. This approach uses a forward (additive) or backward (subtractive) step-wise strategy to navigate attribute subsets. Table II, reports the different results using LOOCV and after applying the feature selection Greedy stepwise technique. As shown in the table II all classifiers do not succeed to get more than 50% of correct classification. For example the performances of NN were decreased (accuracy decreased from 51.11% to 48.89%, the MCC from 0.265 to 0.233 and the kappa from 0.267 to 0.233). However, the remaining classifiers their performances increased but they still have accuracy less than 50%. Even though, the PROAFTN method has an accuracy of 46.667%, it obtained the highest MCC value with 0.331.

As shown in table III the performances of all classifiers were improved with feature selection except for the neural network NN (MLP) where the accuracy decreased from 51.11% to 48.89%. However, the performances of the other classifiers as presented in table III are improved. The PROAFTN method reached 75.556% accuracy of correct classification and the MCC with value 0.45. The k = 3 nearest neighbor has an accuracy of 57.78% of correct classification and MCC with value 0.371 and Kappa with value 0.367. The Bayesian network reached 51.111% of correct classification. This results show that the feature selection improves the performances of the classifiers. As it is well known the feature selection process is not simple in biomarker discovery. It looks for a relevant subset of metabolites and the number of metabolites that can be used effectively in the discovery. It requires a powerful machine learning approach for the data type and diagnostic question. The objective of this paper is to propose new machine learning methodology for selection of a subset of metabolic features that improve the classification performances between AD, MCI and CN. Our results showed the power of feature selection methods for the identification of the most important features (metabolites) from large scale metabolomics measurement. Methods based on outranking approaches and on the simple greedy algorithm for feature selection can face the challenges of biomarker discovery in multi-classes AD classification and the method presented here provides an improvement when compared with several standard methodologies. Presented set of markers allows diagnosis of AD, MCI and CN and the biological significance of this set will be further explored.

### References


<table>
<thead>
<tr>
<th>#Features = 8</th>
<th>Accuracy</th>
<th>Error</th>
<th>FP Rate</th>
<th>Sensitivity</th>
<th>Precision</th>
<th>MCC</th>
<th>Kappa</th>
</tr>
</thead>
<tbody>
<tr>
<td>J48</td>
<td>44.444</td>
<td>55.556</td>
<td>0.276</td>
<td>0.489</td>
<td>0.467</td>
<td>0.175</td>
<td>0.1667</td>
</tr>
<tr>
<td>NN(MLP)</td>
<td>48.89</td>
<td>51.11</td>
<td>0.256</td>
<td>0.489</td>
<td>0.52</td>
<td>0.246</td>
<td>0.233</td>
</tr>
<tr>
<td>3-NN</td>
<td>48.89</td>
<td>51.11</td>
<td>0.256</td>
<td>0.489</td>
<td>0.52</td>
<td>0.246</td>
<td>0.233</td>
</tr>
<tr>
<td>RANDOM FOREST</td>
<td>46.667</td>
<td>53.333</td>
<td>0.267</td>
<td>0.467</td>
<td>0.467</td>
<td>0.196</td>
<td>0.2</td>
</tr>
<tr>
<td>NAIVE BAYES</td>
<td>37.38</td>
<td>62.22</td>
<td>0.311</td>
<td>0.387</td>
<td>0.387</td>
<td>0.071</td>
<td>0.067</td>
</tr>
<tr>
<td>SVM</td>
<td>46.667</td>
<td>53.333</td>
<td>0.267</td>
<td>0.467</td>
<td>0.467</td>
<td>0.219</td>
<td>0.2</td>
</tr>
<tr>
<td>BAYESIAN NETWORK</td>
<td>31.11</td>
<td>68.89</td>
<td>0.344</td>
<td>0.311</td>
<td>0.327</td>
<td>-0.028</td>
<td>-0.033</td>
</tr>
<tr>
<td>PROAFTN</td>
<td>46.667</td>
<td>51.111</td>
<td>0.2667</td>
<td>0.4617</td>
<td>0.4617</td>
<td>0.3113</td>
<td>0.1667</td>
</tr>
</tbody>
</table>

TABLE II
Performance of Classification Algorithms for the AD on 8 Selected Features Obtained by Weka Attribute Selection CFSSubsetEval with Greedy Stepwise Method and Using LOOCV

<table>
<thead>
<tr>
<th># Features = 46</th>
<th>Accuracy</th>
<th>Error</th>
<th>FP Rate</th>
<th>Sensitivity</th>
<th>Precision</th>
<th>MCC</th>
<th>Kappa</th>
</tr>
</thead>
<tbody>
<tr>
<td>J48</td>
<td>35.556</td>
<td>64.444</td>
<td>0.322</td>
<td>0.3556</td>
<td>0.378</td>
<td>0.042</td>
<td>0.033</td>
</tr>
<tr>
<td>NN(MLP)</td>
<td>48.889</td>
<td>51.111</td>
<td>0.256</td>
<td>0.489</td>
<td>0.485</td>
<td>0.232</td>
<td>0.233</td>
</tr>
<tr>
<td>3-NN</td>
<td>57.78</td>
<td>42.223</td>
<td>0.211</td>
<td>0.578</td>
<td>0.586</td>
<td>0.371</td>
<td>0.3667</td>
</tr>
<tr>
<td>RANDOM FOREST</td>
<td>40</td>
<td>60</td>
<td>0.3</td>
<td>0.4</td>
<td>0.408</td>
<td>0.103</td>
<td>0.1</td>
</tr>
<tr>
<td>NAIVE BAYES</td>
<td>44.444</td>
<td>55.556</td>
<td>0.278</td>
<td>0.444</td>
<td>0.432</td>
<td>0.163</td>
<td>0.1667</td>
</tr>
<tr>
<td>SVM</td>
<td>48.889</td>
<td>51.111</td>
<td>0.256</td>
<td>0.489</td>
<td>0.489</td>
<td>0.234</td>
<td>0.233</td>
</tr>
<tr>
<td>BAYESIAN NETWORK</td>
<td>51.111</td>
<td>48.889</td>
<td>0.234</td>
<td>0.511</td>
<td>0.37</td>
<td>0.223</td>
<td>0.2667</td>
</tr>
<tr>
<td>PROAFTN</td>
<td>75.556</td>
<td>24.444</td>
<td>0.122</td>
<td>0.7556</td>
<td>0.7556</td>
<td>0.6401</td>
<td>0.45</td>
</tr>
</tbody>
</table>


Sequential Backward Spatio-Spectral Filter Optimization for Motor Imagery Classification in Ear-EEG Brain-Computer Interface

Yong-Jeong Kim  
Dept. of Brain and Cognitive Engineering  
Korea University  
Seoul, Korea  
kyj140511@korea.ac.kr

No-Sang Kwak  
Dept. of Brain and Cognitive Engineering  
Korea University  
Seoul, Korea  
nskwak@korea.ac.kr

Seong-Whan Lee  
Dept. of Brain and Cognitive Engineering  
Korea University  
Seoul, Korea  
sw.lee@korea.ac.kr

Abstract—In recent, an ear-electroencephalogram (EEG) based brain-computer interface (BCI) has been attracting attention in the aspect of practical BCI applications. And most ear-EEG studies have focused on auditory or visual stimuli-based BCI. However, despite advantages of motor imagery (MI), the feasibility of MI-based ear-EEG BCI has not been investigated yet. Hence, in this paper, we examine the usability of MI-based ear-EEG BCI in two folds; 1) observing MI patterns around the ear and 2) classifying MI tasks with satisfactory decoding performance. We propose a sequential backward spatio-spectral filter optimization (SBSSFO) to find a subject-specific optimal frequency band in ear-around EEG signals. And, we evaluate the performance of two datasets including a public dataset (i.e., BCI competition III dataset IVa). In an analysis, event-related (de)synchronization (ERD/ERS) patterns in the ear-around EEG signals showed the smaller intensity than motor-area EEG signals. But the similar patterns were inspected in time and frequency domain. As a result, the proposed method showed an average accuracy of 70.54% with approximately 10.5% and 3.6% performance enhancement compared to the standard common spatial pattern (CSP) and common spatio-spectral pattern (CSSP), respectively. Our experimental results demonstrate the feasibility of the ear-EEG for MI-BCI systems.

keywords—brain-computer interface, ear-EEG, motor imagery, spatio-spectral filtering

I. INTRODUCTION

Brain-computer interface (BCI) is a mind-reading technology that interprets brain signals and controls external devices according to user intentions [1, 2]. And noninvasive brain signal acquisition methods (e.g., functional magnetic resonance imaging (fMRI), electroencephalography (EEG), magnetoencephalography (MEG), functional near-infrared spectroscopy (fNIRS), etc.) have been widely applied in BCI researches [3]. In particular, EEG is more applicable to real-time systems due to high portability and low cost [4-6].

However, EEG-based BCI systems still have some limitations for real-world applications (e.g., wearing an EEG-cap with lots of electrodes, etc.) [7]. Recently, to overcome these problems, ear-EEG electrodes that measure EEG signals around the ear or in the ear have been proposed. The first concept of in-ear electrodes was introduced in [8]. The authors evaluated the quality of the EEG signals. Also, they observed an alpha-attenuation, visual evoked potential (VEP), steady-state visual evoked potential (SSVEP), and auditory steady-state response (ASSR).

However, to the best of our knowledge, ear-EEG studies on motor imagery (MI) have not been reported. Ear-EEG BCI using MI could be useful because it is more natural paradigms than other exogenous paradigms (note that MI does not need any stimulation). Therefore, in this study, we investigate that 1) MI patterns could be detected in the ear-EEG signals, and 2) the MI tasks could be classified with acceptable accuracy.

To this end, we proposed a sequential backward spatio-spectral filter optimization (SBSSFO) algorithm that finds a subject-specific frequency band and spatio-spectral filter simultaneously through a sequential backward selection (SBS) method [9] and a common spatio-spectral pattern (CSSP) method [10].

The outline of the paper is as follows. The proposed method is explained in section II. Experimental protocol and parameter setting for the proposed method were described in section III. Results of classification performance are presented in section IV. Discussion and contribution of this work are provided in section V.

II. SBSSFO

MI signals are mainly activated in the sensorimotor area. However, spatial and spectral characteristics are different depending on the subject and task [11, 12]. Moreover, MI patterns around the ear could be deteriorated spatially and spectrally due to convolutive volume in the brain. Hence, we propose a SBSSFO algorithm to determine a subject-specific frequency band and spatio-spectral filters simultaneously to classify the MI tasks based on ear-EEG signals. The SBSSFO algorithm operates as follows (see Fig. 1 and Algorithm 1).

• Spatio-spectral filter optimization: preprocessing included EEG channel selection, bandpass filtering, and epoching data. In the channel selection, 21 channels on the motor cortex area and 14 (or 8) channels around the ear were

This work was supported by Samsung Research Funding Center of Samsung Electronics under Project Number SRFC-TC1603-02.
subject-specific frequency band for the MI-classification was investigated. In details, the lower cutoff frequency and the higher cutoff frequency of the bandpass filter were determined through the process of updating a lower bound (LB) and an upper bound (UB) based on the classification performance (Algorithm 1). The update process was as follows. If the classification performance of the current frequency band is higher than that of the previous frequency band, the LB is increased by $k_{lo}$, if not, the LB is set to the low cutoff frequency. After that, the higher cut-off frequency is determined through the same process. Note that the UB is decreased by $k_{up}$ in the update process.

### III. Dataset

Two datasets were used to evaluate the proposed method. Dataset1 was obtained through our experiment. And, we employed BCI Competition III dataset IVa as a dataset2.

#### A. Data Acquisition

1. **Dataset1:** Five subjects participated in the experiment. Right hand, left hand, and foot MI tasks were performed in random order (50 times per class). Each subject performed...

---

**Algorithm 1** SBSSFO

1: while true do
2:   while true do
3:     if $performance_{new} > performance_{old}$ then
4:       $band_{lo} \leftarrow band_{lo} + k_{lo}$
5:       $performance_{old} \leftarrow performance_{new}$
6:     else
7:       lower bound = $band_{lo}$
8:       break
9:     if $performance_{new} > performance_{old}$ then
10:    $band_{up} \leftarrow band_{up} - k_{up}$
11:    $performance_{old} \leftarrow performance_{new}$
12:   else
13:    upper bound = $band_{up}$
14:    break
15: end

---

Fig. 2. Timing scheme of experimental paradigm for dataset1.

Fig. 3. Electrode montage for data analysis in the dataset1 (left) and dataset2 (right). Solid red-line means ear-around channels and dotted black-line means motor-area channels used for the classification.
the MI tasks in the following manner (Fig. 2): A fixation cross appeared at the center of the monitor for 0 to 3 s after a warning sound was given. After that, a visual cue (right, left, down arrow) corresponding to each class was randomly assigned for 3 to 7 s. During the 4 s, the subjects performed the MI. And then, a blank screen was displayed for 7 to 10 s. The experiment was conducted during approximately 30 minutes. EEG signals were recorded from 70 Ag/AgCl scalp electrodes following the international 10-20 system (Easy cap, BrainProduct) referenced to nose tip. AFz electrode served as ground, and the data were sampled at 1000 Hz. The experimental protocols were designed using the [14-17] as references.

b) Dataset2: We used BCI Competition III dataset IVa as the dataset2. Five healthy subjects (aa, al, av, aw, ay) participated in the experiment. One hundred eighteen (118) electrodes were used to record the EEG signals based on 10-20 system and sampled at 1000 Hz. Each subject performed 140 trials per each MI task, respectively [18].

B. Pre-processing and Parameter Setting

All EEG data were down-sampled to 100 Hz. The EEG channels from motor area and around the ear were selected for classification. The number of channels for each area was as follows: twenty one (21) for the motor area on both datasets. Fourteen (14) on dataset1, and eight (8) on dataset2 for the ear area (See Fig. 3). Data were segmented ranging from 0.5 to 2.5 s after the visual cue was given. In the SBSFFO method, initial values of the $\text{band}_{lo}$ and $\text{band}_{up}$ were 5 Hz and 35 Hz, respectively, as the frequency band includes the both $\mu$ (8-13 Hz) and $\beta$ (13-30 Hz) band which have known as the most significant frequency band to classify the MI. The increment/decrement parameters ($k_{lo}$, $k_{up}$) in the update process were set to 5 Hz (Algorithm 1 SBSFFO). The value (5 Hz) was selected empirically, considering the trade-off between the performance and the computational cost. Time-delayed embedding varied from 0.01 to 0.15 s [10]. All data were processed using OpenBMI (Matlab toolbox, https://github.com/PatternRecognition/OpenBMI) [19].

IV. RESULT

A. ERD/ERS Pattern

ERD and ERS represent energy decrease or increase phenomenon in a particular frequency band. They could explain and describe activities of MI obviously [13]. Fig. 4 shows the ERD/ERS patterns of the subject ‘ay’ of the dataset 2 in the motor area (C3, Cz) and the ear area (TP7, TP8). A difference of the amplitude between the two tasks appears explicitly in the motor area (upper line), and the difference also appears in the channels around the ear with decreased amplitude. Additionally, Fig. 5 shows time-frequency representations at C3 and TP3 when the right MI is performed, which indicates that the similar phenomenon as that occurring in the motor area occurs in the peripheral channels, though the intensity is less.

B. Correlation

Fig. 6 shows correlations among all EEG channels. The upper graph shows the right-hand MI, and the lower graph shows the foot MI. The marked parts of each graph show the correlation among the EEG channels of the motor area and around the right and left ears, respectively. They represent a different degree of correlation between the two MI tasks. More specifically, when the right-hand MI was performed, the correlation between the motor area of the right-hemisphere and the right-ear area was higher (black rectangle) and when the foot MI was performed, the correlation between the motor area of the left-hemisphere and the left-ear area was higher (green rectangle). And, the correlation among EEG channels in the motor area was stronger when right-hand MI was performed.

Fig. 4. ERD and ERS patterns of the right and the foot MI. Raw EEG signals have been bandpass filtered between 8-13 Hz. And, the envelope of the signals was calculated by the Hilbert transform.
### TABLE I
CLASSIFICATION PERFORMANCE AND SELECTED PARAMETER FOR DATASET 1

<table>
<thead>
<tr>
<th>Method</th>
<th>Area</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Motor</td>
<td>84</td>
<td>88</td>
<td>80</td>
<td>86</td>
<td>93</td>
<td>88</td>
<td>73</td>
<td>85</td>
<td>84.63</td>
</tr>
<tr>
<td>CSP (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>83</td>
<td>86</td>
<td>76</td>
<td>87</td>
<td>92</td>
<td>87</td>
<td>76</td>
<td>70</td>
<td>82.13</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>56</td>
<td>58</td>
<td>44</td>
<td>66</td>
<td>61</td>
<td>74</td>
<td>67</td>
<td>57</td>
<td>60.38</td>
</tr>
<tr>
<td>CSSP (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor</td>
<td>89</td>
<td>93</td>
<td>87</td>
<td>84</td>
<td>93</td>
<td>86</td>
<td>76</td>
<td>90</td>
<td>87.25</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>85</td>
<td>92</td>
<td>87</td>
<td>81</td>
<td>96</td>
<td>88</td>
<td>78</td>
<td>70</td>
<td>84.63</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>63</td>
<td>67</td>
<td>57</td>
<td>73</td>
<td>73</td>
<td>76</td>
<td>76</td>
<td>60</td>
<td>68.13</td>
</tr>
<tr>
<td>Proposed (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor</td>
<td>92</td>
<td>94</td>
<td>90</td>
<td>91</td>
<td>95</td>
<td>91</td>
<td>83</td>
<td>96</td>
<td>91.50</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>91</td>
<td>93</td>
<td>87</td>
<td>82</td>
<td>96</td>
<td>90</td>
<td>86</td>
<td>73</td>
<td>87.25</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>67</td>
<td>71</td>
<td>67</td>
<td>77</td>
<td>77</td>
<td>79</td>
<td>84</td>
<td>62</td>
<td>73.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Area</th>
<th>10 25</th>
<th>5 35</th>
<th>10 35</th>
<th>5 20</th>
<th>5 35</th>
<th>10 35</th>
<th>5 35</th>
<th>10 35</th>
<th>5 35</th>
</tr>
</thead>
<tbody>
<tr>
<td>Band (Hz)</td>
<td>Motor</td>
<td>[10 25]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[5 20]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>[5 20]</td>
<td>[10 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[5 35]</td>
<td>[5 35]</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[10 30]</td>
<td>[5 35]</td>
<td>[5 35]</td>
<td>[5 35]</td>
<td>[5 35]</td>
<td>[5 35]</td>
<td>[5 35]</td>
</tr>
<tr>
<td>Delay-time (s)</td>
<td>Area</td>
<td>0.13</td>
<td>0.07</td>
<td>0.09</td>
<td>0.09</td>
<td>0.07</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.14</td>
</tr>
<tr>
<td></td>
<td>Motor</td>
<td>0.14</td>
<td>0.10</td>
<td>0.08</td>
<td>0.09</td>
<td>0.07</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.09</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>0.08</td>
<td>0.15</td>
<td>0.01</td>
<td>0.09</td>
<td>0.10</td>
<td>0.15</td>
<td>0.11</td>
<td>0.11</td>
<td>0.18</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE II
CLASSIFICATION PERFORMANCE AND SELECTED PARAMETER FOR DATASET 2

<table>
<thead>
<tr>
<th>Method</th>
<th>Area</th>
<th>aa</th>
<th>al</th>
<th>av</th>
<th>aw</th>
<th>ay</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSP (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor</td>
<td>86</td>
<td>95</td>
<td>74</td>
<td>90</td>
<td>95</td>
<td>88.21</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>84</td>
<td>96</td>
<td>76</td>
<td>90</td>
<td>94</td>
<td>88.00</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>54</td>
<td>63</td>
<td>53</td>
<td>60</td>
<td>68</td>
<td>59.71</td>
</tr>
<tr>
<td>CSSP (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor</td>
<td>92</td>
<td>97</td>
<td>75</td>
<td>98</td>
<td>96</td>
<td>91.57</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>90</td>
<td>98</td>
<td>77</td>
<td>97</td>
<td>95</td>
<td>91.29</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>60</td>
<td>67</td>
<td>64</td>
<td>65</td>
<td>73</td>
<td>65.65</td>
</tr>
<tr>
<td>Proposed (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor</td>
<td>93</td>
<td>97</td>
<td>76</td>
<td>97</td>
<td>96</td>
<td>91.64</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>91</td>
<td>97</td>
<td>78</td>
<td>97</td>
<td>95</td>
<td>91.64</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>63</td>
<td>69</td>
<td>66</td>
<td>66</td>
<td>76</td>
<td>68.07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Area</th>
<th>10 30</th>
<th>[5 35]</th>
<th>[15 35]</th>
<th>[10 35]</th>
<th>[5 35]</th>
<th>[10 35]</th>
<th>[5 35]</th>
<th>[10 35]</th>
<th>[10 35]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Band (Hz)</td>
<td>Motor</td>
<td>[10 30]</td>
<td>[5 35]</td>
<td>[15 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[10 35]</td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>[10 30]</td>
<td>[5 35]</td>
<td>[15 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[5 35]</td>
<td>[10 35]</td>
<td>[10 35]</td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td>[15 35]</td>
<td>[5 25]</td>
<td>[10 35]</td>
<td>[10 30]</td>
<td>[10 30]</td>
<td>[10 30]</td>
<td>[10 30]</td>
<td>[10 30]</td>
<td>[10 30]</td>
</tr>
<tr>
<td>Delay-time (s)</td>
<td>Area</td>
<td>0.08</td>
<td>0.08</td>
<td>0.03</td>
<td>0.08</td>
<td>0.08</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor</td>
<td>0.15</td>
<td>0.08</td>
<td>0.09</td>
<td>0.12</td>
<td>0.12</td>
<td>0.01</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motor+Ear</td>
<td>0.08</td>
<td>0.13</td>
<td>0.10</td>
<td>0.03</td>
<td>0.03</td>
<td>0.07</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### C. Performance Evaluation

TABLE I and II show the classification performance for each dataset and the optimal frequency-band for each subject which was determined through the proposed method. In the classification performance of the proposed method, when only the EEG signals around the ear were used, our method showed the accuracy of 73% and 68.07% for each dataset. And, the accuracy was higher than that of the CSP and CSSP in all subjects. (In the 5-fold cross-validation, the same training and test data were used for a fair comparison). The p-values was computed using a paired t-test. The p-values for dataset 1 and 2 are as follows: p=0.0006, 0.0029 (CSP), p=0.0014, 0.0042 (CSSP).

Additionally, in the range of the frequency band which was determined through the proposed method, the frequency band of the ‘Ear’ was the same as or narrower than the frequency band of the ‘Motor’ in most subjects. And, on the average, the classification performance of the proposed method was higher than that of CSP and CSSP in all areas.

### V. Discussion and Conclusion

In this study, the feasibility of ear-EEG based MI-BCI was investigated. We observed that ERD/ERS patterns of the ear-around EEG signals showed the smaller power than that of motor-area EEG signals, but similar patterns were inspected in time and frequency domain. Also, we proposed the SBSSFO algorithm that optimizes a subject-specific frequency band and spatio-spectral filter simultaneously. As a result, the proposed method showed an average accuracy of 70.54% with approximately 10.5% performance enhancement compared to CSP and 3.6% compared to CSSP when using the EEG signals around the ear. However, the proposed method iteratively updates the lower and higher cutoff frequency based on the performance. And, the proposed method investigates the optimal delayed-time in each frequency band. Consequently, the repeated process of calculating the performance could increase the computational cost according to the search range of the delayed-time (τ) and the update values (k_lo, k_up).

In previous studies, various algorithms have been proposed to find an optimal subject-specific frequency band for the MI classification [10-12]. Bayesian spatio-spectral filter optimization (BSSFO) finds optimal frequency bands for each subject.
through the particle-based method [11]. But, the computational cost is high because the frequency band is searched iteratively in the direction of increasing the discriminative power of the generated feature. Regularized spatio-temporal filter and classification (RSTFC) optimizes an subject-specific spatio-spectral filter using continuous delay signals [12]. And, to prevent the overfitting caused by the increase of the dimension of the filter, the L2-norm regularization is applied. Also, feature selection and classification are simultaneously performed by applying the L1-norm regularization to the least square method. But, in the process, many hyper-parameters must be selected. However, our method requires less computational cost and fewer hyper-parameters. Hence, it could be more suitable for real-time applications.

Consequently, our study demonstrates the feasibility of the ear-EEG for MI-BCI systems with the proposed method. In future works, we will evaluate real-time MI-based BCI system using the ear-EEG.

REFERENCES

Finger-vein quality assessment by joint representation learning from grayscale and binary images

Huafeng Qin
Chongqing Engineering Laboratory of detection control and integrated system
Chongqing Technology and Business University
Chongqing, China
qinhuaefengfeng@163.com

Mounim A. El-Yacoubi
Samovar, CNRS, Telecom-SudParis
University Paris Saclay
Paris, France
mounim.el_yacoubi@telecom-sudparis.eu

Abstract—Finger-vein verification is a highly secure biometric authentication that has been widely investigated over the last years. One of its challenges, however, is the possible degradation of image quality, that results in spurious and missing vein patterns, which increases the verification error. Despite recent advances in finger-vein quality assessment, the proposed solutions are limited as they depend on human expertise and domain knowledge to extract handcrafted features for assessing quality. We have proposed, recently, the first Deep Neural Network (DNN) framework for assessing finger-vein quality, that does not require manual labeling of high and low quality images, as is the case for state of the art methods, but infers such annotations automatically based on an objective indicator, the biometric verification decision. This framework has significantly outperformed the existing methods, whether the input image is in grayscale or is binary. Motivated by these performances, we propose, in this work, a representation learning of finger vein image quality, where a DNN takes as input conjointly the grayscale and binary versions of the input image to predict vein quality. Our model allows to learn the joint representation from grayscale and binary images, for quality assessment. The experimental results, obtained on a large public dataset, demonstrates that our proposed method accurately identifies high and low quality images, and outperforms other techniques in terms of equal error rate (EER) minimization, including our previous DNN models, based either on grayscale or binary input.

Index Terms—Finger-Vein Biometrics, Quality Assessment, Joint Representation Learning, Deep Learning

I. INTRODUCTION

In recent years, personal verification using vein patterns, from palm veins [1,2] or finger veins [3-5], has received increasing attention. However, vein verification faces serious challenges because finger-vein image quality is negatively affected by many factors, broadly classified into two categories: 1) Extrinsic quality degradation factors, such as environmental illumination, light scattering, ambient temperature, and user behavior, and 2) Intrinsic quality degradation factors, related to the imperfection of the verification system, such as inaccurate parameter estimation during the image preprocessing stage. An example of the intrinsic factors is related to finger-vein image enhancement and segmentation methods, proposed to extract vein patterns in most finger-vein verification systems. Incorrect estimation of orientation, scale and rotation angle of veins may create false finger-vein patterns and/or fail to detect some genuine ones, resulting in a poor-quality finger vein image, likely to be rejected or misclassified.

If extrinsic and intrinsic factors are not properly dealt with, a lot of low quality images will be produced, leading to an increase in mismatch error. To solve the problem, many approaches [6-9] assume that attributes like the quantity and contrast of vein patterns affect finger-vein image quality, and employ some hand-crafted descriptors to encode them. The quality vein attributes are therein determined by human intuition and a priori knowledge as they are detected by hand-crafted descriptors. As a result, they suffer from the following drawbacks [10]: First, it is difficult to demonstrate that the manually selected attributes are related to image quality. Second, it is impossible to investigate all the attributes affecting image quality. Third, it is not easy to propose a model to describe all such attributes.

Recently, we have proposed the first Deep Neural Network (DNN) for assessing finger-vein quality [10], and actually for any biometrics, based on very limited knowledge. The main issue for applying Deep Learning, in the image quality estimation context, was the inavailability of annotations (labels) on the input quality. We have set, in this regard, a new DNN framework that does not require manual labeling of high and low quality images as it does by state of the art methods, but infers such annotations automatically based on an objective indicator, the verification decision output. This framework has significantly outperformed the existing methods, whether the DNN input image is in grayscale [10] or is binary [11].

Motivated by these performance levels, we propose, in this work, a representation learning of finger vein image quality, where a DNN takes as input conjointly the grayscale and binary versions of the input image to predict its quality. Our
model allows to learn the joint representation from grayscale and binary images, for quality assessment. To do this, we design a new deep learning model that, through two sub-networks, extracts two feature representations, one from the grayscale and the other from the binary input. The latter are then concatenated to form a joint feature layer that is propagated forward through an additional fully connected layer to learn the joint representation in a hierarchical manner.

The experimental results, obtained on a large public dataset, demonstrate that our proposed method accurately identifies high and low quality images, and outperforms other techniques in terms of equal error rate (EER) minimization, including our previous DNN models, based either on grayscale or binary input. The remaining of the paper is as follows. Section II starts by discussing the main motivations and rationale behind our joint representation learning model, and then gives the details of the proposed approach. In section III, we describe our experiments and report the results obtained. In section IV, we conclude the paper.

II. FINGER-VEIN QUALITY ASSESSMENT BY REPRESENTATION LEARNING

Finger-vein quality assessment has been widely investigated by researchers over the last years. In current works, finger-vein quality is usually defined as a measure of the clarity of valleys (veins); based on a visual inspection of the input image, a quality label is assigned manually. However, manual labeling, besides being a tedious and costly operation, suffers from several issues: the human operator has a subjective notion of image quality, which means a low quality image for one person may look as being of good quality to another person; also, the human operator may frequently misjudge the image quality of a finger because he/she is unaware of the internal workings of the biometric system in terms of preprocessing, feature extraction and matching. As the matching error rate is the primary performance indicator, quality assessment algorithms should target its minimization instead of being based on subjective human perception of sample quality. Based on this target, we assumed, in works [10,11], that low quality finger vein images have high probability to be falsely rejected by a state of the art verification system, and proposed an automatic labeling scheme according to this criterion. Compared to manual labeling, our scheme allows to label the image quality in a more objective way, based on biometric performance. Inspired by recent deep learning breakthrough for learning robust hierarchical feature representations from raw image pixels, we proposed a Deep Neural Model to estimate finger-vein quality, by training it on images with quality labels automatically inferred from the verification system.

As our image quality labeling is objectively related to the Equal Error Rate (EER) instead of visual inspection, our DNN models were shown to effectively learn finger-vein features, directly linked to verification accuracy. In [10], the experimental results on two public datasets demonstrated that our DNN model, estimating finger vein quality based on grayscale images, outperformed significantly existing approaches [6,8] in terms of identifying low and high quality images, with a high impact on equal error rate (EER) decrease. The experiments showed also that our model attributed, to several images, quality labels that are not intuitive to human vision and judgment. For example, some images, with rare and low contrast vein patterns, were labeled as being of high quality, but this was coherent with their acceptance by the verification system. By contrast, some images, with rich vein patterns and good contrast, were surprisingly (to human judgment) labeled as poor, but this was coherent with their rejection by the verification system.

Despite these promising findings, our grayscale-based DNN was faced with a serious issue, which is the learning of features related to low and good quality, based on a small set of grayscale images. To improve the performance, we proposed in [11] a DNN to assess quality, based on binary images instead. The rationale is that learning from binary images is easier given the same small dataset, in spite of the imperfections of the binarization algorithm leading to noisy veins, possibly with missing patterns. This was confirmed by the experiments where the binary-based DNN model was shown to significantly outperform the state of the art, including the grayscale-based DNN.

Given the complementary nature of the grayscale and binary based-DNN models, we propose, in this work, a joint representation learning of finger vein image quality, where a DNN takes as input conjointly the grayscale and binary versions of the input image to estimate quality. Our model allows to learn the joint representation from grayscale and binary images for quality assessment. The rationale behind the joint representation model is twofold. Although our estimation of image quality based on representation learning from binary images had outperformed significantly the state of the art [11], we believe there is still room for improvement as the binary images input to DNN are noisy due to the imperfection of the binarization algorithm producing them. Second, inputting to DNN the grayscale images, instead, overcomes this problem, but induces, however, a difficult problem, which is how to reliably estimate image quality based on a small set of images, associated with solely the images rejected by a state of the art verification system. Our proposal seeks to attenuate the imperfections of the binarization in the first scheme and the difficulty of quality estimation from a small set of raw grayscale images in the second, by considering a joint representation of grayscale and binary information sources. In this joint representation, the binary information, in spite of the imperfection of the extracted vein patterns, is intended to guide the DNN where to focus in the grayscale images, for assessing quality. Such guiding information is crucial given the small set of high and low quality the DNN has to learn quality from. In doing so, our approach takes into account both extrinsic and intrinsic degrading factors for quality assessment. For joint representation learning, we design a new deep learning model, that through two sub-networks, extracts two feature representations, one from the gray scale, and the other from the binary input, that allows the model to learn a higher-level
joint representation from the two. Next, we describe briefly the grayscale-based and binary-based DNN models, and then present the architecture of our Deep Neural Network, proposed for joint representation learning.

A. Quality assessment by representation learning from grayscale images

Based on an automatic labeling of gray-scale images by the baseline verification system in [2], we proposed in [10] a Convolutional Neural Network (CNN) to learn the two-class (low vs. high) quality labeling task. The CNN takes as input the gray-scale image, and is then trained to learn deep feature representations to estimate the quality of finger-vein images. The proposed CNN consists of three convolutional layers and two fully connected layers, as shown in Fig.1. In the convolutional layers, Rectified Linear Units (ReLUs) ($y = \max(0; x)$) are employed as the activation functions. Following each of the first two convolutional layers, a max pooling layer is used to extract information robust to small local variations, and to reduce the output dimension. In the fully-connected layers, the dropout technique [12,13] is adopted to prevent overfitting. In the last layer, a Softmax function is employed to estimate the probability distribution over the two classes. We solve the CNN parameter estimation problem by stochastic gradient descent (SGD) with back-propagation as in [12]. The full details are given in [10].

B. Quality assessment by representation learning from binary images

As verification in vein biometrics relies on matching binary vein image pairs, the enhancement and segmentation methods employed to extract (binarize) the vein patterns from the grayscale input can induce further degradation of quality, as their imperfections may result in missing veins or in introducing noise and spurious patterns in the output binary image.

Therefore, in [11], a CNN is trained on binary -instead of grayscale- images for quality assessment. Similar to the model in [10], the CNN for quality assessment, based on binary images, contains three convolutional layers and two max-pooling layers to extract features hierarchically, followed by two fully-connected layers and a softmax output layer providing the probabilities of the two classes. The (ReLUs) activation function, the max pooling and dropout techniques are employed, in a similar way, to optimize training performance.

C. The proposed approach

1) Architecture of CNN: To learn the joint representation for finger-vein quality assessment, we propose the CNN model, shown in Fig.2, that takes a pair of grayscale and binary finger-vein images as input. These two input images are forwarded along two sub-networks consisting, each, of three convolutional layers, two max-pooling layers and one fully-connected layer, respectively. The two fully-connected layers output from the two sub-networks are then concatenated to form a joint feature representation layer, which is further forwarded to a fully-connected layer, followed by the two-class softmax output layer. As shown by the the CNN architecture in Fig.2, the size of each image in the grayscale-binary pair is 20 × 96; the dimensions of the last fully-connected layers in the two sub-networks are fixed respectively to 2000 (grayscale case) and 1000 for binary (binary input). Similar to [10], all the hyperparameters are optimized based on the validation set.

2) CNN-based Joint Representation Learning for Quality Assessment: Given a grayscale finger-vein image $I$, we label its quality label to $q \in \{0, 1\}$ using the method in [2] (the details will be described in section V-B), where 0 and 1 denote respectively low and high quality finger-vein images. Then, the vein network is extracted from the grayscale image using the state of the art segmentation algorithm in [2], and then stored in a binary image $I'$. The training set is denoted by\{($I_1$, $I'_1$, $q_1$), ($I_2$, $I'_2$, $q_2$), ..., ($I_N$, $I'_N$, $q_N$)\}, where $N$ is the number of training images. The DNN, shown in Fig. 2, is trained on this dataset to estimate finger-vein image quality.

III. EXPERIMENTS

A. Database

We have tested the proposed model on a widely used benchmark image dataset, namely the PolyU [2] finger-vein dataset, which includes 3132 images from 156 subjects. 2520 finger-vein images (105 × 2 fingers×6 images × 2 sessions) were captured from the first 105 subjects, in two separate sessions with an average interval of 66.8 days. The remaining 612 images (51 × 2 fingers × 6 images) were acquired from the last 51 subjects in one session. To test our approach, we use the sub-dataset consisting of the 2520 finger images from the first 105 subjects, because it is more realistic, in the biometrics
context, to consider two acquisition sessions instead of one, to account for natural intra-subject and acquisition condition variations through time. The state of the art system in [2] is employed for preprocessing, feature extraction and matching. In our experiments, all images are normalized to a size of 20 × 96 pixels, as shown in Fig.3.

B. Experiment Settings

We employ the experiment settings in [11] to generate the training and test sets, as follows:

*Image quality assumption*: Based on the biometric image quality definition in [14, 15], we assume low quality finger-vein images are those which have high probability to be falsely rejected by the verification system.

*Template selection*: For each finger image, we compute its average distance score with respect to the other 11 images from the same finger, based on the baseline verification system described in [2]. The image with the lowest average score is then set as the finger’s template. Considering that different fingers from the same subject can be treated as different classes, the dataset consists of 210 (2 fingers × 105 persons) templates, and 2310 (210 × 11) query images.

*Labeling high and low quality images*: We determine the quality label of each query image based on the matching distance against its template. Overall, there are 210 × 11 = 2310 genuine match scores, obtained by matching each template against the remaining 11 finger-vein images, and 210 × 209/2 = 21945 impostor match scores, obtained by matching each subject’s template against the 209 templates from the other subjects (symmetric impostor matches are not considered). Based on genuine and impostor matches, the False Rejection Rate (FRR) and False Acceptance Rate (FAR) are computed. The FAR refers to the system security level in a biometric verification system. At the predefined system security level (normally, FAR = 0.1 %), we set label \( q = 0 \) to a query finger-vein image that is falsely rejected by the system. Otherwise, we set the label \( q \) to 1. Note that 0 and 1 denote low and high quality finger-vein images respectively.

*Generating the training and test sets*: After labeling each query finger-vein image, we select half the fingers (first 105) associated with the 1155 grayscale images (105 × 11) for training, and the images from the remaining fingers for test. This process yields 101 low-quality and 1054 high-quality grayscale images in the training set, while 110 low-quality and 1045 high-quality grayscale images are obtained for the test set. To increase the gap between high-quality and low-quality training images, the 406 samples with highest quality are selected, from the 1054 grayscale images above, as good quality images for training. In general, low-quality images are fewer than high-quality ones in biometric systems. To overcome the class-imbalanced problem, we generate additional low-quality images in the following way. First, 305 images are selected from the remaining 648 (1054-406) high-quality training images. We then change their illumination, scale and rotation angle so that the corresponding binary images cannot match anymore the corresponding templates in the verification system [2]. We thus obtain a low-quality image set of 406 (305 + 101) grayscale images for training, ensuring in this way, class balance. Finally, we extract the vein network from each grayscale image in the training and test sets, using the state of the art method in [2], and build the binary image-based training and test sets.

### Table 1

<table>
<thead>
<tr>
<th>Different approaches</th>
<th>High quality image Accuracy(%)</th>
<th>Low quality image Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hand-craft+SVM[8]</td>
<td>68.80</td>
<td>66.36</td>
</tr>
<tr>
<td>Radon transform[6]</td>
<td>71.96</td>
<td>67.27</td>
</tr>
<tr>
<td>CNN+grayscale image</td>
<td>84.59</td>
<td>83.64</td>
</tr>
<tr>
<td>image[10]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CNN+binary image</td>
<td>88.13</td>
<td>88.18</td>
</tr>
<tr>
<td>image[11]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The proposed approach</td>
<td>89.28</td>
<td>90.00</td>
</tr>
</tbody>
</table>

C. Evaluation of image quality assessment

In our first experiment, we estimate the performance of various methods in terms of identifying high and low quality images. The Radon transform [6] and the method named "hand-crafted features + SVM" [8] are tested for benchmarking. The considered hand-crafted features are gradient, contrast, and information capacity as shown in [8]. In addition, we show the performance of our previous DNN-based quality assessment methods [10,11], based respectively on grayscale and binary images. To simplify the description, we denote them as CNN+grayscale and CNN+binary, respectively. Table 1 reports the accuracy of high/low quality identification. As shown, the worst DNN scheme (grayscale-based) outperforms the best handcrafted-feature based system ([6]) by more than 12% and 16% in identifying high and low quality images respectively. CNN+binary adds about 4% improvement. This show the power of our automatic quality labeling scheme based on DNN. The better performance of CNN+binary w.r.t CNN+grayscale may be explained by the less difficult task of learning robust feature representations from binary images, when the training set is small. Finally, we observe that the proposed DNN approach, learning a joint representation from the grayscale and binary inputs, provide the best performance overall, as it identifies high quality images and low quality images with an accuracy of 89.28% and 90% respectively, which are higher than all the other methods, with a relative error reduction of 3% and 15% over the best of them, the CNN+binary model. This highlights the fact that, despite the
small size of the training set, the grayscale information can be useful to improve quality estimation. This may be explained by the fact that grayscale images are not subject to the degrading artefacts resulting from the binarization algorithm, and thus the joint representation from the grayscale and binary inputs can increase the modeling power of the composite DNN model by using the binarized vein patterns as guiding information in learning better quality feature representations of quality from the gray scale image.

The second experiment is to show how much the finger-vein verification performance [2] can be improved by adopting the proposed quality assessment scheme. Before the filtering, we have 105 × 11 = 1155 genuine matches and 105 × 104/2 = 5460 impostor matches for computing the FRR and FAR, respectively. When the templates are selected based on the proposed method, the EER of the finger-vein verification system [2] is 4.97% for the test image set acquired over two sessions. Using a filtering mechanism harnessing our image quality assessment, we can enhance the reliability of the verification system by automatically rejecting the images labeled as being of low quality by the quality assessment model.

Each quality assessment approach usually outputs, given a query finger vein image, a score between 0 (low quality) and 1 (high quality). A threshold of 0.5 allows the simple setting of the two-class discrimination problem, and provides, accordingly, a fixed number of detected low and high quality images. By changing the threshold, however, we can allow each assessment system to provide a variable number of low and high quality vein images. To evaluate the impact of automatic low quality image detection by each approach on EER decrease, we vary the assessment threshold of each approach, so as to allow them to label different percentages of query images as being of low quality, e.g., 0% (no quality assessment), 1%, 2%,... up to 30%. By rejecting these percentages of low quality images in the verification phase, we can assess which quality assessment methods have the best impact on minimizing the EER. For 5%, for instance, the best methods would be those that discard mostly the 5% lowest quality images, i.e., those most likely to cause a verification mismatch. Fig. 4 displays the EERs of the finger-vein verification system after filtering low quality query images, by the various approaches discussed earlier, at different percentages. The experimental results (Table 1 and Fig.4) show that the proposed scheme outperforms existing approaches in terms of the impact of identifying high/low quality images on reducing the EER. We see that at each rejecting level, given the same percentage of discarded images by all the approaches, our proposed model discards the images with lowest quality, which explains the lowest error rate obtained at each level. Such a good performance can be explained by the fact that our proposed joint representation learning model tackles both extrinsic and intrinsic factors that degrade finger-vein image quality, and learns robust joint binary-grayscale feature representations that better capture what a good/poor quality image is w.r.t to the internal workings of a state of the art verification system.

IV. CONCLUSION AND PERSPECTIVES

In this paper, we proposed a composite deep learning model that learns a joint feature representation from raw grayscale and binary finger-vein images, to assess the captured image quality. Thanks to these two image sources, our model harnesses the noisy but simple information conveyed by the extracted binary veins to guide the model to better learn robust quality features not only from the binary input but also from the associated grayscale image. Our model outperforms the current best state of the art quality assessment model, our CNN+binary model proposed in [11], which shows the complementarity of binary and grayscale information sources, for automatically learning image quality, especially in experimental settings where the training datasets are small. This ability is key as small datasets for quality assessment are a hallmark of not only vein biometric systems, but of biometric systems in general, since the number of poor quality images is usually low irrespective of the biometrics under study.

In the future, we intend to explore further the ability of our joint representation learning model in tackling quality-related tasks, without the need of human annotations. One direction is to investigate how to recover missing vein patterns due to acquisition or binarization issues. We have developed, in this regard, a system that does recover the missing veins from binary input [16], but as our study shows, it will be useful to explore the same recovery problem, but based on the grayscale images directly.

REFERENCES


Failure modelling and anomaly detection of a propulsion subsystem

Catherine Cheung
National Research Council
Aerospace
Ottawa, Canada
cathy.cheung@nrc-cnrc.gc.ca

Julio J. Valdés
National Research Council
Digital Technologies
Ottawa, Canada

Richard Salas Chavez
National Research Council
Aerospace
Ottawa, Canada

Alejandro Lehman Rubio
National Research Council
Aerospace
Ottawa, Canada

Abstract—In this work, the sensor data related to a diesel engine system and specifically its turbocharger subsystem was analyzed. An incident where the turbocharger seized was recorded by the dozens of standard turbocharger-related sensors. By training models to distinguish between normal healthy operating conditions and deteriorated conditions, there is an opportunity to develop prognostic and predictive tools to ideally help prevent a similar occurrence in the future. Analysis of this event provides an opportunity to identify changes in equipment indicators with a known outcome. A number of data analysis tools were used to characterize the healthy and deteriorated states of the turbocharger system, including various supervised classification and unsupervised anomaly detection techniques. This paper describes the results of this modelling process, validated by testing on healthy data from the same propulsion system and a second distinct one. Although this problem posed challenges due to the severely imbalanced class distribution, promising results were nonetheless obtained.

Index Terms—classification, anomaly detection, propulsion system, failure modelling, condition indicators

I. INTRODUCTION

The amount of sensors installed for the purpose of equipment health monitoring in engines, aircraft, and vehicles has increased steadily in the digital age. There can be thousands of sensors in these sensor networks to monitor both the operator input and associated equipment outputs in addition to sensor data. Developing strategies and capabilities to extract useful information from the tremendous amounts of data collected is a challenge that can be used to establish the indicators of system health, a necessary precursor to the implementation of condition based maintenance, that could be explored through the use of data analytics tools and methods.

In previous work [1], initial analysis of the sensor data related to a diesel engine system and specifically its turbocharger subsystem was carried out. An incident involving seizure of the turbocharger was captured by the sensor data, and hence analysis of this event provides an opportunity to identify changes in equipment indicators with a known outcome. The data surrounding this event was analysed using several data analysis tools [1], including the transformation of the original high-dimensional sensor data to a low-dimensional space.

Those results showed that by mapping the 31 sensor measurements to a 3-D space, the data structure could be readily inspected visually. The resulting plots of the data structure in 3-D showed a clear departure from normal behaviour in the turbocharger following the seizure. This was seen as a chain of objects that rapidly differentiated from the rest of the data structure.

In this paper, supervised classification models and unsupervised anomaly detection methods have been used in continued efforts to characterize the healthy and failed states of the turbocharger system and to identify the change in behaviour of the system during that transition. This paper is organized as follows: Section II provides details of the turbocharger data and the system state excursion, Section III describes the methods and tools used for data analysis, Section IV details the data pre-processing steps and experimental settings, Section V presents the results of the classification and anomaly detection models, and Section VI summarizes the findings.

II. TURBOCHARGER

The focus of this work is the turbocharger, which is one of the subsystems in a medium-speed diesel engine system. The diesel engine is air-charged by twin turbochargers, Turbo A and Turbo B, with intermediate air-cooling [1].

A. Turbocharger seizure

This work looks at the data recorded by the sensor system related to a turbocharger seizure suffered by an A Bank turbocharger on a particular vehicle. The sensor system data for the engine was available for a considerable length of time before and after the incident.

The subsequent turbocharger incident investigation described a chain of events that led to the turbocharger seizure with key timings (hh:mm) detailed below:

- Loss of Turbo A speed reading, engine shut down
- Speed sensors between Turbo A and B switched
- Engine restarted, still no Turbo A speed reading (01:12)
- Diesel engine engaged (01:41)
- Speed setting increased (01:42), engine exhaust temperatures increased to alarm without any speed increase (01:43 - 01:44)
- Diesel engine disengaged and shut down (01:44 - 01:45)
Inspection of the turbocharger indicated that the speed sensor installed when the Turbo A and B speed sensors were interchanged had inadequate clearance to the thrust collar of the turbine, which caused the seizure. In other words, the installation of the speed probe caused rubbing against the turbine and prevented it from turning. This caused the turbocharger to seize. Although the cause of the failure was not due to gradual deterioration of a system component, the data analysis is focused on characterizing the healthy and failed states of the turbocharger system.

B. Turbocharger data

The sensor system aboard this particular vehicle comprises thousands of sensors associated with both the operator input and associated equipment outputs, and sensor data. The sensor system data is recorded at rates up to 2 Hz. From the thousands of sensor signals recorded in the sensor network, those associated with the diesel engine were identified, consisting of about 238 sensor measurements. From this engine signal subset, those related to the turbocharger as well as other important signals were selected, reducing the subset to 31 sensor signals. These 31 sensor measurements include turbocharger speeds, inlet, outlet and exhaust temperatures, pressures, and shaft torque as described in Table I.

The 2 months surrounding the turbocharger seizure were analyzed, from a month prior to a month after the event. However, following the incident the engine was not in operation while undergoing repair. In addition to the 2 months surrounding the turbocharger seizure, other healthy engine data was acquired to assist in evaluating the models. Healthy engine data from the same engine was analyzed, as well as healthy engine data from a different engine. In all these data sets, “healthy” data corresponds to the absence of maintenance actions raised for the system during that period.

III. ANALYTICAL TECHNIQUES

The approach adopted for data analysis was aimed at characterizing the healthy and failed states of the turbocharger system. To better identify any data which does not fit to the common behaviour of the system, a set of anomaly detection techniques was attempted to suggest possible abnormal states of the system. These outliers could represent possible current or future system failures, or just highlight uncommon system behaviours that could be of interest to the operator. Two methods for anomaly detection were attempted, in particular One-class Support Vector Machines (SVM) and Isolation Forest (IF). These methods are unsupervised methods.

Supervised classification of the “healthy” and “failed” states of the turbocharger was also carried out. In this analysis, the data labelled as “healthy” and “failed” was provided to classifiers to distinguish one class from the other. The classification methods used included k-nearest neighbours, SVM, neural networks, and Random Forest (RF).

The accuracy metrics for a classification problem include true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). Table II describes these terms as part of a confusion matrix, summarizing the output of a classifier. A perfect classifier will have elements along the main diagonal only. The sum of all elements correspond to the total number of samples evaluated. In this work, a true positive is defined as a failed event predicted as a failed event, while a true negative corresponds to a healthy event predicted as a healthy event. False positives or false alarms should be minimized, since if too frequent it could lead to complacency in operator response. False negatives also need to be minimized and avoided if possible so that any system failure is not overlooked. For prediction of failure states in a mechanical system, it is important that not only is the classification accurate but also that false alarms are minimized.

A. Anomaly detection methods

Anomaly or outlier detection mechanisms are unsupervised processes that identify items, events or observations which do not follow an observed common structure of the data. The definition of such common data structure will depend on the initial parameters given to the mechanism, making it more or less stringent when finding anomalous or outlying data. For this analysis, the Isolation Forest approach and one-class support vector machine (SVM) methods were used.

Isolation Forest (IF) [2], [3] builds an ensemble of so called iTrees (a binary tree structure) for a given dataset. Anomalies are considered those instances which have short average path lengths on the iTrees. There are two training parameters and one evaluation parameter: the number of trees to build and the subsampling size. The evaluation parameter is the tree height limit during evaluation. The detection accuracy of IFs converges rapidly with a very small number of trees. Moreover, they only require a small subsampling size to achieve high detection accuracy and are highly efficient. The
different height limits are used to look for anomaly clusters of different density. Among their distinguishing features are 

i) they exploit subsampling to a much larger extent compared to other methods, 

ii) they do not make use of distance or density measures to detect anomalies, with the corresponding reduction in computational cost compared with distance-based and density-based methods, 

iii) IFs have a linear time complexity with minimal memory requirements (the algorithm has constant training time), something not shared by other techniques, and 

iv) Isolation Forests have the capacity to scale up to handle extremely large data size and high dimensional problems in terms of the number of descriptor variables, possibly containing a large number of irrelevant attributes.

The basic idea is that of isolation, which measures an individual susceptibility to be isolated from the rest of the data. The rationale is to randomly generate binary trees where instances are recursively partitioned. In the case of anomalous instances, these trees will have considerable shorter paths because on the one hand, in the areas containing anomalies, fewer anomalies will result in a reduced number of partitions. On the other hand, instances with distinguishable attribute-values are more likely to be isolated early in the partitioning process. If for such a collection of randomly generated trees systematically shorter path lengths are found for some specific points, it is very likely that they will be anomalies.

In order to have a reliable outlier detection mechanism, the number of false alarms or false positives must be minimized by adjusting, or fine-tuning, the parameter settings of the Isolation Forest function. To obtain such optimal parameter setting values, a genetic algorithm was used for its global optimization capabilities. In simple terms, the genetic algorithm is an optimization technique that dynamically selects the input parameters to the Isolation Forest based on the fitness of previously obtained results, inspired by the process of natural selection [4]. The defined fitness function consisted of maximizing the number of true negatives. The fine-tuning process consisted of the selection of the Isolation Forest mechanism, the definition of the significant input parameters to the function, the defined fitness function, and the optimal input parameter values as obtained by the genetic algorithm. This fine-tuning was performed on the training data.

One-class SVM [5] is a useful tool for outlier detection. SVMs are normally used to implicitly classify a dataset between a set of detected inherent classes in the data. By restricting the SVM to only look for one class, it analyzes the data and identifies what a normal behaviour consists of. Thus, any outlier values will be highlighted. One-class SVM tries to learn a rough, close frontier of the “normal” states. If new observations fall in this frontier they are considered normal, otherwise they are labelled as outliers.

B. Classification methods

A number of supervised classification methods were also attempted to build models to distinguish unhealthy turbocharger data from healthy turbocharger data. In these methods, a representative training set was given to the model with the training data labelled as “healthy” or “failed”. A model was trained on this data to achieve high overall classification (true positives and true negatives) accuracy and then tested on an unseen testing set. Several classification models were attempted using k-nearest neighbours, support vector machine, neural networks, and Random Forest methods.

K-nearest neighbour (kNN) classifiers [6], [7] is a simple classifier that is based on the data structure and distribution of the labelled training data. Given a point to be classified, \(x_0\), the \(k\) training points \(x^{(r)}, r = 1, ..., k\) closest in Euclidean distance to \(x_0\) are identified, and then using majority vote among the \(k\) neighbours the point is classified as one class or another [8].

Support vector machines (SVM) can also be used for classification [9], [10], where nonlinear boundaries can be constructed to separate classes by first transforming the feature space where linear boundaries separating the classes can be constructed [8].

Neural networks (NN) were initially developed as models for the human brain, with each node in the network representing neurons in the brain and the connections between nodes corresponding to synapses. Single hidden-layer backpropagation neural networks, also called single layer perceptrons, can be used for both regression and classification in a two-stage model. Neural networks with multiple hidden-layers, or multi-layer perceptrons (MLP) [11], are also commonly used for classification and regression. For classification of \(K\) classes, there are \(K\) output nodes in the output layer corresponding to 0 – 1 values of each node for each of the \(K\) classes [8].

The Random Forest (RF) method [12] is a modified version of bagging or bootstrap aggregation where large forests of de-correlated trees are constructed and then averaged [8]. By averaging many noisy but approximately unbiased models, the concept behind bagging, the variance can be reduced. Trees can learn complex interactions in data structure and have low bias if the trees are grown deep. However, trees are infamous for being noisy, so averaging over many trees in a forest helps to reduce the noise and variance.

These classification methods were chosen for several reasons. kNN was the baseline, since as a non-parametric method it does not assume a distribution of the training data, such as Naïve Bayes. SVM was used since the visualization of the data in prior work [1] showed that the healthy and failed data had distinct geometries which indicated that SVM could probably find an accurate decision boundary. Random Forest and multi-layer perceptron were used for their ability to deal with high dimensional data, as well as prior work showed that tree-based methods and neural networks succeeded in similar classification problems.

C. Clustering

In order to facilitate analysis of the data, samples of the total objects were built using the leader clustering algorithm [13] in which representative objects (leaders) are chosen to ensure that for every object in the dataset there is at least
one leader sufficiently close (in distance) or similar (given a similarity measure). The algorithm can produce a sample that is structurally compatible with the whole dataset, thus ensuring that odd or outlying elements will not be left unaccounted in the sample, as may occur with conventional random sampling. Here, the dissimilarity measure was Euclidean distance.

IV. EXPERIMENTAL SETTINGS

A. Data pre-processing

As the sensor network data was not originally designed to be used for maintenance or safety applications but for real-time equipment monitoring, the recorded data was in need of processing and consolidation before data analysis could be performed. Each signal was extracted individually from the full database and then unknown and erroneous readings were removed. Afterwards, each signal was linearly interpolated and sampled at one-minute intervals for the desired time range, ensuring that the time range fell within the interpolated values. The table was filtered to include only data corresponding to the diesel engine being activated.

B. Training and testing sets

Training of the classifier and anomaly detection models was carried out using the turbocharger data in the 2-month period surrounding the turbocharger seizure, as described in Section II. With the data down-sampled to 1-minute intervals, there were 9969 data points in that period with 22 of those points labelled as “failed” points based on the timing of the turbocharger seizure. These failed points include all of the data points after the seizure as they correspond to data related to the seized subsystem. All of the data points preceding the seizure event were labelled as “healthy” data points.

The testing of the classification models was carried out on turbocharger data obtained from a different time period, where the turbocharger was considered healthy as there were no active maintenance actions raised on the subsystem. Data from the turbocharger from a second propulsion system from an entirely different vehicle was also used to test the models. The data from this second system was also considered healthy and labelled as such. Table III summarizes the training and testing data used for anomaly detection and classification.

For the anomaly detection models, the training data supplied was 85% of the healthy data from the 2-month period described above selected randomly. Testing of the anomaly detection models was carried out on the remaining 15% of the healthy data from that same 2-month period as well as on all of the failed points following the seizure.

All the input variables in the training data were standardized, converting them to z-scores so that all variables had a mean of zero and a standard deviation of one. By ensuring that all variables have the same unit variance after standardizing, the influence of each variable in similarities, distances, etc. is the same. The means and standard deviations from the training data were then used to pseudo-standardize the testing data so that the models could be applied to the unseen testing data.

C. Experimental Settings

For one-class SVM, a non-linear radial basis function (RBF) kernel was used with the kernel coefficient, $\gamma = 0.1$. The upper bound on the fraction of training errors and the lower bound of the fraction of support vectors was given by $\nu = 0.15$, to avoid overfitting.

For isolation forest and the genetic algorithm used to fine-tune the IF parameters, 136 base estimators formed part of the ensemble. The number of samples to draw from $X$ to train each base estimator was $maxsamples = 0.7926$. The contamination fraction used was 0.1400. Default parameters of 10 maximum features were used, corresponding to the number of features to draw from $X$ to train each base estimator.

The $k$NN classifier consisted of 3 neighbours with uniform weights so that all points in each neighbourhood are weighted equally. Euclidean distance was used as the distance metric.

The architecture of the NN-MLP consisted of 3 hidden layers (15 nodes, 10 nodes, 5 nodes). A rectified linear activation was used and the optimization function was $lbfgs$ (limited-memory Broyden-Fletcher-Goldfarb-Shanno [14]) which is an optimizer in the family of quasi-Newton methods. An $L_2$ penalty of 0.0001 was used for regularization, and maximum of 200 iterations with convergence criteria (tolerance of $1e^{-4}$).

For Random Forest, the number of trees in the forest was 10 and the Gini impurity was used to measure the quality of the split. The minimum number of samples required to split an internal node was 2 and to be at a leaf node was 1.

For the SVM classifier, again a non-linear RBF kernel was used. The class weights were balanced to account for the natural imbalance of the data. A penalty parameter, $C = 100$ was used, which controls the margin of the hyper-plane and with a relatively large value it helps minimize the number of misclassified instances by choosing a smaller-margin hyper-plane. A shrinking heuristic was used and the stopping criterion was $1e^{-3}$ tolerance without a limit on number of iterations.

V. RESULTS

A. Clustering results

From the original data including the failure period there were 9969 data points. To reduce the size of the data, clustering techniques were applied, as described in Section III-C. Several similarity thresholds were attempted resulting in varying number of leaders to represent the original data, as detailed

<table>
<thead>
<tr>
<th>Classification</th>
<th>Training</th>
<th>Testing data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>System 1</td>
<td>System 1</td>
</tr>
<tr>
<td>Total points</td>
<td>9969</td>
<td>36000</td>
</tr>
<tr>
<td>Failed points</td>
<td>22</td>
<td>0</td>
</tr>
<tr>
<td>Healthy points</td>
<td>9947</td>
<td>36000</td>
</tr>
</tbody>
</table>

### TABLE III

TRAINING AND TESTING DATA

<table>
<thead>
<tr>
<th>Anomaly Detection</th>
<th>Training data</th>
<th>System 1</th>
<th>Testing data</th>
<th>System 1</th>
<th>System 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total points</td>
<td>8454</td>
<td>1492</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failed points</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Healthy points</td>
<td>8454</td>
<td>1492</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
in Table IV. Both the original data set and these four cases of leaders were used to train the classifier models.

B. Classification results

Failure modelling was carried out by training several classifier models on the data in the period leading up to the turbocharger failure. The original data sampled at 1-minute intervals was evaluated, as well as the 4 different cases of leaders (1551 leaders, 1079 leaders, 509 leaders, and 314 leaders) as detailed in Table IV. Testing was carried out on healthy data from the same system (system 1) during a different period of time. Testing was also carried out on healthy data from a separate propulsion system (system 2). The results of the four classifier methods, k-nearest neighbours (kNN), SVM, neural network (NN-MLP), and Random Forest (RF), are presented in the following tables as confusion matrices: Table V for the original data, Table VI for the 1551 leaders, Table VII for the 1079 leaders, Table VIII for the 509 leaders, and Table IX for the 314 leaders. An ideal classifier would have elements on the diagonal and zero values in the off-diagonal for training. Since the testing data here only consists of healthy data, the ideal classifier would have all elements equal to zero except for the bottom right element, true negatives, corresponding to labelled healthy and classified healthy entry.

An interesting trend can be found when comparing the original results to the results using leaders. For the original data, the NN-MLP and RF are both able to learn the decision boundary and are able to classify the new data accurately.
The anomaly detection results on the original data showed good performance. These methods are unsupervised, as the models are not trained on outlying or failed data. Determination of anomalies is based on the distribution of the data. A strong result here is that both methods were able to correctly identify almost all of the failed data points (20 of 22) in the testing data as outliers. Both methods had very similar performance determining the healthy data as inliers. However, the misclassification rate, specifically the number of false positives, is concerning and needs to be investigated further. Perhaps, more data cleaning needs to be performed.

### VI. Concluding Remarks

In this work, sensor data from a diesel engine was analyzed using a number of supervised classification and unsupervised anomaly detection techniques. A sudden seizure of a turbocharger in a propulsion system provided the “failed” system data to distinguish from the normal healthy operating data. While the sudden nature of the turbocharger incident was not a typical example of a slowly deteriorating mechanical system, the opportunity to build models with a known outcome was still valuable.

With a challenging imbalanced class distribution, where there were only 22 “failed” data samples among the almost 10000 “healthy” data samples, promising results were obtained using both the classifier models and the anomaly detection models. Especially promising was the success achieved by the classification models when tested on data from a different propulsion system from a different vehicle demonstrating transferability of the models from one system to another.

Future efforts are directed at expanding the scope of the input parameters to build more general models for the propulsion system. Further analysis is also planned to identify and better understand the key input variables required for making the distinction between the healthy and failed system states.

### References


Oral Session I

Deep learning and applications

Wednesday May 16, 2018, 10:00 AM

Zilong Hu, Jinshan Tang, Ping Zhang and Babu Patlolla
Identification of Bruised Apples Using Deep Learning and 3D Near-infrared Imaging

Guoqiang Zhong, Hongxu Wei, Yuchen Zheng, Junyu Dong and Mohamed Cheriet
Deep Error Correcting Output Codes

Mohammad Ali Bagheri Orumi, Mahmoud Famouri, Zohreh Azimifar and Azadeh Nazemi
Deep Learning-Based Corresponding Points Fast Matching

Pengcheng Xi, Rafik Goubran and Chang Shu
Cardiac Murmur Classification in Phonocardiograms using Deep Convolutional Neural Networks

Prerana Mukherjee, Brejesh Lall and Snehlith Lattupally
Object cosegmentation using deep Siamese network
Identification of Bruised Apples Using Deep Learning and 3D Near-infrared Imaging

Zilong Hu  
School of Technology  
Michigan Technological University  
Houghton, United States  
E-mail: zilongh@mtu.edu

Jinshan Tang*  
School of Technology  
Michigan Technological University  
Houghton, United States  
E-mail: jinshant@mtu.edu

Ping Zhang  
Department of Mathematics and Computer Science  
Alcorn State University  
Mississippi, United States  
E-mail: pzhang@alcorn.edu

Babu Patlolla  
Department of Biological Sciences  
Alcorn State University  
Mississippi, United States  
Email: bpatlolla@alcorn.edu

Abstract—In this paper, we propose an algorithm for recognizing bruised apples based only on surface shape information obtained by a 3D near infrared (NIR) imaging system. The proposed algorithm is composed of two parts: construction of feature map, and classification of apples into bruised or unbruised categories. We propose a new algorithm to code 3D shape information into a 2D feature map. For classification, we propose to build a convolutional neural network to extract deep hierarchical features from the 2D feature maps that are optimal for the identification of bruised apples. Experimental results show that the proposed algorithm is better than the algorithm developed previously, which indicates the potential of the proposed algorithm for the identification of bruised apples.

Keywords— Bruised apples, 3D meshes, feature extraction, convolution neural network, deep learning, identification

I. INTRODUCTION

In a study, it was found that bruised apples were found in around 16% of harvests apples with hand-picking method, and this number increased when mechanical pickers were applied [1]. Therefore, identification of bruised apple is commonly used in the production line of fruits to improve the quality of fruits served to the market. Manual identification of bruised apples has many limitations, such as high time cost, affected by human bias, and so on [2]. In order to overcome such problems, many efforts have been done to develop automatic bruise detection systems. Among those systems, image processing technologies are commonly used to identify the bruised apples [3, 4]. Most proposed methods are using 2D imaging technology, however, the average accuracy of the current identification approaches are from 62% for Red Delicious [5], to 82% using NIR infrared spectroscopy [6]. Besides, applying 2D imaging technology also introduces new limitations, including low accuracy, sensitive to the lighting condition as well as the viewpoint of the camera, and incapable of measuring depth information of bruise regions.

3D imaging technologies have become more attractive in recent years. Compared with 2D technologies, 3D imaging technologies can collect accurate shape information for any type of the objects that it scans. In addition, they can obtain depth information which may contribute to bruise grading. Moreover, the 3D imaging systems are more user-friendly due to its insensitivity to the viewpoint of the camera as well as lighting conditions. The 3D near infrared (NIR) imaging technology with certain wavebands is found to be harmless to human beings and foods. This technology has been applied for 3D human tissue measurement [7]. All of those advantages of 3D near infrared imaging technology make itself perfectly suited for bruise detection on fruits. Therefore, in this paper, we propose a fruit bruise identification algorithm based on 3D mesh data obtained by NIR imaging.

Convolutional Neural Networks (CNN) are one type of deep learning architectures, which extract hierarchical features through multiple convolutional layers to learn the deep representation of image data [8, 9]. CNNs have shown its state-of-art performance in ImageNet bench-mark [10], and many of their variants have been proposed in recent studies for object recognition, segmentation [11, 12]. In our algorithm, we design a CNN, which is based on AlexNet, to identify bruised apples from 3D mesh data.

The rest of paper is organized as follows: Section 2 introduces the construction of feature map; Section 3 introduces our design of the CNN classifier; Section 4
presents experimental results; and Section 5 concludes the paper.

II. CONSTRUCTION OF FEATURE MAP

A. Computation of vertex curvatures

Curvature is often used as feature descriptors in 3D shape analysis [13]. Different types of surface curvatures are applied to measure the shape of the surface curve at a particular vertex [14]. Many studies have been done to develop methods to compute curvatures [15-17]. We used the method proposed by Rusinkiewicz. The algorithm is based on the tensor averaging method for the curvature calculation, and it is able to produce accurate result in major types of meshes and having lower outlier estimations [13]. We used the above method to obtain four types of curvatures: (1) mean curvature, denoted as the mean of the principle curvatures; (2) Gaussian curvature, denoted as the product of the principle curvatures; (3) maximum curvature, denoted as the largest value in principle curvatures; and (4) minimum curvature, denoted as the smallest value in principle curvatures. Fig. 1 shows an example of the surface of a bruised apple represented by mean curvatures calculated using the methods above, the values of curvatures are represented using different colors.

B. Convert vertices from 3D Coordinate system to 2D coordinate system

An important requirement of applying CNN for image classification is that the size of data fed into the network is fixed. It is easy to fulfill in matrix data like image array. However, it is difficult to directly feed triangular meshes to CNNs for training owing to the irregular distribution of vertices of meshes. In addition, because triangular meshes do not have implicit ordering, it become extremely difficult to execute pooling process on mesh data. Therefore, we propose to convert vertices from the 3D Cartesian coordinate system into a 2D coordinate system. The procedure details are shown as follows:

Step 1: Compute the geometric center of the mesh, subtract center coordinates from all vertices coordinates so that the new geometric center become the origin of 3D coordinates (see Fig. 2 (a)).

Step 2: Convert the vertices from 3D Cartesian coordinate system into a spherical coordinates system, where each point is represented by the azimuth angle, the elevation angle, and the radius. In this paper, we define the azimuth angle as the counterclockwise angle in the z-x plane measured from the positive y-axis; the elevation angle as the angle from the z-x plane; and the radius as the distance from the origin to the vertex.

Step 3: Use the azimuth angle and the elevation angle to construct a new 2D coordinate system, where the azimuth angle is used as the x-axis, and elevation angle is used as the y-axis (see Fig. 2 (b)).

Instead of directly converting vertices from the 3D coordinate system to the 2D coordinate system, our method is aiming at preserving the feature information of meshes as much as possible. After we establish the 2D coordinate

---

Fig. 1. Illustration of different types of curvatures in the 3D surface of a bruised apple. (a) 2D color image of a bruised apple; (b) corresponding 3D mesh; (c) Mean curvature; (d) Gaussian curvature; (e) Maximum curvature; and (f) Minimum curvature.

Fig. 2. Illustration of generating 2D feature maps based on mean curvatures from the mesh of a bruised apple. (a) presents the original mesh data in the 3D coordinate system; (b) presents the converted vertices of meshes in the 2D coordinate system; (c) presents the interpolation result on scatted data in (b), the intensity value at each query point is represented using different RGB colors; (d) presents the generated 2D feature maps after post-processing.
system, we map the curvature values computed in 3D coordinate system to the coordinate 2D system. The map is based on the following assumption: each pair of azimuth angle and elevation angle corresponds to only one vertex with coordinates \((x,y,z)\) in the 3D coordinate system. Thus, the curvature for a specific pair of azimuth angle and elevation angle in the 2D coordinate system is equal to the curvature value of the vertex which the pair correspond to.

The converted points are irregularly distributed on the 2D plane (see Fig. 2 (b)). Therefore, we generated 2D feature maps through scattered data interpolation by taking curvature features as the intensity values of points on the 2D plane. The interpolation procedure is shown as follows:

**Step 1:** Triangulate the points in the 2D plane;
**Step 2:** Locate the triangle that encloses the query point;
**Step 3:** Compute the intensity value of the query point through the weighted sum of the intensity values of the three vertices in the enclosing triangular (linear interpolation) (see Fig. 2 (c)). The weight is determined by the distance between the triangular vertices and the query point.

After we perform interpolation, we need some post-processing. Post-processing includes extracting regions of interest (ROI) from feature maps which is defined as a 300x300 pixel region ranges from -1.5 to 1.5 in x- and -1.5 to 1.5 in y-axis, and normalizing the curvature values of an ROI to [0, 1] (see Fig. 2 (d)). Because all four types of curvatures are used to generate the feature maps, we combine all feature maps together to obtain a 4-channel 300×300 volume array for each apple data. Two examples of the constructed 4-channel feature array of a bruised apple and an unbruised apple are shown in Fig. 3.

### III. CLASSIFICATION USING DEEP LEARNING

As one type of deep learning architectures, convolutional neural networks (CNN) use convolutional filters in each convolutional layer to extract deep hierarchical features from the image data. Combined with pooling layer, the total number of parameters need to be learned is reduced dramatically, which makes them more effective and more competitive in real applications [18-20].

![Fig. 4. Schematic diagram of the proposed CNN. The network contains 5 convolutional layers (Conv), 3 fully connected layers (FC), 7 batch normalization layers (Bnorm), 7 activation layers (ReLU), 3 max pooling layers (Mpooling), and a softmax layer.](image)

#### A. Mathematical concept

Given an image dataset \(X\), and the class labels of the dataset \(C\), the goal of a CNN is to find a mapping function \(F(X)\) that minimize the loss function \(L\), which is defined as [17]:

\[
L = \sum_{i \in X} I(F(I_i), c_i)
\]

where \(I_i\) denotes the image data of object \(i\) in \(X\), \(c_i\) denotes the corresponding class label, and \(l\) denotes the penalty function, e.g. softmax loss. In general, \(F(I_i)\) can be represented with

\[
F(I_i) = f_N(f_{N-1}(... (f_1(I_i))))
\]

where \(N\) represents the number of hidden layers within the model, \(f\) represents the function in the corresponding layer. Main types of layers in a CNN include convolutional layer, activation layer, and pooling layer (fully connected layer can be regarded as a special type of convolutional layer where the size of the filter is 1x1). In a convolutional layer, \(f\) is a linear function, it is represented as:

\[
f(x) = W \ast x + b
\]

where \(W\) denotes the weights in the convolutional filters, and \(b\) denotes the bias term. In the activation layer, \(f\) is a non-linear function, it can be represented using different non-linear functions, e.g. Rectified linear units (ReLU) : \(f(x) = max(x, 0)\). In the pooling layer, \(f\) is a non-linear downsampling function, common types of pooling functions include average pooling and max pooling. Optimization of a CNN is achieved by modifying the values of \(W\) and \(b\) in each convolutional layer during back propagation, and stochastic gradient descent method is commonly used for the approximation of \(W\) [21].

#### B. Network design

The proposed CNN is designed based on the structure of AlexNet [8]. In the proposed algorithm, we modify AlexNet by changing the size of convolutional filters in the first convolutional layer to 11×11×4, and changing the output of
the last fully connected layer to 2. In addition, we introduce batch normalization layers in the CNN in order to eliminate covariance shift [22], and remove the dropout layer from the AlexNet. Fig. 4 illustrates the schematic diagram of the proposed CNN. In specific, the new CNN has five convolutional layers and three fully connected layers. Batch normalization layer is added right after each convolutional layer (layer 1 to layer 5), and right after the fully connected layer 6 and layer 7. Each batch normalization layer is followed by a ReLU activation layer. ReLU activation layers after the first, second, and fifth convolutional layers are followed by a max pooling layer Due to the small size of the dataset. The batch size used in the proposed algorithm is 32. We choose the number of epoch as 60, the momentum is set to 0.9, the weight decay is set to $5 \times 10^{-4}$, the learning rate is set to decay from $10^{-1}$ to $10^{-6}$ logarithmically.

IV. EXPERIMENTAL RESULTS

The proposed algorithm was implemented using Matlab 2016a, the implementation of the CNN was achieved using the MatConvNet library [23]. All experiments were performed under Windows 10 on a machine with CPU Intel Core i7-4720HQ @ 2.6HZ, GPU NVIDIA GeForce GTX 960M and 16GB of RAM.

The dataset used in the experiments was 3D mesh data taken from the front view of 200 bruised apples and 102 unbruised apples. The mesh data for each apple had around 100,000 vertices. In order to reduce the computational cost and improve the processing efficiency, we down-sampled the mesh data so that each mesh has around 10,000 vertices. The down-sampling of mesh data was implemented using iso2mesh toolbox in Matlab [24].

For classification, owing to the proposed CNN has the input size of 227x227x4, we resize the constructed feature arrays to the same size. 10-fold cross validation was used to evaluate the performance of the proposed algorithm. We divided the whole dataset into 10 subsets, each subset contained 10% (20) of bruised apples and 10% (10) of unbruised apples. Note that because the dataset contains 102 unbruised apples in total, we assigned 12 unbruised apples into the last subset. Each time we took 1 subset for testing, and other 9 subsets for training.

In the first experiment, we explored the effect of epoch number on the performance of the proposed CNN. Fig. 5 shows the boxplot of experimental results at epoch 1, 5, 10, 15, 20, 25, 30, 35, 40, 50, 60. It can be found from the figure that in early epochs the median values (denoted as red lines) diverged from the middle of boxes representing the first and third quartile values, red cross symbols indicate outliers. In addition, the upper and lower boundary of boxes were dropping with the increase of epochs. It indicates that in early epochs CNN was trained to decrease error rate, but still performed poorly in identification. When the value of epoch was larger than 20, median values started converging to the middle of boxes and finally stabilized around the middle. We also calculated the mean average of error rate in each epoch, and plotted it in Fig. 5. Illustration of cross validation results using boxplot as well as mean average. The blue line showed the decreasing of mean error rate with the increase of epochs and stabilized at 10.03% when the epoch number is larger than 35. The final cross validation result show that the identification accuracy of our proposed algorithm was 89.67%, which proved its potential in applications of identifying bruised fruits. In the second experiment, we compared the proposed algorithm with the other method for the identification of bruised fruit we developed previously. The method, denoted as SVM-Mesh, was using a non-linear support vector machine (SVM) classifier with 2nd polynomial kernel to train on local binary pattern (LBP) features directly extracted from the mesh data by using Mesh-LBP method proposed in [25]. For SVM-Mesh, we chose mean curvature, interpolation parameter was set to 6, and the 1st ordered facet ring to generate LBP features from meshes [25]. The proposed algorithm was denoted as CNN-Mesh. Table 1 shows the 1st quartile, 3rd quartile, median, and mean of recognition rate in cross-validation result of two algorithms. The mean identification accuracy of two algorithms are: 89.67% for CNN-Mesh and 85.33% for SVM-Mesh. The result indicated that compared with SVM with LBPs, the CNN has better performance in learning bruise features from the dataset.

V. CONCLUSION

In this paper, we proposed an algorithm to identify bruised apples from a 3D near infrared imaging system. The

Table 1. Bruise apple identification accuracy of two algorithms.

<table>
<thead>
<tr>
<th></th>
<th>CNN-Mesh</th>
<th>SVM-Mesh</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st quartile</td>
<td>83.33%</td>
<td>80.00%</td>
</tr>
<tr>
<td>median</td>
<td>90.00%</td>
<td>86.67%</td>
</tr>
<tr>
<td>3rd quartile</td>
<td>96.67%</td>
<td>90.00%</td>
</tr>
<tr>
<td>mean</td>
<td>89.67%</td>
<td>85.33%</td>
</tr>
</tbody>
</table>
algorithm is composed of two parts: construction of feature maps and classification. For feature map construction, we proposed to transfer the vertices from 3D coordinates to spherical coordinates, and use angle information to construct a new 2D coordinate system. Four types of vertex curvatures were used to represent the shape information of the 3D meshes and they were converted into a 4-channel image array. For classification, we designed a new CNN based on AlexNet to fit our dataset. In the experiments, we used 10-fold cross validation to evaluate the performance of the proposed CNN. The experimental results showed that the identification accuracy of bruised apples was achieved by 89.67%, indicating the potential of the proposed algorithm in the applications of bruised fruit detection.

VI. ACKNOWLEDGEMENT

This research project was sponsored by USDA/NIFA grant: An Automatic System for Bruise Detection on Tomatoes and Apples Using 3-D Near Infrared Imaging Technology. Award No: 2014-38821-22394.

REFERENCES

Deep Error Correcting Output Codes

Guoqiang Zhong\(^1\)*, Hongxu Wei\(^1\), Yuchen Zheng\(^2\), Junyu Dong\(^1\) and Mohamed Cheriet\(^3\)
\(^1\) Department of Computer Science and Technology, Ocean University of China,
238 Songling Road, Qingdao 266100, China.
Email: gzghong@ouc.edu.cn, weihongxu@stu.ouc.edu.cn, dongjunyu@ouc.edu.cn.
\(^2\) Department of Advanced Information Technology, Kyushu University,
744 Motoooka Nishi-ku, Fukuoka 819-0395, Japan.
Email: yuchen@human.ait.kyushu-u.ac.jp.
\(^3\) Synchromedia Laboratory for Multimedia Communication in Telepresence,
École de Technologie Supérieure, Montréal H3C 1K3, Canada.
Email: mohamed.cheriet@etsmtl.ca.

Abstract—In this paper, we present a novel deep ensemble learning algorithm, called deep error correcting output codes (DeepECOC), which adapts traditional ECOC models to deep architectures. DeepECOC are composed of multiple layers of the ECOC module that combines multiple binary classifiers for feature learning. Unlike previous deep networks, such as deep autoencoders, supervisory information can be used during the greedy layer-wise pre-training of DeepECOC. We have conducted extensive experiments to compare DeepECOC with traditional ECOC, feature learning and deep learning algorithms on several benchmark data sets. The results not only show the advantages of DeepECOC over previous related approaches, but also demonstrate the effectiveness of supervised greedy layer-wise pre-training in contrast to unsupervised counterpart.

Index Terms—Error correcting output codes, deep learning, deep architecture.

I. INTRODUCTION

Error correcting output codes (ECOC) are an ensemble learning framework to address multi-class classification problems [1]. A recent work shows that the ECOC methods can also be used for feature learning, in either a linear or a nonlinear manner [2]. However, although sophisticated coding and decoding strategies are applied [3]–[5], the learnability of ECOC is limited by its single-layer structure from a network perspective. Therefore, to exploit the advantages of the ECOC framework, such as supervised ensemble learning and effective coding design, it’s necessary to combine its ideas with that of deep learning.

Since about a decade ago, many deep learning models have been developed in kinds of applications, such as image classification, object detection, document recognition, natural language processing, and video analysis [6]–[12]. Meanwhile, desirable performances in such applications have been achieved. Among others, [6] presents the ground-breaking deep autoencoder that learns the weight matrices by pre-training the stacked restricted Boltzmann machines (RBMs) and fine-tuning the weights using gradient descent. It delivers much better representations of data than shallow feature learning algorithms, such as principal components analysis (PCA) [13] and latent semantic analysis (LSA) [14]. In order to boost the traditional autoencoder and prevent the “over-fitting” problem, [15] introduces the denoising autoencoder that corrupted the data with a random noise. Recently, most of the research focuses on deep convolutional neural networks (CNNs) and recurrent neural networks (RNNs), which greatly improves the state-of-the-art in the areas of object recognition, unsegmented handwriting recognition and speech recognition [7], [16], [17]. However, existing deep networks are generally initialized with unsupervised methods, such as random assignments and greedy layerwise pre-training. In the case of random initialization, to obtain good results, many training data and a long training time are generally used; while in the case of greedy layerwise pre-training, as the whole training data set needs to be used, the pre-training process is very time-consuming and difficult to find a stable solution.

To overcome the limitations of both traditional ECOC methods and deep learning models, and meanwhile, take advantages of both of them, in this paper, we propose a novel deep learning model called deep error correcting output codes (DeepECOC). DeepECOC are composed of multiple stacked ECOC modules, each of which combines multiple binary classifiers for feature learning. Here, the weights learned for the binary classifiers can be considered as weights between two successive layers, while the probabilistic outputs of the combined binary classifiers as the outputs of a hidden layer or new representations of data. On the one hand, the ECOC modules can be learned layer by layer using the given supervisory information, and on the other hand, based on the ternary coding design, some classes of data are automatically neglected when training the binary classifiers, such that the weights are learned only using part of the training data. Hence, the supervised pre-training of DeepECOC is in general very effective and efficient. We have compared DeepECOC with traditional ECOC, feature learning and deep learning algorithms to demonstrate the effectiveness and superiority of DeepECOC. The results are reported in Section [V].

The rest of this paper is organized as follows: In Section [II] we give a brief overview to related work. In Section [III] we present the proposed model, DeepECOC, in detail. The experimental results are reported in Section [IV] while Section [V]
concludes this paper with remarks and future work.

II. RELATED WORK

Traditional ECOC framework has two steps: coding and decoding. In the coding step, an ECOC matrix is defined or learned from data, and the binary classifiers are trained based on the ECOC coding; in the decoding step, the class label is given to a test sample based on a similarity measure between codewords and outputs of the binary classifiers. The widely used coding strategies include one-versus-all (OneVsAll) [18], one-versus-one (OneVsOne) [19], discriminant ECOC (DECOC) [20], ECOC optimizing node embedding (ECOCONE) [21], dense and sparse coding [22], [23], and so on. Among them, the OneVsAll, OneVsOne, dense and sparse coding strategies are problem-independent, whilst the DECOC and ECOCONE are problem-dependent. Generally, the length of the codeword by the OneVsAll, OneVsOne, DECOC and ECOCONE coding designs is related to the number of classes, but by the dense and sparse coding design is relatively flexible. In this work, we design the structure of DeepECOC based on the properties of each coding strategy. The commonly used binary ECOC decoding strategies are the Hamming decoding [18] and Euclidean decoding [19]. For ternary ECOC decoding strategies, the attenuated Euclidean decoding [24], loss-based decoding [23], and probabilistic-based decoding [25] are widely used. Currently, the state-of-the-art ternary ECOC decoding strategies are the discrete pessimistic beta density distribution decoding and loss-weighted decoding [1]. In this work, for the simplicity of back propagation, we directly add a Softmax layer at the top of DeepECOC for the decoding. Note that, although many coding and decoding strategies have been proposed and applied in recent years [3]–[5], from the viewpoint of connectionism, the learnability of ECOC is limited by its single-layer structure. To further exploit the advantages of ECOC, such as supervised ensemble learning and effective coding design, it’s necessary to update them to deep architectures.

There are some pieces of work that attempt to construct a deep architecture with multiple feature learning methods [6], [26]–[29]. For instance, deep autoencoder is built up by RBMs [6], and deep semi-NMF combines multiple steps of matrix factorization [26]. Similarly, deep CNNs and RNNs can also be considered as deep models that learn the new representations of data layer by layer [7], [16], [17]. The success of these existing models demonstrate that deep networks are beneficial to the representation learning tasks, especially for the large scale applications. However, existing deep learning models are generally initialized with unsupervised methods, such as random assignments and greedy layerwise pre-training, which result in a long training time of the deep models. In this work, we propose the DeepECOC model, which is based on the stacked ECOC modules. When pre-training DeepECOC, the ECOC modules can be learned with the available supervisory information. Intuitively, as this manner of supervised pre-training has deterministic objective, the learned value of the parameters will be very close to the best local minimum on the solution manifold. Experimental results for multi-class classification tasks shown in Section IV also demonstrate this fact.

III. DEEP ERROR CORRECTING OUTPUT CODES (DeepECOC)

In this section, we first introduce the traditional ECOC framework, which is the important building block of DeepECOC. Then we present the learning procedures of DeepECOC in detail.

A. The ECOC Framework

ECOC combine multiple binary classifiers to solve multi-class classification problems. In their coding step, the ECOC coding matrix \( M \in \{-1,1\}^{C \times L} \) (binary case) or \( M \in \{-1,0,1\}^{C \times L} \) (ternary case) is first defined or learned from the training data, where each row of \( M \) is the codeword of a class, each column corresponds to a dichotomizer (binary classifier), \( L \) is the length of the codewords (the number of binary classifiers), \( C \) is the number of classes, symbol ‘1’ indicates positive class, ‘-1’ indicates negative class, and ‘0’ indicates that a particular class is not considered by a given classifier. Then, the binary classifiers (dichotomizers) are trained according to the partition of the classes in the columns of \( M \). Fig. 1 shows two coding matrices encoded with the one-versus-all (binary case) and one-versus-one (ternary case) coding strategies.

![Two coding matrices encoded with the one-versus-all (binary case) and one-versus-one (ternary case) coding strategies.](image)

![Fig. 1. Two coding matrices encoded with the one-versus-all (binary case) and one-versus-one (ternary case) coding strategies.](image)

In order to take the probabilistic outputs of the base classifiers as new representations of data, we adopt support vector machines (SVMs) as the binary classifiers in this work. Applying a decoding strategy on the outputs of the binary classifiers, the ECOC framework can be used for multi-class classification, while applying the sigmoid function on the values of the discriminant function, ECOC can be used for feature learning [2]. This is the key foundation of the proposed DeepECOC model in this paper.
B. DeepECOC

To combine the advantages of ECOC and deep learning algorithms, we build the DeepECOC architecture as follows

\[
x \xrightarrow{q_D} \tilde{x} \xrightarrow{\mathbf{W}_1} h_1 \xrightarrow{\mathbf{W}_2} h_2 \ldots \xrightarrow{\mathbf{W}_{n-1}} h_{n-1} \xrightarrow{\text{softmax}} y, \tag{1}
\]

where the first step makes the clean input \( x \in [0, 1]^d \) partially destroyed by means of a stochastic mapping \( \tilde{x} \sim q_D(x | x) \). In the corrupting process, we set a parameter called denoising rate \( \nu \). For each input \( x \), a fixed number elements, \( v_D \), of \( x \) are chosen at random, and their value is forced to be 0, while the others are left untouched. This operation makes the model more robust and prevents the overfitting problem in most cases \cite{15}. Subsequently, the “corrupted” data are taken as inputs for the DeepECOC model. \( \mathbf{W}_1 \) and \( b_1 \) are the weight matrix and bias vector learned from the first ECOC module. The output of the first hidden layer is denoted as

\[
h_1 = s(\mathbf{W}_1^T x + b_1), \tag{2}
\]

where \( s(\cdot) \) is the sigmoid activation function \( s(x) = \frac{1}{1 + e^{-x}} \). From the second layer to the \((n - 1)\)-th layer, we use the stacked ECOC modules to learn the weight matrices and biases, which can be considered as weights between two successive layers of a deep network. Similarly, we use the output of the \((k - 1)\)-th layer as the input of the \( k \)-th layer,

\[
h_k = s(\mathbf{W}_k^T h_{k-1} + b_k). \tag{3}
\]

Here, \( h_k \) can be viewed as an activation output and a new representation of the input data \( x \).

For example, if we adopt the OneVsAll coding strategy for one layer of the ECOC module, we first define the coding matrix \( \mathbf{M}_C \times C \), where \( C \) is the number of classes. Then, we can train \( C \) SVM classifiers to obtain the weight matrix \( \mathbf{W} = \{\mathbf{w}_1, \ldots, \mathbf{w}_C\} \) and the bias \( \mathbf{b} = \{b_1, \ldots, b_C\} \). Next, we calculate the output of the first layer by using Eq. (2).

After the pre-training step, we use back propagation \cite{30} to fine tune the whole architecture. Moreover, we also employ a technique called “dropout” for regularization \cite{31}. When a large feedforward neural network is trained on a small training set, dropout generally performs well on the test set. The basic idea of dropout is that each hidden node is randomly omitted from the network with a probability of \( \beta \). In another view, dropout is a very efficient way to perform model averaging with neural networks. Through these processes, we finally obtain the DeepECOC model, which is robust and easy to be applied to multi-class classification tasks.

Note that, compared to existing deep learning algorithms, DeepECOC have some important advantages. Firstly, unlike previous deep learning algorithms, DeepECOC are built with the ECOC modules and pre-trained in a supervised learning fashion. Secondly, if we adopt ternary coding strategies, due to the natural merit of ECOC, the weights can be learned using only part of the training data. Thirdly, in contrast to the learning of the weight matrices in previous deep learning models, the binary classifiers in each ECOC module can be learned in parallel, which may greatly speed up the learning of DeepECOC.

### IV. Experiments

To evaluate the effectiveness of the proposed method, DeepECOC, we conducted 4 parts of experiments. In the first part, we present the following results.

#### TABLE I

<table>
<thead>
<tr>
<th>Problem</th>
<th>( \eta )</th>
<th>Epoch</th>
<th>Problem</th>
<th>( \eta )</th>
<th>Epoch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dermatology</td>
<td>0.1</td>
<td>2000</td>
<td>Yeast</td>
<td>0.01</td>
<td>4000</td>
</tr>
<tr>
<td>Iris</td>
<td>0.1</td>
<td>400</td>
<td>Satimage</td>
<td>0.01</td>
<td>4000</td>
</tr>
<tr>
<td>Ecoli</td>
<td>0.1</td>
<td>2000</td>
<td>Letter</td>
<td>0.01</td>
<td>8000</td>
</tr>
<tr>
<td>Wine</td>
<td>0.1</td>
<td>2000</td>
<td>Pendigits</td>
<td>0.01</td>
<td>2000</td>
</tr>
<tr>
<td>Glass</td>
<td>0.01</td>
<td>4000</td>
<td>Segmentation</td>
<td>0.01</td>
<td>8000</td>
</tr>
<tr>
<td>Thyroid</td>
<td>0.1</td>
<td>800</td>
<td>Optdigits</td>
<td>0.01</td>
<td>2000</td>
</tr>
<tr>
<td>Vowel</td>
<td>0.1</td>
<td>4000</td>
<td>Shuttle</td>
<td>0.1</td>
<td>2000</td>
</tr>
<tr>
<td>Balance</td>
<td>0.1</td>
<td>4000</td>
<td>Vehicle</td>
<td>0.1</td>
<td>4000</td>
</tr>
</tbody>
</table>

### V. Classification Accuracy and Standard Deviation Obtained by DeepECOC and the Compared Approaches on 16 UCI Data Sets

The best results are highlighted in boldface.

#### TABLE II

<table>
<thead>
<tr>
<th>Problem</th>
<th>Single</th>
<th>AE</th>
<th>DAE</th>
<th>DeepECOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dermatology</td>
<td>0.9513</td>
<td>0.9429±0.0671</td>
<td>0.9674±0.0312</td>
<td>0.9702±0.0354</td>
</tr>
<tr>
<td>Iris</td>
<td>0.9600</td>
<td>0.9600±0.0562</td>
<td>0.9333±0.0889</td>
<td>0.9600±0.0535</td>
</tr>
<tr>
<td>Ecoli</td>
<td>0.8147</td>
<td>0.7725±0.0608</td>
<td>0.8000±0.0362</td>
<td>0.8529±0.0403</td>
</tr>
<tr>
<td>Wine</td>
<td>0.9605</td>
<td>0.9765±0.0264</td>
<td>0.9563±0.0422</td>
<td>0.9875±0.0264</td>
</tr>
<tr>
<td>Glass</td>
<td>0.6762</td>
<td>0.6669±0.1032</td>
<td>0.6669±0.0715</td>
<td>0.7895±0.0788</td>
</tr>
<tr>
<td>Thyroid</td>
<td>0.9210</td>
<td>0.9513±0.0614</td>
<td>0.9599±0.0567</td>
<td>0.9656±0.0513</td>
</tr>
<tr>
<td>Vowel</td>
<td>0.7177</td>
<td>0.6985±0.0745</td>
<td>0.7101±0.0756</td>
<td>0.7475±0.0991</td>
</tr>
<tr>
<td>Balance</td>
<td>0.8222</td>
<td>0.8036±0.0320</td>
<td>0.8268±0.0548</td>
<td>0.9137±0.0412</td>
</tr>
<tr>
<td>Yeast</td>
<td>0.5217</td>
<td>0.5641±0.0346</td>
<td>0.5891±0.0272</td>
<td>0.5959±0.0599</td>
</tr>
<tr>
<td>Satimage</td>
<td>0.8537</td>
<td>0.8675±0.0528</td>
<td>0.8987±0.0304</td>
<td>0.8961±0.0480</td>
</tr>
<tr>
<td>Letter</td>
<td>0.9192</td>
<td>0.9234±0.0547</td>
<td>0.9318±0.0641</td>
<td>0.9532±0.0534</td>
</tr>
<tr>
<td>Pendigits</td>
<td>0.9801</td>
<td>0.9831±0.0123</td>
<td>0.9886±0.0034</td>
<td>0.9908±0.0031</td>
</tr>
<tr>
<td>Segmentation</td>
<td>0.9701</td>
<td>0.9584±0.0317</td>
<td>0.9596±0.0211</td>
<td>0.9711±0.0286</td>
</tr>
<tr>
<td>Optdigits</td>
<td>0.9982</td>
<td>0.9785±0.0101</td>
<td>0.9856±0.0088</td>
<td>0.9867±0.0096</td>
</tr>
<tr>
<td>Shuttle</td>
<td>0.9988</td>
<td>0.9953±0.0012</td>
<td>0.9976±0.0014</td>
<td>0.9988±0.0021</td>
</tr>
<tr>
<td>Vehicle</td>
<td>0.7315</td>
<td>0.6987±0.0521</td>
<td>0.7348±0.0454</td>
<td>0.7561±0.0480</td>
</tr>
<tr>
<td>Mean rank</td>
<td>2.7813</td>
<td>3.4063</td>
<td>2.6563</td>
<td>1.1563</td>
</tr>
</tbody>
</table>
we compared DeepECOC with some deep learning models and single-layer ECOC approaches on 16 data sets from the UCI machine learning repository\(^1\). In the second part, we compared DeepECOC with traditional feature learning models, some deep learning models and single-layer ECOC approaches, and tested DeepECOC with different number of hidden layers, on the USPS handwritten digits\(^2\). In the third part, we used the MNIST handwritten digits\(^3\) to further demonstrate the effectiveness of DeepECOC for handwritten digits recognition. Finally, the CIFAR-10 data set\(^1\) was used to demonstrate the effectiveness of DeepECOC on image classification tasks. For all the data sets, the features were normalized within \([0, 1]\). In the following, we report the experimental results in detail.

A. Classification on 16 UCI Data Sets

In these experiments, we compared DeepECOC with autoencoder (AE)\(^6\), denoising autoencoder (DAE)\(^15\) and single-layer ECOC approaches (Single)\(^3\). We built DeepECOC with the ECOC optimizing node embedding (ECOCONE) coding method\(^21\). Here, we initialized ECOCONE with one-versus-one coding strategy. In addition, the state-of-the-art linear loss-weighted (LLW) decoding strategy was used for the decoding of ECOCONE. Finally, a structure with 3 hidden layers was adopted for DeepECOC, where the denoising rate and dropout rate were both set to 0.1:

\[
x \xrightarrow{q_0} x \xrightarrow{w_{1}} h_1 \xrightarrow{w_{2}} h_2 \xrightarrow{w_{3}} h_3 \xrightarrow{\text{softmax}} y. \quad (7)
\]

For the fine-tuning process, we used the stochastic gradient descent algorithm. The learning rate and epochs for different data sets are described in Table I. The autoencoder and denoising autoencoder’s architectures are as same as DeepECOC with ECOCONE initialized by one-versus-one for fair comparison. For single-layer ECOC approaches, we chose the best results shown in [3] as the baseline results. For DeepECOC, we used SVMs with RBF kernel function as base classifiers. The SVMs were implemented with the LIBSVM toolbox and their parameters were set to default\(^32\).

We used 10-fold cross validation to evaluate the performance of DeepECOC and the compared approaches. Table I\(^1\) shows the obtained average classification accuracy and standard deviation on the 16 UCI data sets. We conducted the Friedman and the Nemenyi test\(^33\) with confidence level 0.05 on the results presented in Table I\(^1\). Statistical comparison showed that the performance difference between DeepECOC and the compared approaches was significant. This demonstrates the effectiveness of DeepECOC.

B. Classification on the USPS data set

The USPS handwritten digits data set includes 7291 training samples and 2007 test samples from 10 classes. The size of the images is \(16 \times 16 = 256\). Our experiments on this data set were divided into 2 parts. Firstly, we compared DeepECOC with two traditional feature learning models (principal components analysis (PCA)\(^34\) and marginal Fisher analysis (MFA)\(^35\), autoencoder (AE), denoising autoencoder (DAE), LeNet\(^36\), PCANet\(^37\) and single-layer ECOC approaches. Here, PCA is an unsupervised method, and MFA is a supervised method. For MFA, the number of nearest neighbors for constructing the intrinsic graph was set to 5, while that for constructing the penalty graph was set to 15. For DeepECOC, we used ECOCONE (initialized by one-versus-one) as the ECOC coding strategy. We used batch gradient descent for the fine-tuning process, the batch size was set to 100, the learning rate was set to 1, the number of epoch was set to 40000, while the denoising rate and dropout rate were set to 0.1, respectively. We used SVMs with RBF kernel function and default parameters as base classifiers. For single-layer ECOC approaches, we adopted ECOCONE (initialized by one-versus-one) as coding design method and the linear loss-weighted (LLW) decoding strategy. For the LeNet model, we used 2 convolutional layers, two pooling layers and two fully connected layers. The kernel size of the convolutional layers and pooling layers was set to \(2 \times 2\), the stride was set to 1, the number of nodes of the first layer was set to 200, the epoch was set to 8000, the initial learning rate was set to 0.001, and the momentum was set to 0.9. For the learning rate policy, we set it to “inv”, i.e.

\[
r_i = r_0 \times (1 + \gamma \times i)^{-m}, \quad (8)
\]

where \(r_i\) was the learning rate of the \(i\)th iteration, \(r_0\) was the initial learning rate, \(\gamma\) was set to 0.0001, and \(m\) was set to 0.75. For the PCANet model, we used two PCA-filter stages, one binary hashing stage and one blockwise histograms. The filter size, the number of filters, and the block size were set to \(k_1 = k_2 = 3\), \(L_1 = L_2 = 4\), and \(7 \times 7\), respectively. The experimental results are shown in Tab. III.

From Tab. III\(^1\) we can see that DeepECOC achieved the best result than other methods include traditional feature learning models, existing deep learning methods and single-layer ECOC approaches.

In the second part of our experiments, we evaluated DeepECOC with different number of hidden layers. We used 2 to 6 hidden layers in our experiments. The parameter settings were as same as the above part. Fig. 2\(^2\) shows the experimental results. We can see that DeepECOC obtained the best result

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LeNet</td>
<td>0.8062</td>
</tr>
<tr>
<td>PCANet</td>
<td>0.9502</td>
</tr>
<tr>
<td>AE</td>
<td>0.9422</td>
</tr>
<tr>
<td>DAE</td>
<td>0.9481</td>
</tr>
<tr>
<td>PCA</td>
<td>0.8844</td>
</tr>
<tr>
<td>MFA</td>
<td>0.9023</td>
</tr>
<tr>
<td>Single</td>
<td>0.9043</td>
</tr>
<tr>
<td>DeepECOC</td>
<td><strong>0.9601</strong></td>
</tr>
</tbody>
</table>

\(^1\)http://archive.ics.uci.edu/ml/
\(^2\)http://www-i6.informatik.rwth-aachen.de/~keysers/usps.html
\(^3\)http://yann.lecun.com/exdb/mnist/
\(^4\)http://www.cs.toronto.edu/~kriz/cifar.html
when using 3 hidden layers. When the number of hidden layers is less than 3, the effectiveness of DeepECOC increases with the increasing of the number of hidden layers. Along with the number of hidden layers continues to grow, the effectiveness of DeepECOC decreases a little. However, the performance of DeepECOC is generally robust.

C. Classification on the MNIST data set

MNIST handwritten digits data set has a training set of 60,000 examples, and a test set of 10,000 examples with 784 dimensional features. We designed 2 architectures for autoencoder, denoising autoencoder and DeepECOC. The first architecture was $784 - Z_1 - Z_2 - Z_3 - 10$, where $Z_i$ was the number of hidden neurons designed based on some ECOC coding strategies. We designed this architecture because we wanted to make autoencoder and denoising autoencoder had the same structure with DeepECOC. The second architecture is $784 - 500 - 500 - 2000 - 10$. This architecture was used in [5].

In order to make DeepECOC adapt to this structure, we used the dense and sparse coding design methods that can control the codeword length. Note that, the dense and sparse coding design methods are totally random and data-independent. For the parameter settings, the denoising rate and dropout rate were set to 0.1, the batch size was set to 100, the learning rate was set to 0.01, and the number of epoch was set to 80000. For LeNet, we adopted the parameters as same as [36]. For PCANet, we used two PCA-filter stages, one binary hashing stage and one blockwise histograms. In the PCANet, the filter size, the number of filters, and the block size were set to $k_1 = k_2 = 8$, $L_1 = L_2 = 7$, and $7 \times 7$, respectively.

Tab. IV(a) and Tab. IV(b) show the experimental results on the two architectures. We can see that DeepECOC are comparable with existing deep learning methods on the second architecture and outperform them on the first architecture. Note that, DeepECOC with both two architectures outperform the single-layer ECOC approaches.

D. Classification on the LBP-CIFAR10 Data Set

The CIFAR-10 dataset is a relative large scale data set which consists of 60000 $32 \times 32$ colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. For the purpose of reducing computational cost, we attempted to extract features of the data using an efficient local binary patterns algorithm. As a result, the representations with dimensionality 36 and 256 were adopted and the data were normalized to [0, 1] as well. We called the new data sets LBP-CIFAR10 (36) and LBP-CIFAR10 (256), respectively. We used 3 hidden layers for all the deep learning methods. The learning rate was set to 0.1, and the epoch was set to 4000. For the LeNet model, we used two convolutional layers and two fully connected layers without pooling layers. The kernel size was set to $2 \times 2$, the stride was set to 1, the number of node of the first fully connected layer was set to 64, the epoch was set to 4000, the initial learning rate was set to 0.01, learning rate policy was set to “inv”, and the momentum was set to 0.9. For the PCANet model, we used two PCA-filter stages, one binary hashing stage and one blockwise histograms. In the PCANet, the filter size, the number of filters, and the block size were set to $k_1 = k_2 = 3$, $L_1 = L_2 = 4$, and $7 \times 7$, respectively. The classification accuracy are reported in Table V.

From Table V, we can easy to see that DeepECOC achieved the best results. Moreover, DeepECOC achieved the better results than autoencoder and denoising autoencoder, LeNet and PCANet. Hence, we can conclude that, DeepECOC are a general model to handle multi-class classification problems and achieves desirable results in most cases.

V. CONCLUSION

In this paper, we propose a novel deep learning model, called deep error correcting output codes (DeepECOC). DeepECOC extend traditional ECOC algorithms to deep architectures, and meanwhile, brings new elements to the deep learning area, such as supervised initialization, and automatic neglecting of part of the data during network training. Extensive experiments on 16 data sets from the UCI machine learning repository, the USPS and MNIST handwritten digits and the CIFAR-10 data set demonstrate the superiority of

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single</td>
<td>0.9331</td>
</tr>
<tr>
<td>AE</td>
<td>0.9023</td>
</tr>
<tr>
<td>DAE</td>
<td>0.9223</td>
</tr>
<tr>
<td>DeepECOC(Dense)</td>
<td>0.9791</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LeNet</td>
<td>0.9911</td>
</tr>
<tr>
<td>PCANet</td>
<td>0.9894</td>
</tr>
<tr>
<td>DAE</td>
<td>0.9880</td>
</tr>
<tr>
<td>DeepECOC(Sparse)</td>
<td>0.9892</td>
</tr>
<tr>
<td>DeepECOC(Dense)</td>
<td>0.9878</td>
</tr>
<tr>
<td>Single</td>
<td>0.9331</td>
</tr>
</tbody>
</table>
DeepECOC over traditional ECOC, feature learning and deep learning methods. In future work, we will further exploit the learnability of DeepECOC on large scale applications.

ACKNOWLEDGMENTS

This work was supported by the Science and Technology Program of Qingdao (No. 17-3-3-20-nsh), the CERNET Innovation Project under Grant No. NGII20170416, and the Fundamental Research Funds for the Central Universities of China. The Titan X GPU used for this research was donated by the NVIDIA Corporation.

REFERENCES

[34] I. Jolliffe, Principal component analysis. Wiley Online Library, 2002.

<table>
<thead>
<tr>
<th>Problem</th>
<th>AE</th>
<th>DAE</th>
<th>LeNet</th>
<th>PCANet</th>
<th>DeepECOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP-CIFAR10</td>
<td>0.3501</td>
<td>0.3678</td>
<td>0.3256</td>
<td>0.2569</td>
<td>0.5089</td>
</tr>
<tr>
<td>(356)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LBP-CIFAR10</td>
<td>0.4352</td>
<td>0.4587</td>
<td>0.3221</td>
<td>0.2569</td>
<td>0.5588</td>
</tr>
<tr>
<td>(256)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The best result for each scenario is highlighted in bold face.
Abstract—Fast and accurate keypoint detection and matching between two images are challenging tasks in image computer vision. Scale Invariant Feature Transform (SIFT) is one of the most accurate keypoint descriptors, however, it is not fast enough. On the other hand, other descriptors such as Speeded Up Robust Features (SURF) and Oriented FAST and Rotated BRIEF (ORB) are faster than SIFT, but in terms of accuracy cannot compete SIFT. While the extracted keypoints by these methods are dense the keypoints do not distribute uniformly over input image. This paper describes a novel approach to obtain more accurate and faster descriptors based on randomly predefined keypoints that present uniform distribution even over the textureless regions of sample image.

This research, combines an autoencoder and convolutional neural network (CNN) followed by dense layers to estimate the corresponding positions of predefined keypoints in the input image. The obtained results indicate that computation time decreased.

Keywords—Corresponding Points Extraction, Deep Neural Networks, Autoencoders, Convolutional Layer

I. INTRODUCTION

Nowadays, image descriptors are prevalent tools for image processing applications. These applications, such as object detection, 3D reconstruction, object recognition, segmentation and pixel-wised labelling utilize these descriptors. The main part of these applications is an image descriptor and their results firmly depend on the performance of these descriptors.

In 1999 Lowe [1] introduced Scale Invariant Feature Transform (SIFT) as a feature detector as the most accurate, rotation and scale invariant image descriptor. SIFT’s major drawback is its computational cost that prevents SIFT utilization in real-time applications. Various modifications have been applied on SIFT to reduce its computation cost. Speeded Up Robust Features (SURF) is a fast version of SIFT without losing its attributes [2]. Oriented FAST and Rotated BRIEF (ORB) [3] is another alternative to SIFT and SURF, which has lower computational cost. The ORB used BRIEF descriptor with many improvements such as orientation assignment and merged it with FAST keypoint detector to increase the performance.

The reviewed methods consist of a detector part and a descriptor part. Almost all of these methods run slowly depending on the texture of the input image. Due to the dependency of keypoint detection on the texture of the image, there is a possibility that some regions consist of many keypoints while others are represented by much fewer or even without any keypoints.

This research proposes a novel method based on deep neural networks to localize the specified points which are predefined in the training phase. The network takes the image as an input and estimates position of expected points utilizing convolutional and autoencoders properties. Figure 1 (top) demonstrates a sample image with its predefined keypoints which are selected uniformly from the image. Figure 1 (bottom) illustrates a deformed sample of the top image. The corresponding points to the predefined keypoints estimated by proposed method, are shown in the deformed image. In this image, correctly estimated and wrongly estimated points are specified with green and red stars, respectively. Computation time to find corresponding points approximately decreased by 98%.

Fig. 1. Top: a sample with uniformly specified points without any characteristics. Bottom: deformed version of the top image with determined points by proposed method.
The rest of this paper is organized to present related work and general review of the current methods in Section II. Section III explains methodology, Section IV delivers experimental results and evaluation, and finally, conclusion and further development appear in Section V.

II. RELATED WORK

SIFT descriptor is a classic approach and is known as the original inspiration for most of the descriptors proposed later. SIFT algorithm consists of four basic steps. The first step is scale-space extrema detection using Difference of Gaussian (DoG) to recognize potential interest points. Then, in keypoint localization step, identified points are refined by rejecting the low contrast points. In the next step, an histogram of orientation is assigned to each point based on image gradient and finally a description is generated for each keypoint based on magnitude and orientation. SIFT is the most accurate descriptor in the scale and orientation descriptors family, however, it is mathematically complicated and consequently has heavy computations, which make it inappropriate for real-time applications [5]. Although, SIFT takes advantage of good performance, there are some outliers among its matched pairs. Several researches have been done to overcome the outliers effect [6] [7].

SURF almost employs the same method for feature detection. In the first step, SURF goes further and uses box filter to approximate Laplacian of Gaussian (LoG) rather than DoG in SIFT. Since the convolution with the square is much faster using the integral image, thus, the convolution is easily calculated. Furthermore, both scale and location rely on the determinant of Hessian. In orientation assignment step, SIFT uses wavelet result by applying adequate Gaussian weights in both horizontal and vertical directions. Additionally, in the last step, SURF utilizes wavelet responses. SURF is faster than SIFT while performance is comparable to SIFT. SURF can handle images with blurring and rotation, but it is not good at handling viewpoint and illumination changes [2].

ORB is based on FAST keypoint detector and improved BRIEF descriptor. Initially, to identify keypoints ORB employs FAST and then finds the top N points among them by applying Harris corner detector [8]. It handles the orientation issue by computing the intensity weighted centroid of patch. Although ORB is a suitable alternative for SIFT and SURF to reduce the computation cost, in terms of robustness cannot compare to SIFT and SURF.

Recently, researchers started to employ neural networks to detect keypoints. LIFT introduced by Yi et al [4] that proposed a stack of convolutional networks that combines detection, orientation assignment and description networks, into a single network. Similar to Sift, LIFT first detect keypoints and then describe them. Using deep learning to address many computer vision challenges became pervasive in the recent decade. Deep neural networks, especially convolutional networks started to outperform traditional methods and have been used successfully in image processing tasks such as object detection, image classification and 3D reconstruction [9]. The designed network in this research is a fusion of convolutional networks and autoencoders network. The network takes the image as an input and estimates position of expected points utilizing convolutional and autoencoders properties.

III. PROPOSED METHOD

This study includes three major modules: I) Generating the dataset, II) feature extraction from the input image using autoencoder network, and III) combining encoder part of the employed autoencoder in the second module with convolution layers and estimating the position of points as is illustrated in Figure 2. This section, briefly describes the reasons of generating training dataset based on the type of required data and the procedure to create this dataset and then explains the proposed approach in detail.

A. Generating The Dataset

The main target is to find some specified points in the image based on its concept rather than any special characteristics and features. Since there is no available appropriate dataset in this research area to meet the requirements such as including the same scene with different views of an image, thus the authors made effort to provide a synthetic training dataset including all deformed, slightly rotated, scaled and affine transform versions of samples. This dataset will be published in open source repository to be available for other researchers to develop the matching of corresponding points in images. For this purpose, a sample image is considered and the positions of different

---

Figure 2. The structure of designed deep neural network.
number of grid points are specified on it. Then, various deformation such as slight rotation and cylindrical deformation with different distances are applied to it and new positions of grid points are calculated.

### B. Feature Extractor

Extracted points by mentioned descriptors were found in parts of image that has rich texture like edges and corners [10] however, the specified points for proposed method do not have any special characteristics and they can be localized by their neighborhoods. In this module an autoencoder extracts features from image that can completely describe it.

An autoencoder is a kind of feedforward neural network that tries to learn an approximation of the identity function in order to equalize target value to the input. Generally, autoencoders consist of two parts, the encoder that produces representation of input and the decoder that reconstructs input. If consider encoder as function \( f(x) \) and decoder as \( g(c) \), so autoencoder can be defined as:

\[
g(f(x)) = r
\]

where \( x \) is the input, \( c \) is produced code by encoder and \( r \) is the specify output of network. The autoencoder network is used as a tool for representation learning and tries to find some intermediate feature representation for each image [11].

Employed autoencoder in this work is an undercomplete autoencoder that its code dimension is less than input dimension. The feature extractor module is encoder part of the autoencoder network. Following loss function was employed to train the network.

\[
\min(mse(x, g(f(x))))
\]

(2)

So if the output was different from input, loss function penalizes \( g(f(x)) \) and trains the network.

The layers of autoencoder network are convolutional. As illustrated in Figure 2 the encoder part contains three convolution layers and activation of these layers are \( \text{relu} \) that is defined as:

\[
f(x) = \max(0,x)
\]

(3)

where \( x \) is the input to a neuron [12]. By using \( \text{relu} \) function as activation of layers in the network, both encoder and decoder learn nonlinear functions because the \( \text{relu} \) is a piecewise linear function. Given \( x_1 \) and \( x_2 \) as inputs of \( \text{relu} \) such that:

\[
R(x_1) = y_1
\]

(4)

\[
R(x_2) = y_2
\]

(5)

then

\[
R((\alpha \times x_1) + (\beta \times x_2)) = (\alpha \times y_1) + (\beta \times y_2)
\]

(6)

Undercomplete autoencoder with nonlinear encoder and decoder can learn to span more powerful subspace of PCA [11]. After training this network, the weights of encoder part are cut and used as a feature extractor.

### C. Position Estimator

Position estimator finds points positions using extracted features. This part contains three convolution layers followed by pooling and a fully connected layer as the network output layer. Convolutional network is a type of feed forward network that is known as shift and space invariants and is generally used for the kinds of data that have grid-like topology such as image which is a 2D grid of pixels. Two major features of convolutional layers are sparse weights that refers to the use of few parameters as kernel of convolutional layers to extract meaningful features and fewer computations, second is parameter sharing because of using all the kernel members for all pixels of image [11]. Convolution layers learn to use the best extracted features from image. Each convolution layer is followed by a max-pooling layer to summarize their output neighbors. Pooling layer makes the network to be invariant to small translations.

Network finishes with a dense layer that its nodes depends on the number of specified points for estimating. Suppose that output of the network contains \( n \) points, so the output of fully connected layer includes \( 2n \) numbers that represent the coordinate values \((x, y)\), the first \( n \) ones denote \( x \) positions and the second \( n \) ones show \( y \) positions. Because the expected outputs of the network are numbers between 0 and 1, sigmoid function was utilized as an activation of fully connected layer to produce numbers in specified range. The activation of convolution layers is \( \text{relu} \). Kernels of convolution layers cause the network to consider local neighborhood of points. Consider a convolutional layer with an input with height \( H \), width \( W \) and channels \( C \) such that \( I \in \mathbb{R}^{H \times W \times C} \). Subsequently for \( D \) filters, a tensor \( T \) exists such that \( T \in \mathbb{R}^{k_1 \times k_2 \times C \times D} \) where \( k_1 \times k_2 \) are filter dimensions and biases \( b \in \mathbb{R}^D \). The \( ij \)-th output of convolution layer is as follow:

\[
(I \ast T)_{ij} = \sum_{m=0}^{k_1-1} \sum_{n=0}^{k_2-1} \sum_{c=0}^{C-1} T_{m,n,c} .I_{i+m,j+n,c} + b
\]

(7)

that shows \( k_1 \times k_2 \) local neighbors are used for \( ij \)-th output.

Last layer of network is fully connected that receives the output of last convolutional layer in flatten shape as input. Suppose flatten layer has \( M \) outputs, and \( n_j \) denotes the \( j \)-th output of dense layer. So:

\[
z_j = \sum_{i=1}^{M} w_{ij} m_i + b_j
\]

(8)

\[
n_j = \text{sig}(z_j)
\]

(9)

Equations 8 and 9 shows that dense layer makes the network to take global neighborhood of points in consideration. therefore if some points do not exist in an image, the network can estimate their positions.

To train the designed network, first the autoencoder is trained. After concatenating the encoder (feature extractor) with
estimator, the layers of feature extraction are freeze in order to find sophisticated features. Second step just trains the estimator.

IV. EXPERIMENTAL RESULTS

Images are generated with MATLAB and the network implemented with Keras. The datasets contain 10 000 images divided into 7000 train images, 1500 test images and 1500 validation images. In the first dataset each image contains 200 predefined points and in the second one, each image contains 600 predefined points. Selected points are on a grid, but they can be anywhere in the image and it is possible to specify more points. In the third dataset, ORB points employed for training.

Since, SIFT is known as an accurate method, the proposed method is compared with SIFT. The comparison is based on true matching rate, where:

\[
\text{True Matching Rate} = \frac{\text{True Matched keypoints}}{\text{Matched keypoints}}
\]

TABLE I. COMPARISON BETWEEN TIME CONSUMING AND TRUE MATCHING RATE OF THE PROPOSED METHOD AND OTHER RELATED METHODS.

<table>
<thead>
<tr>
<th></th>
<th>Found Keypoints</th>
<th>True Matching Rate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFT</td>
<td>3320</td>
<td>87%</td>
<td>0.77 s</td>
</tr>
<tr>
<td>SURF</td>
<td>6896</td>
<td>66.62%</td>
<td>0.50 s</td>
</tr>
<tr>
<td>ORB</td>
<td>500</td>
<td>56.3%</td>
<td>0.036 s</td>
</tr>
<tr>
<td>LIFT</td>
<td>600</td>
<td>91.87%</td>
<td>0.74 s</td>
</tr>
<tr>
<td>Proposed Method for 200 point</td>
<td>200</td>
<td>88%</td>
<td>0.37 s</td>
</tr>
<tr>
<td>Proposed Method for 600 point</td>
<td>600</td>
<td>79.54%</td>
<td>0.36 s</td>
</tr>
<tr>
<td>Proposed Method for ORB points</td>
<td>500</td>
<td>94.2</td>
<td>0.36 s</td>
</tr>
</tbody>
</table>

A matched pair is labeled as a true matched pair, if the distance of the estimated point of matched pair to its real position is less than 5 pixels. The offline training phase of this template-based method is time consuming, while in online phase it is considerably faster than SIFT. As it is observed from Table 1, this research improves true matching rate and processing time is reduced. As Figure 3 illustrates, another weakness of SIFT is that points are centralized in special parts like edges, but in proposed method, determined points can have diversity and have good result in textureless images.

V. CONCLUSION

The research is undertaken proposed a novel deep neural network trained with back-propagation. It contains feature extractor and position estimator steps. Experimental results confirmed the high speed finding points performance of this method even in the lack of them in the image. This method is scale and affine invariant and robust against deformation. The number of detected points is a user defined parameter and the designed network can find any number of points. The proposed method uses local and global features in the image to find point positions. Further development will be conducted to investigate blur and more rotated images issues regarding match points.

VI. REFERENCES

Cardiac Murmur Classification in Phonocardiograms using Deep Convolutional Neural Networks

Pengcheng Xi  
Carleton University &  
National Research Council Canada  
Ottawa, Canada  
pengcheng.xi@nrc-cnrc.gc.ca

Rafik Goubran  
Systems and Computer Engineering  
Carleton University  
Ottawa, Canada  
goubran@sce.carleton.ca

Chang Shu  
Computer Vision and Graphics  
National Research Council Canada  
Ottawa, Canada  
chang.shu@nrc-cnrc.gc.ca

Abstract—Cardiac murmurs are the first signs of pathological changes in heart valves. Their subtle presence poses great challenges for detection through auscultation or phonocardiograms (PCGs); therefore computer-aided detection (CAD) of heart murmurs has medical significance in assisting health care professionals. Traditional CAD approaches relying on engineered features are prone to changes in environmental noise and data collection methods, whereas deep Convolutional Neural Networks (CNN) have shown robustness in advancing the performance of computer vision tasks through automatic feature learning from large amount of data. With a limited set of labelled PCG recordings, this work conducts data augmentation and successfully tunes state-of-the-art deep CNN architectures with transfer learning. The fine-tuned deep CNNs demonstrate their effectiveness in classifying cardiac murmurs from PCG recordings without segmentation.

Index Terms—computer aided analysis, feature extraction, neural networks, image classification

I. INTRODUCTION

According to World Health Organization (WHO), cardiovascular disease (CVD) is the leading cause of mortality worldwide [2]. The majority of the problems found in CVD are related to heart valves, in which cardiac murmurs are the first signs of pathological changes. Therefore, classic auscultation and phonocardiography (PCG) play essential roles in disease evaluation and early detection of CVDs, providing guide for further examinations.

Cardiac valve problems include mitral valve prolapse, mitral valve or aortic stenosis, aortic sclerosis and stenosis, and mitral or aortic regurgitation, etc. Stenosis leads to heart working harder to pump blood to the rest of the human body. Regurgitation means the blood is going the wrong way through the valves, resulting in heart working harder to force blood through the damaged valves. Both can wear out the heart and lead to heart failure.

Cardiac murmur is the sound of blood flowing through a problematic heart valve. It may also be a condition which makes the heart beat faster and forces heart to handle more blood quicker than normal. Most heart murmurs are innocent and do not require treatment; however murmurs linked to damaged or overworked heart valves need surgery or treatment. Some murmurs are congenital and others are a part of ageing or from other heart problems.

The evaluation of cardiac murmurs is based on its timing, shape, location, intensity and duration in the cardiac cycle (S1, systolic, S2 and diastolic). They can be categorized into systolic or diastolic murmurs. In each situation, the murmur takes a particular shape, e.g., crescendo-decrescendo shape for aortic stenosis, decrescendo shape for aortic regurgitation, plat shape for mitral regurgitation or tricuspid regurgitation. This implies that visual representations of heart recordings (in time and frequency domains) can be utilized for classifying cardiac murmurs.

Because of the medical significance of cardiac murmurs, there has been considerable effort on developing computer-aided detection (CAD) approaches [15] [14] [3]. Conventional approaches follow three steps: segmentation, feature extraction and classification. The features are selected in three domains: time, frequency, and time-frequency. While the manual features work in certain cases, they are prone to changes in environmental noise and data recording instruments.

Recent advances in deep neural networks have enabled automatic feature learning from large amount of training data, providing an end-to-end solution from feature extraction to classifier building. Moreover the learning scheme contributes to the robustness of deep neural networks to dataset noise, making them suitable for solving the cardiac murmur classification problem.

Among deep neural networks, Convolutional Neural Networks (CNN) are especially effective in processing visual data. Consequently, deep CNNs are the best candidates for cardiac murmur classifications because of the following reasons. First, with minimum information loss, PCGs can be converted to spectrograms, which are visual data expected as the input of deep CNNs. Second, deep CNNs learn hierarchical features automatically from data. Therefore, they provide a powerhouse for feature learning in order to distinguish cardiac murmurs from normal ones. This work addresses the following questions: i) how deep CNNs will learn from a limited amount of labelled PCGs, ii) whether deep CNNs are robust in classifying PCGs without segmentation, and iii) whether the fine-tuned models get the same performance boost as they did in ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) [13].

To address the limited training data problem, we employ...
data augmentation and transfer learning to successfully fine-tune state-of-the-art deep CNN architectures. Experimental results indicate that our method provides an effective and practical solution to the problem of classifying cardiac murmurs.

Our contributions are three-fold:

- Significantly leveraged deep CNNs’ hierarchical feature extraction capabilities through transfer learning. This enables the extraction of features for identifying the subtle and challenging differences between cardiac murmurs and normal PCGs without segmentation.
- Successfully tuned deep CNNs on a small training dataset without over fitting. This is achieved by employing a simple yet effective data augmentation approach.
- Provided a guidance on how to select deep CNN models for learning from small datasets effectively.

II. RELATED WORK

In this section, we first review the top three results on classifying normal and abnormal heart sound recordings in the PhysioNet/CinC challenge 2016 [10]. What is common among the winning methods is the use of shallow neural networks. In contrast, our work attempts to use deep neural networks because of their demonstrated advantage over traditional methods in many computer vision tasks. However, one of the challenges in training deep neural networks with limited data is to avoid over fitting. Therefore, we discuss the many techniques for dealing with the problem.

Potes et al. [11] proposed an approach of using ensemble of feature-based and deep learning-based classifiers for the detection of abnormal heart sounds. The classifier ensemble approach obtained the highest score in the competition. Their classifier using CNN was trained using PCGs decomposed into four frequency bands. Each of the CNNs consists of three layers: the input layer followed by two convolution layers.

Zabihi et al. [20] used ensemble of neural networks without segmentation and achieved the second best score in the PhysioNet/CinC Challenge 2016. They use 20 feed-forward neural networks - two hidden layers in each and 25 hidden neurons in each layer. A combination rule is then applied to identify the quality and abnormality of each input.

Kay and Agarwal [8] built a fully-connected two-hidden-layer neural network trained by error back-propagation, and regularized with DropConnect to classify heart sounds as normal or abnormal.

The above approaches all use shallow neural networks as part of their solution mainly due to the limited amount of training data. Other factors that may affect their performance include the manually decided thresholds used in the combination rules [20] and the decomposition of frequency bands [11]. Segmentation on the input is avoided in [20].

Limited amount of training data leads to overfitting in deep neural networks. Solutions include dropout, batch normalization and transfer learning [1]. Among all, the simplest is to add regularization term on the connection weights. Dropout works by removing neurons or connections from the network probabilistically during training [17]. Batch normalization works by normalizing layer inputs. It allows the use of much higher learning rates and results in less sensitivity to weight initialization [7]. Finally, Yosinski et al. [19] proposed a transfer learning approach that makes use of a neural network which has been pre-trained on large amount of data and fine-tune some of the layers. Other transfer learning approaches learn from unlabeled data using stacked auto-encoders and then transfer the knowledge to a labeled dataset [6].

Data augmentation is another way to avoid over fitting in deep learning. It works by increasing the amount of training data from the existing training set. In image processing, this includes operations such as cropping, flipping and rotating the original images. It has been proved to be effective in avoiding over fitting.

III. METHODOLOGY

A. Data Selection

To study heart sound problems, there are four public heart sound databases available: (i) the PhysioNet/CinC Challenge 2016 [10], (ii) the PASCAL Heart Sounds Challenge 2011 [4], (iii) the Michigan heart sound and murmur database (MHSDB) and (iv) the Cardiac Auscultation of Heart Murmurs database (eGeneralMedical). The PhysioNet/CinC dataset is by far the largest dataset with heart sound recordings labelled as normal and abnormal; however no further labels on murmurs are available. The MHSDB and eGeneralMedical datasets have very limited data. The PASCAL dataset is selected for this work as it comprises at-scale heart sound recordings labelled as normal or murmur.

From PASCAL dataset, we selected a subset which was collected from a clinical trial in hospitals using a digital stethoscope. Within the dataset, 66 murmur recordings and 200 normal recordings are used for this study. All the recordings were sampled to 4,000 Hz and their length varies from 1 second to 30 seconds.

B. Data Pre-processing

To take advantage of the automatic feature extraction capabilities of deep CNNs, spectrograms are first computed from PCGs. According to [12], most cardiovascular murmurs show a frequency range extending from almost zero to 700 Hz. To compute spectrograms, we applied a 512-point Hamming window with 75% overlapping for computing Fast Fourier Transform (FFT) of size 512. A plot of the spectrograms computed on one murmur PCG and one normal PCG is shown in Fig. 1, in which limits of the color axis for spectrograms are set to be between -120db and -40db.

Traditional approaches segment PCGs into S1, systole, S2 and diastole, while this work performs feature extraction and classification without segmentation. Our motivation is to conduct feature extraction and classification from data automatically.

C. Data Augmentation

An observation on the PCGs and computed spectrograms shows that most cardiac cycles last no more than 1.5 seconds;
therefore, to prepare training images for deep CNNs, a sliding
window with length of 1.5 seconds is used to extract image
patches from spectrograms with an initial step size of 500
samples. As a result, those heart recordings with a length
shorter than 1.5 seconds are excluded from this study, leaving
the final number of training subjects for murmur and normal
categories to be 64 and 180 respectively.

All the windowed image patches are put into two categories,
i.e., murmur or normal, and a pre-trained deep CNN is
modified to classify two classes at the output. Image patch
samples from the two categories are plotted with Fig. 2.

We conduct five-fold cross validations on the training
dataset. For each fold, an independent testing set of 12 subjects
are set aside for the murmur category and a testing set
of 35 subjects are for the normal category. The remaining
subjects within each category are used for extracting training
image patches. The grouped image patches are used to fine-
tune deep CNNs. In testing, image patches are first extracted
from a spectrogram and then fed into the trained model for
classification. All the classification results are used to vote for
the final classification result of current spectrogram.

D. Feature Extraction and Model Building

As one of the transfer learning approaches, the image
patches from both categories are fed into the modified deep
CNNs for fine-tuning. The first part of each model is frozen,
leaving the last layers actively tuned for best performance.

To test trained deep CNN models, we apply the same sliding
window approach (same width and step size) to consecutively
extract image patches along a spectrogram. Each patch is fed
into the trained deep CNN to predict a label. The predicted
labels are then agglomerated to a voting scheme for creating
a final prediction. The approach is illustrated in Fig 3.

E. Architectures of Deep CNN

In visual computing, tremendous progress has been made in
object classification and recognition thanks to the availability
of large scale annotated datasets such as ImageNet Large
Scale Visual Recognition Competition (ILSVRC) [13]. The
ImageNet dataset contains over 15 million annotated images
from a total of over 22,000 categories.

Recent years witnessed great performance advancement on
ILSVRC using deep CNNs. Comparing to traditional hand-
crafted image features, deep CNNs automatically extract fea-
tures from a large dataset for tasks they are trained for. In this
work, we experiment on modifying four of the best-performing
models in recent ImageNet challenges and compare their
performance on classifying normal and murmur PCGs.

- AlexNet - in 2012, Krizhevsky et al. [9] entered ImageNet
ILSVRC with a deep CNN and achieved top-5 test error
rate of 15.3%, compared to 26.2% achieved by the
second-best entry. The network was made up of 5 conv
layers, max-pooling layers, dropout layers, and 3 fully
connected layers. This work led to a series of deep CNN
variants in following years which consistently improved
the state-of-the-art in the benchmark tasks.
- VGGNet - in 2014, Simonyan and Zisserman [16] introduced a deeper 19-layer CNN and achieved top result in the localization task of ImageNet ILSVRC. The network used very small 3x3 convolutional filters and showed significant improvement. This influential work indicated that CNNs need to have a deep network of layers in order for the hierarchical feature representations to work.
- GoogLeNet - in 2014, Szegedy et al. [18] introduced a deeper CNN to ILSVRC and achieved top 5 error rate of 6.7%. Instead of sequentially stacking layers, this network was one of the first CNNs that used parallel structures in its architecture (9 Inception modules with over 100 layers in total).
- ResNet - in 2015, He et al. [5] introduced a new 152-layer network architecture and set new records in ILSVRC. ResNet achieved 3.57% error rate in the classification task. The residual learning framework is 8 times deeper than VGGNet but still has lower complexity.

All the deep CNN architectures were designed for a 1000-class classification task. To adapt them to our task, the last three layers were removed from each network. Three new layers (fully connected layer, softmax layer and classification layer) were appended to the remaining structure of each network. Higher learning rates were set for the newly added fully connected layers so that the first part of each network remains relatively unchanged during training and the newly added layers get fine-tuned with our dataset. Cross validation is then used to test the robustness of each trained deep CNN.

IV. RESULTS
A. Comparison on Results from Deep CNN Architectures

We apply the following parameter settings for training each modified deep CNN: Stochastic Gradient Descent with Momentum (SGDM) as the optimization algorithm, batch size of 10, initial learning rate as 1e−4, and the learning rate for the last fully connected layer as 20.0. Each network stops from further training if the mean accuracy on the fifty most recent batches reaches 99.5%. On average, the training takes between 20 and 40 minutes to complete on an NVIDIA GeForce GTX TITAN X GPU (20 minutes for AlexNet, 40 minutes for VGGNet, and 30 minutes for both GoogLeNet and ResNet). The final size of fine-tuned VGGNet is about 20 times that of GoogLeNet, with in-between sizes for AlexNet and ResNet.

The voting strategy during testing on a spectrogram is as follows. If over 20% of the image patches extracted from a spectrogram are classified as murmur, the corresponding PCG is then classified as murmur, otherwise as normal. For each deep CNN, we compute the average accuracies (sensitivity, specificity and overall accuracy) from the five-fold cross validations in Table I. Varying the percentage threshold used in voting makes small updates to the accuracies and they are not big enough in changing the difference among the methods. As an example, increasing the threshold to 60% leads to overall accuracies for the four models being 86.36%, 84.38%, 83.02%, and 60.86%, respectively.

<table>
<thead>
<tr>
<th>Model</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Overall Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlexNet</td>
<td>85.00%</td>
<td>90.36%</td>
<td>87.93%</td>
</tr>
<tr>
<td>VGGNet</td>
<td>81.67%</td>
<td>89.14%</td>
<td>85.40%</td>
</tr>
<tr>
<td>GoogLeNet</td>
<td>95.00%</td>
<td>46.29%</td>
<td>70.64%</td>
</tr>
<tr>
<td>ResNet</td>
<td>93.33%</td>
<td>22.29%</td>
<td>57.81%</td>
</tr>
</tbody>
</table>

One natural question is how data augmentation will affect the classification performance of different models. Keeping the rest parameter settings the same, we reduce the window step size from 500 to 100 in order to create more training image patches for both categories of murmur and normal. Because different folds contain PCG recordings at different lengths, we count the average number of murmur and normal training images. The averages increase from (6,633, 9,345) to (13,240, 18,617) after step size change. Feeding the larger dataset to the four deep CNNs for training and then testing with five-fold cross validations lead to updated performance in Table II.

<table>
<thead>
<tr>
<th>Model</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Overall Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlexNet</td>
<td>81.67%</td>
<td>98.86%</td>
<td>90.26%</td>
</tr>
<tr>
<td>VGGNet</td>
<td>76.67%</td>
<td>97.71%</td>
<td>87.19%</td>
</tr>
<tr>
<td>GoogLeNet</td>
<td>98.33%</td>
<td>45.14%</td>
<td>71.74%</td>
</tr>
<tr>
<td>ResNet</td>
<td>76.67%</td>
<td>56.57%</td>
<td>66.62%</td>
</tr>
</tbody>
</table>

B. Prediction on Spectrograms using AlexNet and ResNet

We further analyse the performance of different deep CNN models on classifying individual image patches extracted from two spectrograms (one from murmur PCG and the other from normal PCG). In particular, we select AlexNet and ResNet as they represent the best and worst performing models in overall prediction accuracy. We randomly select two spectrograms (one murmur and the other normal) from the independent test set and run classifications with the fine-tuned deep CNNs, which are also randomly selected from one of the five-fold models trained with each architecture.

The first set of prediction results in Fig. 4 are the outputs from deep CNNs trained on image patches extracted at 500 samples apart. In each prediction plot, we use ‘1’ for output class ‘murmur’ and ‘0’ for ‘normal’. Their location in the plots matches the center of each sliding window. These plots help identifying the area of the spectrogram that is classified properly or misclassified. The left column from top to bottom is the spectrogram for a murmur PCG, prediction results from one of the five-fold AlexNet models, and results from one of the five-fold ResNet Models. The right column of Fig. 4 are the prediction results on a normal PCG.

Fig. 4 indicates that both networks give accurate predictions for the murmur spectrogram, while ResNet predicts poorly on the normal spectrogram (lower right sub-plot in Fig. 4).

The second set of prediction results in Fig. 5 are the outputs from deep CNNs trained on image patches extracted at 100
samples apart. Parameter setting for testing remains the same.

Fig. 5 indicates that, after data augmentation and model re-
tuning, ResNet improves its prediction performance on the
same normal spectrogram (lower right sub-plot in Fig. 5).

C. Prediction on Spectrograms using VGGNet and GoogLeNet

Using the same test examples as previous experiments, we
conduct classification experiments on consecutive spectrogram
image patches using the fine-tuned VGGNet and GoogLeNet.
The classification output results are in Fig. 6. In comparison
with Fig. 5, VGGNet has similar performance as AlexNet
in that both make perfect classifications on the normal spec-
trogram image patches. Also both models make mistakes in
classification on two image patches which locate differently
in the murmur spectrogram.

Likewise, GoogLeNet makes perfect classifications on the
murmur spectrogram as ResNet does in Fig. 5. For the normal
spectrogram in both figures, ResNet has more correct hits than
GoogLeNet.

In summary, the first comparison in section IV-B demonstr-
ates that increasing the granularity during training image
extraction leads to improved performance for ResNet. In both
cases, AlexNet gives close to perfect classification result on
the two test spectrograms. Maintaining the same granularity
as in Fig. 5, we make a further comparison in section IV-C
after running classifications with fine-tuned VGGNet and
GoogLeNet models. Fig. 6 indicates that VGGNet makes good
classification on both spectrograms whereas GoogLeNet has
misclassified most of the image patches extracted along the
normal spectrogram.

V. DISCUSSIONS

The four deep CNN models in comparison demonstrated
consistent improvement on the ILSVRC benchmark tasks;
however, the same trend does not hold in the cardiac murmur
classification task. Our explanation is that deeper models
(ResNet being the deepest among the four) have more com-
plexity and parameters, therefore they are more data thirsty in
fine tuning. This has been verified through our experiments in
which data augmentation with denser image patch extractions
leads to improved performance of the deep CNNs, most
notably for ResNet.

We use a simple voting strategy during testing. In a different
voting strategy, the probability of a PCG recording being
classified as either murmur or normal can be computed as the
average probabilities assigned to its randomly selected window
patches.

The comparisons are conducted on two randomly selected
testing spectrograms with randomly selected fine-tuned deep
CNN models. With a different selection on both testing data
and models, the results will be slightly different but the observed difference should remain the same.

VI. CONCLUSIONS AND FUTURE WORK

The classification between cardiac murmur and normal PCGs is a challenging problem because of subtle differences. This work successfully employs deep CNNs with transfer learning and data augmentation to learn from a small dataset of PCG recordings. The power of deep CNNs on automatic feature extractions enables accurate classifications without the need of segmentation. Four state-of-the-art deep CNN architectures are compared on classification performance. The best models are able to classify PCGs at a high accuracy (the best overall accuracy at 90.26% by AlexNet in our experiments).

In the ILSVRC benchmark tasks, deeper models showed higher performance than their counterparts with less number of layers. Training with the limited murmur dataset, the deeper models do not share the same performance gains. In fact, the least deep model in selection demonstrates the best performance in all comparisons. Fine tuning the deep CNN models is data thirsty and data augmentation improves performance for all the deep CNN models. The result can be used as a guidance for deep model selections in a similar problem given limited amount of training data.

Future work includes studying the problem with a cardiologist to better understand the classification results.

ACKNOWLEDGMENT

The authors would like to thank Dr. Hongyu Guo for the discussions on experimental designs used in this work.

REFERENCES

Object cosegmentation using deep Siamese network

Prerana Mukherjee*, Brejesh Lall* and Snehith Lattupally*
*Dept of EE, IIT Delhi, India.
Email: {eez138300, brejesh, eet152695 }@ee.iitd.ac.in

Abstract—Object cosegmentation addresses the problem of discovering similar objects from multiple images and segmenting them as foreground simultaneously. In this paper, we propose a novel end-to-end pipeline to segment the similar objects simultaneously from relevant set of images using supervised learning via deep-learning framework. We experiment with multiple set of object proposal generation techniques and perform extensive numerical evaluations by training the Siamese network with generated object proposals. Similar objects proposals for the test images are retrieved using the ANNOY (Approximate Nearest Neighbor) library and deep semantic segmentation is performed on them. Finally, we form a collage from the segmented similar objects based on the relative importance of the objects.

Keywords—Cosegmentation, Siamese Network, Multiscale Combinatorial Grouping, Nearest Neighbor

I. INTRODUCTION

Automated foreground segregation and localization of objects constitute the fundamental problem in computer vision tasks. Further the lack of sufficient information about the foreground objects makes it highly complex to deal with it. The exploitation of the commonness prior and the joint processing of similar images (containing objects of same category) can aid in the process of such object related tasks. Cosegmentation refers to such class of problems which deals with the segmentation of the common objects from a given set of images without any priori knowledge about the foreground. It was first hypothesized in [1] that in most cases the common objects for cosegmentation represent the ‘objects of interest’ which appear in the images instead of common background details. These objects have huge variations in terms of scale, viewpoint, rotation, illumination, location and affine changes. In other cases, it may be highly occluded by other objects. Even same class of objects may drastically differ in appearance resulting in high intra-class variation.

The works in [2]–[4] solve the generic object cosegmentation by applying the localization constraint that in all the images the common object will always belong to the salient region. Some of the methods are confined to the cosegmentation between image pairs [5]–[7] while others require some user intervention [8], [9]. Further [5], [10] pose it as segmenting only those objects that are exactly similar. These approaches are unable to handle the intra-class variations or other synthetic changes or noise which might be present in case of images that are downloaded from Internet. In recent years, with increase in the computational power and access to widespread availability of semantic annotations for object classes, deep learning has achieved dramatic break-through in various applications. Siamese Network has also been extensively used for many vision applications. They have been used to learn the similarity metrics by aligning the similar objects together and dissimilar objects far away. This motivates us to solve the cosegmentation problem using the high-level features extracted using deep networks. We propose to couple the similarity based clustering and cosegmentation task so that they can coexist and benefit from each other synergistically.

In this work, we pose cosegmentation as a clustering problem using the Siamese network. For a given set of images, we train the Siamese twin architecture to assess the similarity of two equally sized patches. These patches are the object proposals of an image. Co-segmenting the objects using trained model is done using high-level features utilizing fully convolutional networks [11] rather than low-level features like SIFT, HOG etc. Finally, we create a visual summary from the segmented images based on their similarity score in the respective class. In view of the above discussions, the major contributions of this paper are:

1) Cosegmentation is posed as a clustering problem to align the similar objects using Siamese network and segmenting them. We also train the Siamese network on non-target classes with no to little fine-tuning and test the generalization capability to target classes.

2) Generation of visual summary of similar images based on relative relevance.

Rest of the paper is organized as follows. In Sec. II we describe the proposed approach in detail. In Sec. III we present the results and discussions. Finally, we conclude the paper in Sec. IV.

II. METHODOLOGY

In the following subsections, we describe the components of the proposed method. Fig. 1 shows the overall pipeline of the proposed method.

A. Siamese network

For given set of images, we first generate the object proposals using different object proposal techniques as described in Sec. III. The generated object proposals are given to Siamese Network for training. Siamese Networks are useful in finding similarities and relationship between different structures. The Siamese configuration consists of two convolutional neural networks (CNNs) with shared weights with a contrastive loss layer. The input to the Siamese network are two input patches (object proposals) along with a similarity label. Similarity label “1” indicates that patches are similar while “0” indicates dissimilar patches. Two CNNs generate a N-Dimensional feature vector in forward pass. The N-Dimensional vectors are fed to the contrastive loss layer which helps in adjusting the weights such that positive samples are closer and negative samples are far from each other. Contrastive loss function penalizes the
positive samples that are far away and negative samples that are closer. Let us consider two patches \((x_1, x_2)\) that are fed to Siamese network. Let us assume the N-Dimension vectors generated by convnets as \(f(x_1)\) and \(f(x_2)\). \(Y\) be the binary label \(Y \epsilon \{0, 1\}\), \(Y=1\) for similar pairs and 0 otherwise. Margin \(m\) is defined for the contrastive layer such that positive samples are at a distance less than margin and negative samples are at a distance greater than margin. Thus, the contrastive loss function is given as,

\[
L(W, Y, x_1, x_2) = Y \frac{1}{2}D_W^2 + (1 - Y) \frac{1}{2}\max(0, m - D_W^2)
\]

(1)

where \(D_W^2 = \|f(x_1) - f(x_2)\|^2\) is the Euclidean distance between the two feature vectors of the input patches. The outputs from the fully-connected layers are fed to contrastive layers, which measures the distance between two features. The weights \(W\) are adjusted such that the loss function is minimized.

After training the Siamese Network, we deployed the trained model on test images. First we extracted the object proposals for the test images. A N-Dimensional feature vector is generated for each of the proposals. In our experiments, we used 256-Dimensional feature vector. The features generated for test image proposals using trained Siamese network are fed to Annoy (Approximate Nearest Neighbor) Library \[^1\]. It measures the Euclidean distance to all other features and indices of neighbors are assigned in the increasing order of their Euclidean distance. It has many advantages as compared to other nearest neighbor search algorithms. These include (i) small memory footprint (ii) Annoy creates data structures (read-only) which can be shared among multiple processes.

**B. Segmentation**

Segmentation is performed on the retrieved similar object proposals. We used Fully convolutional Networks for semantic segmentation proposed by Jonathan Long et al. \[^11\]. Convolutional networks are used as powerful tools for extracting the hierarchy of features. The contemporary classification networks are adapted to the segmentation tasks by transferring the learned representations. It utilizes a skip architecture which combines the semantic information from deep (coarse information) and shallow (fine appearance information) layers. The fully connected layers can also be considered as convolutions with kernels covering entire image. Transforming the FC layers into convolutional layers converts the classification network to generate a heat map. However, the generated output maps are of reduced size as compared to the input size. So, dense predictions are made from coarse maps by upsampling. Upsampling is performed by backward convolution (also called as deconvolution) with stride as \(f\). Skip layers are added to fuse semantic and appearance information.

**C. Visual Summary based on relative importance**

A visual summary is created from the segmented proposals. While retrieving the similar object proposals using ANNOY library, we preserved the Euclidean distances corresponding to each of the proposals. A basic collage is formed with 10 slots constituting the most similar proposal (least Euclidean distance) getting a larger block. The remaining segmented

\[^1\]https://github.com/spotify/annoy
objects are placed in the other slots and a background is added to the image.

III. EXPERIMENTAL RESULTS

In this section, we discuss the empirical results on two publicly available benchmark co-segmentation datasets. We describe the datasets used followed by implementation details and baseline. Caffe [12] is used for the constructing the Siamese network.

Datasets. MSRC dataset [13] consists of 14 categories. Each category consists of 30 images of dimension 213 x 320. iCoseg dataset [8] consists of 38 categories. Each category consists of about 20 to 30 images, which are of 300 x 500 size.

Baselines and Parameter setting. We report results with two baselines. The first baseline involves training the Siamese network with pretrained ILSVRC [14] models. The weights are fine-tuned for target classes as in the datasets and then segmentation is performed on the clustered test set data. In the second baseline, we train the network on non-target classes and test the generalization ability on target classes. We evaluated on two objective measures: Precision ($P$) and Jaccard Similarity ($J$). $P$ indicates the fraction of the pixels in the segmented image common with the ground truth. $J$ is the intersection over union measure with the ground truth images.

We generated the object proposals using different methods and evaluated the performance on these metrics. The techniques used are Multiscale Combinatorial Grouping (MCG) [15], Selective Search (SS) [16], Objectness (Obj) [17], SalProp [18] and Edgeboxes [19]. We further perform a non-maximal suppression and near duplicate rejection in the proposal set. We preserved the top-10 object proposals, so that all the object instances in the images are covered. We used GoogLeNet architectures [20] for training the Siamese in our experiments. We used transfer learning, in which we initialized the weights with pre-trained model weights. We then fine-tuned the weights using back propagation technique. Siamese network is trained and the N-Dimensional (N=256) features are extracted for the test images. The N-Dimensional features are fed to ANNOY and similar object proposals are retrieved. The parameters used for Annoy library include number of trees, $n_{trees}$=350 and number of nodes to inspect during searching $search_{k}$=50. Similar object proposals are segmented using FCN based semantic segmentation as discussed in Sec. II-B.

We trained the Siamese architecture by employing the standard backpropagation on feed-forward nets by stochastic gradient descent with momentum to adjust the weights. The mini-batch size was set to 128, with an equal learning rate for all layers set to 0.01. The number of iterations is set as 100,000, contrastive loss margin as 1.

We also trained the Siamese network on datasets which contains similar (but not same) classes to iCoseg and MSRC datasets. We used Pascal [21], Animals [22] and Coseg-Rep [23] datasets to train the Siamese model and tested on iCoseg and MSRC datasets. Initially, we randomly selected positive and negative pairs for training the Siamese network. However, once most of the pairs are correctly learned, then using those pairs, Siamese cannot learn anymore. So, to address this issue, we used strategy of aggressive mining [24] for preparing hard negative and positive pairs.

Results. We divided iCoseg dataset into 80% training samples and 20% as testing set for each class. For MSRC dataset the split was 70%-30% (training-test). The results of the $P$ and $J$ are shown in Fig.2. It can be observed that Siamese network fed with MCG proposals outperforms all other object proposal generation techniques with the closest being SalProp followed by SS, Obj and Edgeboxes. For both the datasets, the average precision and Jaccard index over all the classes with MCG proposals is higher than SalProp technique with a gap on an average being 2.48% and 1.84% in $P$ and $J$ respectively.

The 256-D feature vector of the training set are visualized using t-SNE (t-Distributed Stochastic Neighbor Embedding) as shown in Fig.3. Firstly for high dimensional data, a probability distribution is built such that similar objects gets selected with high probability and dissimilar points have very low probability of being selected. In the second step, similar to a high-dimensional map, probability distribution over the points in the low-dimensional map is constructed. The color-coding shown in Fig. 3 helps in better separation of the classes compared to before fine-tuning. As can be seen, the results of clusters of classes are well separated with only few cluster of confusion.
We computed the average precision and jaccard similarity and compared with the other state-of-the-art methods in Tab. [11]. On testing with complete iCoseg dataset, we achieve a gain in \( P \) of 27.27\% (Joulin et al. [25]), 23.52\% (Kim et al. [26]), Quan et al. [27] outperform with a margin of \( P : 9.67\% \) and \( J : 13.15\% \) compared to the proposed technique (Siamese (MCG) + FCN segmentation). Similarly with MSRC dataset, we achieve a gain in \( P \) of 20\% (Joulin et al. [25]), 9.09\% (Jian et al. [28]), 44.82\% (Kim et al. [26]) and in \( J \) of 15.51\% (Yong Li [3]). Rubinstein et al. [29] outperform with a margin of \( P : 8.6\% \) and \( J : 1.47\% \) compared to the proposed technique (Siamese (MCG) + FCN segmentation). In FCN segmentation, we used VGGNet architecture, with FC layers replaced with convolutional layers. Deconvolutional layers are fixed using bilinear interpolation. We have abstained from using any auxiliary training and use the pretrained weights to avoid over-fitting in the FCN segmentation network. The object proposals that are similar and clustered together are fed as input to FCN segmentation to obtain the co-segmentation results. We consider only those object proposals for the cosegmentation task which have an intersection over union (IoU) score \( \text{IoU} \geq 0.5 \). Since, the segmentation is performed on the tight object proposals it segments the regions specific to the object class only and thus refrains from performing semantic segmentation over entire image. Owing to the performance boost by aggressive mining we achieve an average gain of 4\% and 2.52\% on both the datasets in \( P \) and \( J \) respectively over training with MCG proposals (Siamese (MCG) + FCN segmentation). Fig. 4 shows the qualitative results on few example classes in the datasets. However, it is important to note here that since we perform cosegmentation on the subset of images owing to the retrieval results we observe that there is a drop in performance with respect to few reported techniques. The advantage of the proposed technique over other compared techniques ([8], [25]–[27], [29]–[31]) involve: (i) co-segmenting without explicit knowledge of localization prior in the form of saliency map (ii) co-segmentation pipeline being formulated as clustering followed by segmentation of the similar object classes thus eliminating the need for providing as input the relevant set of class-specific images as required in graph-based co-segmentation techniques. The proposed method takes less than 100ms for the generation of 256-dimensional features from trained Siamese network.

![Fig. 4. Visual segmentation results on iCoseg and MSRC datasets. First three rows are classes of iCoseg (Cheetah, Panda, Taj-Mahal) and next two rows are MSRC (Car, Cow).](image)

We create a visual summary of the co-segmented similar objects. We preserved the Euclidean distances while retrieving the similar objects. Image is divided into different blocks and objects are placed such that the object with least Euclidean distance is at the center. A proper background is added to improve the visual appearance. Fig. 5 shows the sample collage results formed the Chair class in MSRC. A 512x512 image is divided into 10 blocks consisting a blue sky back-ground to form collage. Future work would be aimed to further improve the segmentation results and utilization of more cues for the relevance ranking in the collage-generation.

![Fig. 5. Example of Collage results for Chair class (MSRC).](image)

### Table I. Comparison of Average Precision and Jaccard Similarity with State-of-the-Art Methods. (*-" indicates that the metric has not been provided in the respective paper) on iCoseg Dataset

<table>
<thead>
<tr>
<th>Method</th>
<th>( P )</th>
<th>( J )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siamese (Edgeboxes) + FCN segmentation</td>
<td>0.76</td>
<td>0.64</td>
</tr>
<tr>
<td>Siamese (Obj) + FCN segmentation</td>
<td>0.78</td>
<td>0.62</td>
</tr>
<tr>
<td>Siamese (SS) + FCN segmentation</td>
<td>0.79</td>
<td>0.64</td>
</tr>
<tr>
<td>Siamese (SalProp) + FCN segmentation</td>
<td>0.79</td>
<td>0.64</td>
</tr>
<tr>
<td>Siamese (MCG) + FCN segmentation</td>
<td>0.81</td>
<td>0.654</td>
</tr>
</tbody>
</table>

**Fig. 4.** Visual segmentation results on iCoseg and MSRC datasets. First three rows are classes of iCoseg (Cheetah, Panda, Taj-Mahal) and next two rows are MSRC (Car, Cow).

**Fig. 5.** Example of Collage results for Chair class (MSRC).

**IV. Conclusion**

We addressed object cosegmentation and posed it as a clustering problem using deep Siamese network to align the similar images which are segmented using semantic segmentation. Additionally, we compared the performance of various object proposal generation schemes on Siamese architecture.
We performed extensive evaluation on iCoseg and MSRC dataset and demonstrated that the deep features can encode the commonness prior and thus provide a more discriminative representation for the features.

### References


### Table II. Comparison of Average Precision and Jaccard Similarity with state-of-the-art methods. (−) indicates that the metric has not been provided in the respective paper on MSRC dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>P</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siamese (Edgeboxes) + FCN segmentation</td>
<td>0.77</td>
<td>0.62</td>
</tr>
<tr>
<td>Siamese (Obj) + FCN segmentation</td>
<td>0.80</td>
<td>0.63</td>
</tr>
<tr>
<td>Siamese (SS) + FCN segmentation</td>
<td>0.81</td>
<td>0.64</td>
</tr>
<tr>
<td>Siamese (SalProp) + FCN segmentation</td>
<td>0.81</td>
<td>0.64</td>
</tr>
<tr>
<td>Siamese (MCG) + FCN segmentation</td>
<td>0.83</td>
<td>0.65</td>
</tr>
<tr>
<td>Siamese (MCG) + FCN segmentation-Aggressive mining</td>
<td>0.76</td>
<td>0.60</td>
</tr>
<tr>
<td>Siamese (MCG) + FCN segmentation-Aggressive mining</td>
<td>0.79</td>
<td>0.61</td>
</tr>
</tbody>
</table>
Oral Session J

Forensic science

Thursday May 17, 2018, 10:00 AM

Ali K. Hmood, Tamarafinide V. Dittimi and Ching Y. Suen

Counterfeit Coin Detection Using Stamp Features and Convolutional Neural Network

Dan Liu, Ching Yee Suen and Olga Ormandjieva

A Novel Way of Identifying Cyber Predators

Maryam Sharifi Rad, Saeed Khazaee and Ching Y. Suen

Counterfeit Coin Detection Based on Image Content By Fuzzy Association Rules Mining

Danny Roberge, Alain Beauchamp and Serge Levesque

Objective Identification of Bullets Based on 3D Pattern Matching and Line Counting Scores

Li Liu, Yue Lu and Ching Suen

Visual Detection of Fake Coins Using Fisher Vectors
Counterfeit Coin Detection Using Stamp Features and Convolutional Neural Network

Ali K. Hmood, Tamarafinide V. Dittimi, Ching Y. Suen

Centre for Pattern Recognition and Machine Intelligence (CENPARMI)

Computer Science and Software Engineering Department

Concordia University, Montreal, Canada

{a_alfraj, t_dittim, suen}@encs.concordia.ca

Abstract: This paper proposes a robust counterfeit coin detection method to tackle the advancements and sophistications of modern forger methods. The proposed method uses transfer learning by fine-tuning a pre-trained convolutional neural network (CNN) and then focuses on the features and measures of the characters on the coin surface. The fine-tuning process customizes the general image features from original images used for training the CNN i.e. natural image into adjusted features that are suitable for coins. On the other hand, characters represent one of the two major parts of the coin stamp and is the one used mainly by human vision system to recognize and authenticate coins. The ensemble method combines the classification results of two classifiers trained using features from convolutional layers and a third classifier trained on character features e.g. distances between characters, stroke width, height and depth of characters. The ensemble method achieved precision and recall rates as high as 99.5% and 99.3% respectively, demonstrating the robustness of combining fine-tuned CNN classification results and other selected features from characters in authenticating coins. The method is evaluated on a real life dataset of Danish coins as part of collaborative research with Danish authorities.

Keywords: Coin Authentication, Coin Stamp Features, Character Measures, Transfer Learning, Convolutional Neural Network.

I INTRODUCTION

Counterfeit coins have always been a concern to authorities throughout the years. In addition to highly valuable collectable and ancient coins, there are large numbers of spurious coins in circulation around the world with that number increasing yearly. A few solutions have been discussed in the literature on coin authentication with promising results. Yet, the coin authentication research field is still immature and the demands of a reliable and efficient method are increasing. Recently, a coin collector, Mike Marshall, informed Canadian authorities about spurious collectable Canadian coins available for sale on eBay’s website [1]. While, several governmental reports suggested that the currently circulated coins are susceptible to forgery and there are around 14 million spurious Pound coins in the UK circulated between 2003 and 2004 [2]. While more than 47 million spurious £1 coins in 2014 as the Royal Mint estimation reported. Most recently in 2017, Royal Mint found approximately one in thirty £1 coins in circulation is a counterfeit before they decide to replace the current coin by a new one. Therefore, governments are seeking solid solutions to overcome the counterfeiting problem. While, coin recognition aims to classify coins based on their actual time, value, denomination, and minting country [3]. In contrast, coin authentication works on differentiating genuine coins from spurious ones [4].

Traditionally, the coin authentication is carried out by human experts and the authentication is based on their personal experience. Over time several physical characteristics have been utilized such as the coin weight or diameter. These approaches have proved its lack of accuracy due to advancements in methods used by forgers to counterfeit coins. Therefore, the use of image processing for coin recognition and authentication is more accurate and promising. Unlike counterfeit detection system in other applications, coins are (1) subject to severe degradation due to circulation in our daily use; (2) no prior knowledge of coin orientation; (3) bump height of characters minted on the coin’s surface varies which affect the stroke sharpness and width; (4) different lighting sources highly affect the character appearance due to highlight and shadow variation; (5) different languages used for coins; (6) identical color of the characters and the background.

In this paper, a new coin authentication is proposed using the state-of-the-art deep learning method for image classification (CNN) along several other security features extracted from characters on the coin. CNN is a well-established image classification technique that extracts and trains an optimal image features set for a given classification task. CNN requires a very large training set even with transfer learning technique in order to achieve a high accuracy. In counterfeit coins research, such a large number of counterfeit coins for training CNN is not available due to
security reasons. Therefore, the proposed method uses a fine-tuned pre-trained CNN to classify and extract features to train another classifier while combines other features from characters that are minted on the surface to achieve higher precision. Fig. 1 represents the general framework of the proposed method. Since the method focuses also on extracting features from characters, therefore, the outer circle of the coin is located and transformed into rectangular shape to overcome the orientation problem. Then a set of morphological operations is performed to find the vertical and horizontal profiles, and then the adaptive mask is applied to initialize the height and width of characters. The mask is dynamic in terms of its size based on the relative height and width of each character. The character features along with the classification results of convolutional layer features have shown promising results in detecting counterfeit coins accurately.

II RELATED WORK

Several studies have been discussed in the literature to authenticate coins. Most of existing studies authenticate coins based on physical characteristics such as metal features, thickness, diameter, size, weight, conductivity, electromagnetic field properties, and piezoelectric characteristics [4]. However, new technologies have made it easier for forgers to duplicate a coin with the same physical characteristics [4]. Moreover, other researchers utilized sound and light to authenticate coins such as [5] that used frequencies obtained to discriminate spurious from genuine coins. However, these frequencies rely mainly on the type of metal. Thus, if the same metal has been used for the spurious coins, it will pass this test. On the other hand, other methods based on X-ray fluorescence have been proposed [4]. X-ray methods are harmful by nature and the use of X-ray fluorescence is impractical and expensive process [4]. In addition, these methods are also based on characteristics of metal type.

Therefore, the use of image processing and pattern recognition in coin authentication is introduced and revealed more accurate and promising results. In literature, much research has been conducted on coin recognition and proposed different image processing techniques and those methods are mostly applied coin extraction, feature extraction, and decision. Generally, there are two main methods for coin extraction: the edge-based methods and Hough transformation [6], the latter is also used after edge-based methods to find the center and the radius of coins to separate the coin from its background.

Feature extraction is the most crucial step in coin recognition systems where several methods studying many features set were proposed such as Edge-based statistical features in [6], local image features [7, 8] e.g. SIFT [7, 8] and SURF [8, 9], and texture features [10, 11]. The last step in a coin recognition system, decision, it decides where each coin belongs. Image processing techniques [10] and machine learning [4, 6, 7] are used to classify the set of features and recognize coins.

However, most of the computer-aided methods described above are designed to recognize and classify the coins into their actual time, political background, denomination, and minting country [6, 7, 8] while there are very few studies on coins authentication [4, 12, 13]. Generally, coin authentication methods employ similar image processing, pattern recognition, and machine learning techniques as the ones used for coin recognition.

Sun et al. [12] proposed a counterfeit coin detection algorithm based on the character properties such as the stroke width, height and width of the character, relative distances and angles between characters, as well as, local image features. However, the researchers conducted the experimental work on a dataset of 13 coins only and the method claims no guarantee to fit other coins due to the very limited experimental dataset.

Khazaee et al. [13] have presented a counterfeit coin detection for 3D coin images. They examined the outer circle of the coin where all characters and numbers occur using the height and depth information obtained by the 3D scanner to identify genuine coins. The method has shown a promising results when images from specialized 3D scanner are obtained. The access to such scanner is not feasible in daily life and it requires expertise to obtain 3D images of coins.

Recently, Liu et al. [4] proposed an image-based counterfeit coin detection system. The proposed system compares the local image features (keypoints) between the test coin and a set of predefined coins. The DOG detector is used to detect the keypoints which are described using the SIFT descriptor. Each comparison between the test image and the predefined image is stored as a vector in dissimilarity space. Finally, the SVM is used to classify the coins into genuine or fake class.

Convolutional neural networks (CNNs) have been the topic of interest to many researchers for its classification results and optimization. CNNs are a deep learning method that implicitly extract features from images and learn more comprehensive data from the image [14]. Training CNNs from scratch to perform an optimal feature extraction and training to classify images requires a large amount of images (tens of thousands if not millions) which is not feasible in our study. Hence, many researchers with limited dataset have used a pre-trained CNNs on different image dataset i.e. natural image and transfer learning from those CNNs to the new dataset. The transfer learning methods in other domains such as face recognition or medical image classification reported the state-of-the-art results.

In this research, we used the fine-tuning process to transfer learning from GoogLeNet architecture introduced by Szegady et al. [15] to fit our coins dataset. However, the feature sets from GoogLeNet used to train a softmax and SVM classifiers. In addition, the classification results of GoogLeNet are combined with another SVM classification results of other features extracted from characters i.e. stroke width, character height and width, spaces between characters, and edge smoothness to further classify images.
accurately. In order to extract features from characters, we transform the coin from circular into rectangular shape and then perform morphological operations to compute the vertical and horizontal projection profiles and apply dynamic adaptive mask.

![Diagram of the proposed method](image)

**III Method**

The growing needs for coin authentication system have received the attention of researchers to develop an efficient and cost-effective methods. The previous methods based on physical characteristics and features of coins e.g. weight and size are naïve compared to the new methods used by forgers nowadays. Therefore, this paper proposes a new counterfeit coin detection based on combining convolutional neural networks and other features mainly considered by human experts. Like coin recognition systems, the proposed method starts by coin scaling (extraction) where the goal is to scale the coin to fit the whole image and remove the unnecessary background and marginal information.

The GoogLeNet architecture is then used to fine-tune the features to better extract features from coins. The features from convolution layers are used to train two classifiers, the softmax and SVM. On the other hand, we used another SVM classifier trained on a set of character features from the coin i.e. characters height, width, stroke size, edge smoothness, spaces between characters, and distances of characters from top and bottom. A dynamic adaptive mask is used to handle the measures of each character separately. The dynamicity of the adaptive mask is achieved through finding the height and width of each character and then decide the mask size w.r.t. character size. Those features are extracted only after applying the straightening algorithm which aims at transforming the circular shape of coin into rectangular shape. Here to mention that only part of the coin image where the text appears is transformed see Fig. 2(c). The posterior probabilities of softmax and two SVM classifiers are used to determine the actual class of the coins.

**A. Preprocessing**

Most of coin recognition, grading, and authentication systems start with coin scaling which is an essential step to remove marginal and unwanted background pixels. Some of the proposed coin recognition solutions in the literature used a simple threshold to discriminate the coin pixels from other marginal pixels. However, this is not always applicable and it only fits images with high variances between coin and marginal pixels. In this research, all edges $E$ from image $I(x, y)$ are detected using Canny edge detection and morphological structuring elements i.e. circular shape is applied to the edge image $I_E$, one to many circular shapes are presented then and the method starts identifying a set of center points $p(x, y)$ from $I_E(x, y)$ also set of radius point $r(x, y)$ corresponding to each circle whose center $p(x, y)$. The largest circle $c_i$ with $p_i(x, y)$ and $r_i(x, y)$ is selected as a candidate circular shape of the coin. An adaptive mask is then placed over the coin w.r.t. the coin size and the image is cropped to fit the circular shape of the coin. There are still some noisy background pixels appear on the image due to the circular shape of the coin, hence, all pixels occurring outside the margins of the adaptive mask are blacked out as shown in Fig. 2(b).

![Image](image)

**B. Convolutional Neural Network**

Convolutional neural networks (CNNs) are the new era in machine learning. CNNs are a machine learning technique that uses deeper networks to implicitly extract features from images. Deep networks comprise different layers to extract more sophisticated information to better understand the image and need no engineering methods to extract features. Different architectures have been discussed in the literature w.r.t. number of layers and their
GoogLeNet architecture introduced a network-within-network concept by using a new module “Inception” that is a subnetwork containing a number of convolutional filters of different sizes and dimensions working in parallel and concatenating their outputs. GoogLeNet has 9 “Inception” layers and each “Inception” layer comprises 6 convolution layers and 1 pooling layer which makes it a very deep network. The CNN architecture works on capturing a sparse representation of image information while gradually reducing dimensionality. The architecture consist of 22 layers excluding the 5 pooling layers. GoogLeNet have shown a better results to other CNN architecture specifically AlexNet while needs fewer trainable weights [15].

GoogLeNet has originally trained to extract features and classify natural images into 1000 classes. The fine-tuning process that is used to transfer learning keeps the first two convolution layers fixed, as those layers extract generic type of features e.g. edges, and fine-tune all intermediate layers while replaces the last fully connected (FC) layer intended for the 1000 classes of the original dataset used to train GoogLeNet into a new FC layer of two classes to fit our dataset of genuine and counterfeit classes. The original filter weights of natural image are fine-tuned to optimize weights of coins dataset through back-propagation and stochastic gradient descent to find optimal weights. On the other hand, the feature vectors from the fine-tuned layers are also used to train SVM classifier beside the new FC layer that uses softmax as activation function.

Assume that coin dataset $D$ of $x$ images, fine-tuning is an iterative process to update the filter weights $w$ to reach the minimum error rate. As suggested in the literature, we reduced the learning rate and the best results achieved at learning rate $\ell = 6 \times 10^{-4}$ for our dataset.

### C. Extraction Character Features

Characters represent a major part of the coin and they are the main part used by human to recognize coins and by experts to authenticate them. The most common method to forge a coin is to strike a coin using a fake die that is in return molded (designed) from original coin stamp. This method yield in a small variation in salient width which is reflected as the edges in a coin image. In addition, extracting features from characters is influenced by the findings from the literature where some counterfeit coin detection methods proposed using features from characters as part of the methods. To extract features from characters accurately, we have used the same work we proposed in character extraction from coins method [3] to locate coin but not extracting them as shown in Fig. 3(a).

After characters are located on the coin a set of features are extracted and used to classify coins. The stroke width, edge smoothness, character height and width, number of pixels, and spaces between characters. Stroke width calculation is inspired by the stroke width transformation method [16] where for every edge boundary pixel $i$, the algorithm trace the neighbor pixels until it reaches another edge boundary pixel $j$ Fig. 3(b). However, the proposed technique works on reading the pixels horizontally and not vertically; also we consider the other gradient direction other than the actual stroke width such as the line between pixel $l$ and $m$ in Fig. 3(b). The number of pixels between each edge boundary pixels is stored as stroke width. The character height and width is specified by the horizontal and vertical profiles that specify the size of the dynamic adaptive mask around each character.

On the other hand, the total number of edge pixels (stroke pixels) is calculated as there is deviation in character sizes between character of the genuine and counterfeit coins. In addition, the spaces between characters are calculated by taking the distance between the centers of every two immediate adaptive masks Fig. 3(d). Finally, the edge smoothness is found by first applying thinning algorithm to reach a one pixel edges of the character stroke as shown in Fig. 3(c) and then the number of thinned edge pixels is calculated as well as the pixels contained in the thinned edges.

![Feature Extraction Techniques from Characters](image)

**Fig. 3:** Feature Extraction Techniques from Characters

The sets of those features are then used for training a linear support vector machine (SVM) since we have only genuine and counterfeit classes. SVM works on finding a hyperplane separation with maximal margins between two classes. We employed the linear SVM with radial basis function (RBF) kernel to authenticate coins.

### D. Ensemble Learning

The classification results along with their posterior probabilities of the three classifiers are used to decide the class of a coin image. The feature sets from fine-tuned GoogLeNet are used to 1) train the FC layer and the softmax function which discriminates classes based on the largest values in each feature vector while suppresses the less significant values. 2) train a one-vs-one SVM classifier. Additionally, the character feature sets are used to train another on-vs-one SVM classifier. The posterior probabilities for both SVM classifiers are estimated and used with posterior probability of softmax function to
improve the final classification results using  
\[ \hat{c} = \frac{\sum_k p_{ik}(c)}{k} \]
Where \( c \) is the classification result of classifier \( k \) for image \( i \). \( p_{ik}(c) \) is the posterior probability of \( c \) for image \( i \) given by classifier \( k \). \( \hat{c} \) is the final classification results given to image \( i \).

**IV. EXPERIMENTAL RESULTS AND DISCUSSIONS**

The proposed method was evaluated on different datasets that contain genuine and counterfeit Danish coins. The Danish coins datasets comprise the obverse side of Denmark 20 Korner of different years as shown in Table I. The datasets are randomly partitioned into training, validation and test sets. Training set is used to train each classifier while the evaluation of classifiers is conducted using the testing set. In addition, the validation set is used to test and set the parameter settings. In addition, the impact of using any single classification results and the parameters involved in achieving the proposed method is considered in the evaluation process.

We evaluated the three classifier results and the ensemble methods proposed in this paper. The fine-tuned GoogLeNet with softmax (FTmax), fine-tuned GoogLeNet with SVM (FTsvm), and character features with SVM (CFsvm) are the three classification methods based on different feature set where the first two are the softmax and SVM based on the convolutional layers feature sets while the latter is the SVM classifier based on character feature sets.

![Table I](image)

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Training Set</th>
<th>Validation Set</th>
<th>Testing Set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Genuine</td>
<td>Counterfeit</td>
<td>Genuine</td>
</tr>
<tr>
<td>Danish 1990</td>
<td>857</td>
<td>586</td>
<td>1267</td>
</tr>
<tr>
<td>Danish 1991</td>
<td>857</td>
<td>586</td>
<td>1267</td>
</tr>
<tr>
<td>Danish 1996</td>
<td>857</td>
<td>586</td>
<td>1267</td>
</tr>
<tr>
<td>Danish 2008</td>
<td>802</td>
<td>587</td>
<td>1050</td>
</tr>
</tbody>
</table>

On the other hand, the proposed method (PM) of ensemble classification probabilities of the three classifiers is shown to improve the prediction of counterfeit coins as illustrated below. Fig. 4 illustrates the precision results of the three classifiers and the proposed ensemble method. The figure depicts the accuracy results of the four dataset used in evaluation of the three classification method separately and the result of ensemble method proposed in this paper. The character feature sets are classified using binary SVM and are shown to return the lowest precision rate among other methods. The reported lower results can be seen as due the lower number features than convolutional layer features, as well as, confirm the findings from literature of convolutional layers show a better representation of image features.

![Fig. 4](image)

![Fig. 5](image)

Fig. 4: Precision Rates of Three Classification Results and the Proposed Method

Followed by fine-tuned GoogLeNet feature set with softmax function that returned an improved results between 1%-4.1% but lower than fine-tuned GoogLeNet feature set with SVM where the latter shows the best classification results among the other two classification methods where approximately 0.5% improvement over FTmax results. Our ensemble model have returned as high as 99.5% precision rate for Danish 1996 dataset.

We have also evaluated the number of correctly classified images using different combination of two methods among the four methods. Fig. 5 illustrates the normalized results of correctly classified coin images by a pair of methods.

The results suggest combing any two methods improves the classification results. The fine-tuned GoogLeNet with softmax and fine-tuned GoogLeNet with SVM combination showed a better classification results than using any single method, as well as, showed 0.74% higher true positives than a combination of fine-tuned GoogLeNet with softmax and character features with SVM.

On the other hand, the proposed method achieved the highest number of correctly classified images when combined by any other method where the combination of the proposed method and fine-tuned GoogLeNet with SVM showed the highest normalization result of true positive classification among the other combinations of 0.97.

Additionally, we compare the proposed method to the methods described in the literature. Table II shows a
Comparison of the proposed method to the most recent related work from the literature. The proposed system achieved a higher \textit{f-measure} value than the state-of-the-art methods. The character feature sets were influenced by the results of the literature that achieved a high accuracy based on character features. However, the use of the whole coin features using the convolutional layers of the fine-tuned pre-trained GoogLeNet have achieved a higher accuracy than features from characters only. In addition, the proposed ensemble method of combining the results of three classifiers achieved even higher accuracy and was capable of correctly distinguish counterfeit coins from genuine with a high precision rate.

**CONCLUSIONS**

In this paper, we studied the problem of counterfeit coin detection and proposed an ensemble method of three classification results. The proposed method used a fine-tune process to optimize feature extractors of pretrained CNN and transfer learning to improve feature extraction from our coin dataset. The extracted features from fine-tuned GoogLeNet were used to train two classifiers, softmax and SVM, and define the posture probability of their classification results. In addition, sets of features were extracted from characters minted on the coin and these sets were used to train another SVM classifier. The proposed solution was evaluated on different coin datasets that comprise genuine and counterfeit coins. The proposed ensemble method was experimentally proved to be capable of correctly classify majority of coin images. Experimental results suggest the use of ensemble method can better represent the image information and distinguish between coin images of different classes.

<table>
<thead>
<tr>
<th>Paper</th>
<th>Method</th>
<th>Dataset</th>
<th>Accuracy (\textit{f-measure})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sun et al. [12]</td>
<td>Character and shape features</td>
<td>11 genuine, 2 fake</td>
<td>~97%</td>
</tr>
<tr>
<td>Khazaee et al. [13]</td>
<td>Features of image</td>
<td>322 genuine, 162 fake</td>
<td>99.3%</td>
</tr>
<tr>
<td>Proposed method</td>
<td>Ensemble method</td>
<td>7250 genuine, 4557 fake</td>
<td>99.4%</td>
</tr>
</tbody>
</table>

**ACKNOWLEDGMENT**

The research is supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC). The authors sincerely acknowledge the assistance and cooperation of the Centre for Pattern Recognition and Machine Intelligence (CENPARMI) members at Concordia University.

**V REFERENCES**


A Novel Way of Identifying Cyber Predators

Dan Liu
CENPARMI, CS & SE Department
Concordia University
Montreal, Canada
1_dan17@encs.concordia.ca

Ching Yee Suen
CENPARMI, CS & SE Department
Concordia University
Montreal, Canada
suen@cs.concordia.ca

Olga Ormandjieva
CENPARMI, CS & SE Department
Concordia University
Montreal, Canada
ormandj@cse.concordia.ca

Abstract—Recurrent Neural Networks with Long Short-Term Memory cell (LSTM-RNN) have impressive ability in sequence data processing, particularly for language model building and text classification. This research proposes the combination of sentiment analysis, new approach of sentence vectors and LSTM-RNN as a novel way for Sexual Predator Identification (SPI). LSTM-RNN language model is applied to generate sentence vectors which are the last hidden states in the language model. Sentence vectors are fed into another LSTM-RNN classifier, so as to capture suspicious conversations. Hidden state enables to generate vectors for sentences never seen before. Fasttext is used to filter the contents of conversations and generate a sentiment score so as to identify potential predators. The experiment achieves a record-breaking accuracy and precision of 100% with recall of 81.10%, exceeding the top-ranked result in the SPI competition.

Keywords—Sentence vectors, RNN, LSTM, Sentiment Analysis

I. INTRODUCTION

Greater popularity of social networks gives rise to cyber-criminal activities conducted by Sexual Predators (SPs). In this context, PAN initiated the Sexual Predator Identification (SPI) Task in 2012 [1]. PAN collects and shares an overwhelming amount of data on online chats, inside which there are predators, to facilitate research in predator behaviors.

There are two separate tasks in SPI, namely, identification of SPs inside chats and highlighting specific SP's lines in chats. The research reported in this paper focuses on the first task. As [1] indicated, the first step is to find out which conversations are suspicious, then identify such conversations to belong to which author, [2], [3], and [4] use a similar method to identify the predators.

However, the organizer set the goal for SPI task as creating a large and realistic dataset. The side effects of realistic data is high noise level, unbalanced training samples, and various lengths of conversations (from 1 to more than 500). More specifically, there are lots of general and sex-related conversations, while among them only a few involve SPs. Furthermore, there are a lot of chat abbreviations and cyber slangs in conversations, such as "ur" for "your", "yr" for "year", "sorryyy" for "sorry", to name a few. Such words are crucial to feature selection and should be considered in the process of feature selection. Therefore, traditional machine learning methods cannot achieve satisfying performance unless with data truncation using numerous rules. Even if n-gram is used, with hundreds of thousands of conversations, the noise will result in extreme sparsity, and the performance will be weakened consequently [5].

LSTM-RNN [6] sentence vectors are introduced to solve the above-mentioned noise and performance problems. Different from n-gram, sentence vectors are able to capture sentence features more efficiently and compress the size of input data, as the classifier will only take sentences, instead of words, as features. Meanwhile, LSTM-RNN classifier can also be used for suspicious conversation detection (SCD) as it is good to learn long-term dependencies in time series data. The experiment generated an accuracy rate of 99.43% on SCD and 98.35% on SPI, respectively. Finally, 206 out of 254 predators were identified by the intersection of two classifiers with zero error, which exceeded the best result [2] of the official ranking (203 out of 254 with 3 misclassifications).

The contributions of this paper are three-fold, namely: i) LSTM-RNN is introduced to generate sentence vectors especially for sentences never seen before with known words; ii) IMDB reviews [7] is used to test the performance of sentence vectors model, and iii) sentiment score is introduced to improve the sexual predators identification performance.

The rest of the paper is organized as follows: the related work is summarized in section II. Our approach is explained in section III. Section IV describes the experimental work carried on in this research. Finally, section V, VI concludes the paper.

II. RELATED WORK

A common strategy for SPI is the use of two classifiers. The first classifier will detect suspicious conversations which can be seen as positive (with predators) or negative (no predators involved) [2, 3, 4]. However, very complex and specific rules were applied to remove noise or to extract features. Especially, in [2], only about 10% samples remained for training and testing. Such removal could influence the generalization ability of the classifier. Manual rules for features extraction in [8, 9, 10] will reduce the stability as only samples that match the rules can be classified. Neural network language model approach can conquer those problems mentioned above. The second classifier is about predator identification. Support Vector Machines (SVM) [8, 11], Naive Bayes [12, 13] and other classical machine learning approaches [3, 5, 9, 14] were introduced. From official rank [1], those approaches whose precision was greater than 90% had a lower recall (less than 80%).

Neural network language model is introduced in [15]. After that a series of derived version [16, 17, 18], i.e. word embedding, sentence embedding etc. is widely applied in Natural Language Processing (NLP) work. For the sentence embedding or document embedding, it shows very strong ability in NLP tasks. Because it is hard for those language models represent sentences never seen before, the new sentence which is not in the training
dataset will be a problem. Unless those embeddings, [19] proposed a more compatible method with sequence to sequence model to vectorize sentences based on previous and next sentences.

There are two mainly types of sentiment analysis task in NLP area online service rating [20] and movie review [7]. Severny and Moschitti [21] used the distance to margin of SVM as sentiment score, the larger it is the more positive or negative it will be. Deep neural network is also a popular area for sentiment analysis. Reference [22] compared the performance of different neural network models, such as NBSVM-bi [23] and LSTM-RNN etc. Although the deep neural networks have strong capability on NLP tasks, the training cost of deep neural network cannot be neglected. Fasttext [24], is a neural network with shallow and beautiful structure which had excellent performance on sentiment analysis and language model tasks.

III. APPROACH

The experiment involves three types of neural networks. Firstly, the LSTM-RNN-based language model, which is used to express the relation inside a sentence. The hidden state of LSTM-RNN of each sentence will be used as sentence vectors. Secondly, a two-layer LSTM-RNN classifier is used to find suspicious conversations by learning the dependencies among sentences in a conversation. Each sentence in a conversation will be regarded as a single timestep and fed into the classifier. Lastly, following the detection of suspicious conversations, a Fasttext-based sentiment classifier is introduced to identify SPs. The conversations are split into different groups by author. In this way, the SPs can be identified sentiment score.

A. Processing

PAN2012 dataset contains a great number of chat abbreviations, cyber slangs, emoticons and conversations of various lengths, which will increase the perplexity of language model. To predict the next word more accurately, noise removal and replacement are indispensable. However, some words or abbreviations may convey important information, for example, yrs means years, ur means your, etc., therefore recovery of these abbreviations is necessary. Those strategies are listed below:

- Replace the number by symbol 00NUM.
- Replace the words longer than 30 characters by 00LW.
- Replace the URL with symbol 00URL in the data.
- Remove all non-ascii chars.
- Remove all emoticons.
- Recover popular yet unofficial abbreviations.

B. Recurrent Neural Network

Recurrent neural networks (RNNs) are neural network models that process elements of a sequence one by one and learn the dependencies among previous inputs. There are three layers, i.e. Input X, Hidden S, and Output Y. The input of RNN at time step t is \( x_t \in \mathbb{R}^n \) and the hidden state is \( s_t \in \mathbb{R}^m \).

\[
s_t = f(Wx_t + Us_{t-1}) \tag{1}
\]

\( s_t \) is calculated by (1), based on the previous hidden state \( s_{t-1} \) and the current input step \( x_t \) where the function \( f \) is a nonlinear function. The output is (2).

\[
y_t = \text{Softmax}(Vs_t) \tag{2}
\]

RNN is featured by its ability to capture dependencies in sequences and share the same parameters \((U, V, W)\) throughout all steps. Theoretically, RNN learns through all previous timestep, however, due to vanishing gradients problem [25], it is hard to capture long-term dependencies.

LSTM-RNN proposes a gating mechanism to avoid vanishing gradients problem. More specifically, a new state \( c_t \) is introduced to calculate hidden state \( s_t \) (Fig. 1). \( c_t \) and \( s_t \) are calculated as below:

\[
i_t = \sigma(x_tU_i + s_{t-1}W_i) \tag{3}
\]

\[
f_t = \sigma(x_tU_f + s_{t-1}W_f) \tag{4}
\]

\[
o_t = \sigma(x_tU_o + s_{t-1}W_o) \tag{5}
\]

\[
g_t = \tanh(x_tU_g + s_{t-1}W_g) \tag{6}
\]

\[
c_t = f_t \cdot c_{t-1} + i_t \cdot g_t \tag{7}
\]

\[
s_t = o_t \cdot \tanh(c_t) \tag{8}
\]

where \( i_t \), \( f_t \) and \( o_t \) are input, forget gate and output gate. \( g_t \) is the hidden state like \( s_t \) in RNN which is used to compute new \( s_t \).

C. Recurrent neural network language model

The neural network language model takes a word sequence \( W = [w_1, ..., w_t] \), \( w_t \in V \) where \( V \) is the vocabulary set as input and learns to predict the probability \( p(w_{t+1} | w_1, ..., w_t) \) of the next word \( w_{t+1} \) by applying Softmax activation function at the output layer. \( P(w_{t+1} | w_1, ..., w_t) = \text{Softmax}(s_t^T e_{w_t}) \) where \( e \in E[e_{w_1}, ..., e_{w_t}] \) [15]. This input is mapped to vectors \( e_{w_t} \) in feature space within the neural network. The training of neural network uses error back-propagation algorithm over time to maximize the log-likelihood (9) of training data.

\[
L(\theta) = \sum_t \log p(w_t | w_{t-n+1}, ..., w_{t-1}) \tag{9}
\]
D. Sentence vectors

LSTM-RNN neural network language model with two hidden layers is used to generate sentence vectors (Fig. 2). Specifically, the last time step hidden state in LSTM-RNN language model $s_t$ is used to represent the input sequence $\{w_1, \ldots, w_t\}$. The first layer of this model is the embedding layer that represents words in dense vectors. The second and third layers are LSTM-RNN that learn the dependencies among the words in sentences. The final layer is the Softmax layer, a multinomial logistic regression layer used to solve multi-class prediction problems. Compared with word embedding, the sentence vectors presentation can reduce the length of inputs. LSTM-RNN-based sentence vectors have the advantages of being able to capture the dependency and compress the size of conversations. In the meantime, those words with term-frequency of less than 10 are removed as noise and the remaining words are sorted by term frequency-inverse document frequency (TF-IDF) weights.

E. Conversation Classification

With regard to suspicious conversation identification, LSTM-RNN has strong ability to learn the long-term dependencies among timesteps, which means it can capture relations among sentences that contain the features of predators. Conversations with sentence vectors are input into a three-layer LSTM-RNN-based model and the latter will learn the context features (Fig. 3). Considering different number of sentences in conversations (from 1 to more than 500), those extra-long conversations will be padded by zeros and then split into parts, each with an equal length of 100 (an experience-based value). This strategy will prevent underfitting in LSTM-RNN model when processing long conversations as there are only a few of them. Due to the well-distributed features in suspicious conversation, a predator is very likely to carry out criminal activities throughout an entire conversation.

F. Author Classification

Different from LSTM-RNN, which is good at capturing time series features, Fasttext is a very shallow neural network capable of global feature extraction. Once the suspicious conversations are identified by LSTM-RNN model, the Fasttext-based classifier can identify SPs among the authors. In addition, those conversations only containing predators will not be deleted because there might be useful features inside such conversations. In order to improve the accuracy of SPs identification, sentiment score is assigned. There are three types of scores, i.e. P, V, N, to differentiate authors from predators, victims, and normal users. The output of Softmax layer is the most ideal score model. An author with a higher score in a scoring type among the three will be classified into that group. It is unlikely for two predators to appear in the same conversation, therefore an author with the highest score in P category will be identified as predator. As there might be conversations initiated by the same author at different time, the sentiment score of those authors is averaged.

IV. EXPERIMENTS

For the purpose of detecting SPs, firstly, PAN2012 dataset is used to train the neural network language model and the SCD classifier. After that, Fasttext classifier is trained on PAN2012 dataset grouping by authors. The IMDB sentiment review dataset is used to evaluate the performance of the model, i.e., sentence vectors methodology.

A. Dataset

The performance of the models is evaluated on two datasets, PAN2012 dataset and Stanford Large Movie Review Dataset (IMDB sentiment review dataset) [7]. Performance will be measured against existing publications on sentiment classification tasks.

1) PAN2012 Dataset

PAN2012 dataset is provided in the context of Sexual Predator Identification (SPI) Task in 2012 initiated by PAN (Plagiarism analysis, authorship identification, and near-duplicate detection) lab. In the training dataset, there are 66,927 chat conversations with over 97,000 different users and only 142 of them are SPs. The test dataset contains 155,128 chat conversations with over 218,000 different users and only 254 of them are SPs (TABLE I).

<table>
<thead>
<tr>
<th>Type</th>
<th>Training</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Original</td>
<td>Filtered</td>
</tr>
<tr>
<td>Positive</td>
<td>2,016</td>
<td>1,088</td>
</tr>
<tr>
<td>Negative</td>
<td>64,911</td>
<td>52,854</td>
</tr>
<tr>
<td>Non-predators</td>
<td>97,547</td>
<td>97,291</td>
</tr>
<tr>
<td>Predators</td>
<td>142</td>
<td>138</td>
</tr>
</tbody>
</table>
2) IMDB movie reviews

IMDB Large Movie Review Dataset provides 50,000 binary labeled reviews extracted from IMDB. In this dataset, highly polar movie reviews, with a score lower than 4 or higher than 7 on a scale of 10, is split evenly into 25,000 training samples and 25,000 test samples. The overall distribution of labels is balanced. The distribution of sentence length of reviews is shown below (TABLE IV).

<table>
<thead>
<tr>
<th>Number of...</th>
<th>Original data</th>
<th>Filtered data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chat conversations</td>
<td>66,928</td>
<td>6,588</td>
</tr>
<tr>
<td>Users</td>
<td>97,690</td>
<td>11,038</td>
</tr>
<tr>
<td>Sexual Predators</td>
<td>148</td>
<td>136</td>
</tr>
</tbody>
</table>

B. Experimental setup

1) Suspicous conversation detection

For the SCD task, LSTM-RNN language model is trained with the architecture shown in (Fig. 3). The sentence vectors are the last hidden state of LSTM-RNN language model. Each conversation being represented by a group of sentence vectors is fed into a new LSTM classifier. The performance of this classifier is demonstrated in (Fig. 4). In the methodologies mentioned above, one embedding layer, two LSTM-RNN layers with 200 units and 50 timesteps as well as a Softmax layer are implemented on Tensorflow framework for language model. The SCD classifier is implemented on Keras framework has the similar structure as LSTM-RNN language model, except that SCD classifier replaces Softmax layer with sigmoid layer.

2) IMDB sentiment task

The IMDB sentiment task is introduced for evaluating the classification performance of sentence vectors. This task shares the same structure and configuration as SCD task.

Related performance data is shown in Fig. 6, TABLE XI, and TABLE XII.

V. Results

A. Language model generated from PAN2012

By using a LSTM-RNN neural language model with two hidden layers, together with 35 timesteps and 200 hidden units, the perplexity of language model on test dataset reached 10.948 after 30 iterations.

B. Suspicious Conversation Detection

The result of suspicious conversation detection task is shown in Fig. 4 and TABLE V. It is obvious that the best result is obtained at epoch 5. The accuracy of sentence-vector model is 99.43%, exceeding the accuracy of 98.83% with SVM obtained by [2] (TABLE VI).

![Fig. 4. The performance of SCD classifier.](image)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Weighting</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F-1 Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>binary</td>
<td>0.9848</td>
<td>0.9361</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>tf-idf</td>
<td><strong>0.9883</strong></td>
<td><strong>0.9516</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td>binary</td>
<td>0.9874</td>
<td>0.9464</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td>tf-idf</td>
<td>0.9825</td>
<td>0.9254</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

C. Sexual Predator Identification

In the sexual predator identification task, the performance of Fasttext model is very stable. The training procedure is completed in 20 minutes. The sentiment score is generated from Softmax layer of the Fasttext model. The score of the same participants in different conversations is averaged. The result (TABLE X) shows that all predators have a very high sentiment score compared with victims.
D. IMDB sentiment reviews

1) Language model generated from IMDB reviews

The language model is built with the same method as SCD and the test perplexity is 126.903 which is not so good. The perplexity is significantly higher than SCD’s. The timestep of this model is 50, which is longer than SCD’s as the average number of sentence per input of IMDB dataset is larger (see TABLE II and TABLE IV).

2) Sentiment results of IMDB

Compared to the training result, the test result of the sentence-vector model on IMDB dataset indicates that the performance of this model is very unstable. Although the accuracy of 83.2% with sentence-vector model, the sentence-vector reduces the length of the inputs and accelerates the training and test speed.

VI. CONCLUSIONS

This paper presents a combined method to identify sexual predators. The sentiment score from Softmax layer outputs is crucial in the final identification step. The approach of taking LSTM-RNN last hidden state as sentence vectors is highly efficiency as the long conversations are shortened by sentence vectors. The reason of sentence-vectors-based classifier does not work well on IMDB dataset could be that the perplexity of its language model is too high to represent the sentence. In the future work, other language models will be applied to reduce the perplexity to see if the accuracy will be enhanced. In the

TABLE XII. COMPARISON ON THE IMDB SENTIMENT TASK.

<table>
<thead>
<tr>
<th>Model</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>NBSVM-bi (Wang and Manning, 2012)</td>
<td>N/A</td>
<td>91.2%</td>
</tr>
<tr>
<td>Paragraph Vector (Le and Mikolov, 2014)</td>
<td>N/A</td>
<td>92.7%</td>
</tr>
<tr>
<td>Paragraph Vector (Hong and Fang, 2015)</td>
<td>97.1%</td>
<td>94.5%</td>
</tr>
<tr>
<td>Sentence Vector + 2-layers-LSTM-RNN</td>
<td>83.4%</td>
<td>83.2%</td>
</tr>
</tbody>
</table>
meantime, attention mechanism [26] in neural network can also be introduced to detect keywords in predators’ conversations.

ACKNOWLEDGMENT

We would like to thank Natural Sciences and Engineering Research Council of Canada for supporting this project.

REFERENCES

Counterfeit Coin Detection Based on Image Content
By Fuzzy Association Rules Mining

Maryam Sharifi Rad
Department Of Computer Science and Software Engineering
CENPARMI, Concordia University
Montreal, Canada
sey_shar@encs.concordia.ca

Saeed Khazaee
Department Of Computer Science and Software Engineering
CENPARMI, Concordia University
Montreal, Canada
s_khaza@encs.concordia.ca

Ching Y. Suen
Department Of Computer Science and Software Engineering
CENPARMI, Concordia University
Montreal, Canada
suen@encs.concordia.ca

Abstract—In this paper, a new framework is proposed for counterfeit coin detection that shows the effectiveness of image mining techniques in this field. The proposed system is developed in two modules. At the first module, the segmentation of digital coin images is applied to find the region of interest (ROI). These ROIs serve as inputs to the next stage. At the next module, image mining is applied to find frequent image patterns present in coin images using fuzzy association rules mining. The experimental results show that fuzzy association rules mining based on image content is feasible and gives strong rules that can be further used for effective classification of coin images.

Keywords—Image mining, fuzzy association rules, feature extraction, fake coin detection

I. INTRODUCTION

With the growth of data and huge number of digital photographs, data mining and knowledge discovery have become a priority. In recent years, image mining which can automatically review meaningful information from image dataset, has established its position as a prominent and important research area. Image mining techniques can discuss the extraction of implicit knowledge and the relationship among patterns stored in the image databases. With the consideration of this trend and the extensive amount of data involved in all kinds of digital images, image mining approaches seem to be appropriate for this purpose. Basically, image mining is an interdisciplinary endeavor that draws upon computer vision, image processing, image retrieval, machine learning, and artificial intelligence, etc [1].

With the widespread use of automatic systems, such as vending machines, parking meters, banks and so on, coin recognition plays a vital role in all aspects of daily life. Moreover, museums have increased the demand on automatic systems to classify historical coins [2]. However, in recent years, a lot of illegal counterfeiting rings manufacture and sell fake coins, which have caused great loss and damage to the society [3]. Basically, counterfeit coin detection is a difficult process because of widely varying input patterns, cluttered images, and various rotations which are the great challenges. As a significant topic of security, counterfeit coin detection has become the focus of research in the field of numismatics.

Up to now, several methods based on image processing techniques and classification algorithms have been developed [3-5]. It is worthy to note that many research have been conducted on coins’ colors and radius based features to detect fake coins. However, there is a growing interest in counterfeit coin detection community toward the application of image mining techniques in this field. Several interesting research involving image mining have been reported [1, 6-9]. Although several research have been developed based on different approaches for mining of images to extract strong association rules, still it is a challenging task. Hence, we propose a method based on fuzzy association rules mining to detect fake coins. To our knowledge, the potential of these classification techniques based on fuzzy association rules mining has not been applied to coin image databases for counterfeit coin detection and is still an open field of research. In this study, the feasibility of applying fuzzy association rules mining for counterfeit coin detection will be demonstrated. It is noted that fuzzy logic can be used to identify complex pattern or structural variations in image datasets.

The rest of this paper is organized as follows. In Section 2, the concepts and preliminaries related to the fuzzy association rules mining are described. In Section 3, the proposed approach for counterfeit coin detection is introduced in detail. In Section 4, the experimental results are given to show the effectiveness of the proposed method. Finally, Section 5 draws conclusions.

II. FOUNDATIONS OF RULES MINING

A. Fuzzy Association Rules Mining

The objective of data mining is to obtain useful and non-explicit information from data stored in large repositories [10]. Association rules mining is one of the most important topics in data mining research. Mining fuzzy association rules is the discovery of association rules by using fuzzy set concepts, such that databases with both categorical and quantitative features can be handled [11]. Agrawal et al. [12] have proposed Apriori algorithm which is the best known fundamental algorithm to find association rules. During past years, several research have been conducted to extract association rules from datasets. In spite of the novelties of the proposed methods, it is clear that they were not robust enough to distinguish the sharp boundary
In other words, they either ignore or over-emphasize the elements near the boundary of intervals in the mining process. As a remedy to the sharp boundary problem, the fuzzy set concept, introduced by Zadeh [13] has been used more frequently in mining association rules. It is noted that the fuzzy sets theory can provide smooth transition and increase the flexibility of systems. In this research, fuzzy association rules mining is used for counterfeit coin detection and it is considered as the key component of the proposed method because of its affinity with the human knowledge representation.

B. Definitions

In this section, the preliminary concepts are introduced. Let \( I = \{ i_1, \ldots, i_n \} \) be an itemset, each of which is an object in an image and \( T \) a fuzzy transaction set, in which each fuzzy transaction is a fuzzy subset of \( I \). Given the transaction \( t \in T \), we will use \( t(i) \) to denote the membership degree of item \( i \) in the transaction \( t \). Given an itemset \( I \) and a transaction set \( T \), where each transaction is a subset of \( I \), a fuzzy association rule is said to be an “implication” of the form \( A \Rightarrow C \) denoting the presence of itemsets \( A \) and \( C \) in some of the \( T \) transactions, assuming that \( A \subseteq I \), \( A \cap C \neq \emptyset \), and \( A \neq \emptyset \). The measures for establishing a fuzzy association rule’s fitness are the Support and the Confidence. So, a fuzzy association rule can be represented as \( A \Rightarrow C \) \((s,c)\), where \( s \) is called the Support and \( c \) is called the Confidence of rule:

\[
\text{Support} (A \Rightarrow C) = \frac{N_{A \cap C}}{N_{\text{Database}}} \quad (1)
\]

\[
\text{Confidence} (A \Rightarrow C) = \frac{N_{A \cap C}}{N_A} \quad (2)
\]

where \( N_{A \cap C} \) represents the number of transactions which contain \( A \) and \( C \); \( N_A \) represents the number of transactions, which contain \( A \); and \( N_{\text{Database}} \) represents the number of transactions in the database [10]. The problem of mining fuzzy association rules is to extract all fuzzy rules that have Support and Confidence greater than some user specified minimum Support and minimum Confidence thresholds, respectively.

A fuzzy association rule in image database is a rule that associates visual object features and the relationship among objects in images. In the proposed framework, we extract fuzzy association rules, which demonstrate frequent pattern that occurs together in similar types of coin images.

III. PROPOSED FRAMEWORK

The proposed framework for counterfeit coin detection has two modules. It also works in two phases for training and test. In both phases, preprocessing and mining methods have been applied. Figure 1 shows a schematic view of the proposed image miner system. In the first module, the digital coin images are segmented for finding region of interest (ROI). In the second module, the frequent patterns in the images using fuzzy association rules mining are extracted. The fuzzy association rules can discover relationships among patterns in image datasets. So, each input digital coin image is associated with a keyword i.e. fake or genuine. The details of proposed framework are described in the following subsections.

A. Input

The coins used in this research were provided by the Law Enforcement Office. We would note that the access to more fake coins was restricted. Statistical details of four types of Danish 20 Kroner based on our previous research [14] used here are summarized in Table 1. In Figure 2, samples of genuine and fake coins of years, 1990, 1991, 1996, and 2008 are indicated.

<table>
<thead>
<tr>
<th>Year of coin (Danish 20 Kroner)</th>
<th>Number of Genuine coins</th>
<th>Number of Fake coins</th>
<th>Training set</th>
<th>Test set</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>23</td>
<td>113</td>
<td>15</td>
<td>75</td>
</tr>
<tr>
<td>1996</td>
<td>100</td>
<td>10</td>
<td>75</td>
<td>6</td>
</tr>
<tr>
<td>1991</td>
<td>100</td>
<td>14</td>
<td>75</td>
<td>8</td>
</tr>
<tr>
<td>1990</td>
<td>99</td>
<td>25</td>
<td>75</td>
<td>15</td>
</tr>
<tr>
<td>All coins</td>
<td>322</td>
<td>162</td>
<td>225</td>
<td>104</td>
</tr>
</tbody>
</table>
B. Preprocessing

To extract fuzzy association rules, our coin images should be transformed into a suitable format. It is noted that preprocessing is necessary to modify the quality of coin images and develop the feature extraction method more reliably. In the preprocessing module, the images are segmented and regions of interest (ROI) are found. In this study, a progressive water immersion algorithm is incorporated into the proposed method to determine ROIs. Water immersion technique has been remarked as a noticeable method for ROI detection. This algorithm takes an image as the input arguments and returns objects. The details of algorithm are shown as follows [1]:

1) Fix a N\times N sliding window and locate the first point of the window
2) The pixel gray value in the window is compared to find the optimal point
3) If the optimal point is the center of the window then this point is stored as seed
4) Otherwise move the slide window and make the optimal point as the center of the window and increment the count value
5) For each seed, 8-directional neighbor pixel of the seed is compared. If the absolute value neighbor pixel contrast is less than the seed threshold, then this pixel is pushed and marked into queue and then the count value is incremented.
6) Sort the pixel in queue according to the gray value and then pop the last pixel in the queue. Count value is decremented after pop operation.
7) Repeat the steps 5 and 6 until queue is empty.

In the preprocessing step, the median filtering to remove digitization noise has been used. Furthermore, thresholding operation and contrast enhancement have been applied for segmentation of coin images. Through experiments, a global threshold has been set, which is selected for transforming gray scale image into binary [0,1] format. Figure 3 shows results of segmentation step.

C. Image Miner

The image miner module is the most important component of the proposed system. The aim of this module is to discover implicit information from the coin images. At this point, the images are considered as a transaction of objects. A fuzzy association rule in coin images is a rule that associates object features and the relationship among objects in images.

In this study, fuzzy grids-based rule mining algorithm (FGBRMA) [10] is used to mine fuzzy association rules. In this algorithm, each object feature is viewed as a linguistic variable, and the variables are divided into various linguistic terms. To define fuzzy membership functions, three linguistic values are determined for every feature and triangular membership functions are used for each linguistic value. The fuzzy partition on image domain and membership function definition can be seen in Figure 4. At this point, first, the values of each feature are partitioned into three clusters. Then for each generated cluster, the membership degrees of samples are exploited to fit a triangle. As it can be seen in Figure 4, the membership degrees are plotted by “diamond”, “circle”, and “star” symbols, respectively. By using the corresponding membership functions, defined with each linguistic value, the original dataset is changed into a fuzzy dataset.
As mentioned earlier, the FGBRMA [10] is used to mine fuzzy association rules. The problem of fuzzy rule generation is to extract all fuzzy rules that have Support and Confidence greater than a threshold parameter established by the user. In this algorithm, the frequent item sets are found by computing the Fuzzy Support counts of candidate item sets. To check whether each candidate item set is large or not, its Fuzzy Support is computed. When its Fuzzy Support is larger than or equal to the pre-determined Minimum Fuzzy Support (called Min_Fs), it can be said that it is a frequent itemset. After finding all frequent item sets, fuzzy association rules can be generated. To check whether each fuzzy rule \( R \) is effective or not, its Fuzzy Confidence is computed. When its Fuzzy Confidence is larger than or equal to the pre-determined Minimum Fuzzy Confidence (called Min_FC), the fuzzy rule is considered as an acceptable rule [10].

\[ \text{Membership Value} = \text{confidence} \times \text{support} \]

\[ \text{Feature Value} \]

**Fig. 4. Fuzzy Partitions for Feature Values**

**IV. EXPERIMENTAL SETUP AND RESULTS**

The experiments of this study were conducted in the environment of Microsoft Windows 8.1-64 bit and hardware of the test environment consisted of an i7-4500U 4.2 GHz CPU (only one core was used), DDR3 6 GB RAM. The proposed method was coded in MATLAB 2015 simulation environment.

A. **Performance evaluation metrics**

In this subsection, the performance of the proposed framework in terms of the most common metrics has been evaluated. In this research, we used a total of 322 genuine and 162 fake coins for training and evaluating the system respectively. Several experiments were conducted to evaluate the combination of ROIs method and fuzzy association rules mining aiming to counterfeit coin detection. In this research, the performance of the FAR-based classifier, as a tool for the detection of fake coin, has been evaluated in terms of sensitivity, specificity and accuracy. These measures are calculated as follows:

\[ \text{Sensitivity} = \frac{TP}{TP+FN} \]  
\[ \text{Specificity} = \frac{TN}{TN+FP} \]  
\[ \text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \]

where, TP and TN are the number of true positives and true negatives, respectively. Also, FP and FN are the number of false positives and false negatives, respectively.

B. **Analysis of the FAR mining**

As it is mentioned in the previous sections, the proposed framework consists of two modules. First, the automated segmentation of coin images to obtain region of interest (ROI) has been performed. Next, the transaction database has been generated and fuzzy association rules mining by using FGBRMA algorithm have been extracted. With the application of fuzzy logic, the linguistic features have been extracted, and all of them have been added to the final feature subset. Our transaction database has a record for every input coin image and each record has a keyword i.e. genuine or fake along with unique labels.

It is noted that extracting fuzzy rules at the highest Min_Fs and Min_FC percentage yields the best results in terms of sensitivity, specificity, and accuracy. When the proposed framework is implemented with Min_FS=85%, and Min_FC=90%, the total of 1084 rules have been discovered. From these, there are 132 rules with two elements, 437 rules with three elements, 301 rules with four elements, and 214 rules with five elements. In this way, some of the mined rules are listed in Table 3.

**TABLE 3. Some of fuzzy association rules with fuzzy support (FS)≥85% and fuzzy confidence (FC)≥90%**

<table>
<thead>
<tr>
<th>Fuzzy Rules</th>
<th>FS Value</th>
<th>FC Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>2O_{i,2} \land O_{2,2} \Rightarrow 1O_{i,2}</td>
<td>95%</td>
<td>100%</td>
</tr>
<tr>
<td>1O_{i,2} \land O_{2,2} \Rightarrow 2O_{i,2}</td>
<td>90%</td>
<td>95%</td>
</tr>
<tr>
<td>1O_{i,2} \land O_{2,2} \Rightarrow 2O_{i,2} \land 1O_{2,2}</td>
<td>95%</td>
<td>100%</td>
</tr>
<tr>
<td>2O_{i,2} \land O_{2,2} \land 1O_{2,2} \Rightarrow 1O_{i,2} \land 2O_{i,2} \land 2O_{2,2}</td>
<td>85%</td>
<td>95%</td>
</tr>
<tr>
<td>1O_{i,2} \land 2O_{i,2} \land 2O_{2,2} \Rightarrow 1O_{i,2} \land O_{i,2} \land 2O_{2,2}</td>
<td>95%</td>
<td>100%</td>
</tr>
</tbody>
</table>

As it can be seen in Table 3, \( O_{i,j} \) is an object in transaction dataset and integers show quantifying the occurrence of the object.
TABLE 4. Performance of the proposed method as compared with some other counterfeit detectors in terms of accuracy

<table>
<thead>
<tr>
<th>Model</th>
<th>Type of Coins</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optical Mouse Sensor [15]</td>
<td>Two-Euro</td>
<td>97 %</td>
</tr>
<tr>
<td>Image Content by Fuzzy Association Rules Mining (proposed)</td>
<td>20 Kroner 1990, 1991, 1996, and 2008</td>
<td>98 %</td>
</tr>
</tbody>
</table>

Figure 5 illustrates the performance of the proposed framework in terms of accuracy, sensitivity and specificity (98.06%, 96.81% and 99.02%, respectively). It is interesting to note that, as expected, capabilities of the proposed framework reveal the effectiveness of data mining techniques.

The performance of proposed method has been compared with some other methods in Table 4, as well. As shown in Table 4, the proposed system has higher accuracy, as compared with system reported in [15]. However, the accuracy of the proposed system is slightly lower than the system reported in [14]. Although the accuracy resulted in the method of reference [14] is 99%, the accuracy of the proposed system is 98%, which is an acceptable level when compared with some research in this field. Furthermore, the accuracy of the proposed framework depends steadily on the precision of the parameters such as Min_FS and Min_FC in this research. Therefore, determining optimum values of these parameters, can improve the performance of the proposed framework, effectively.

V. CONCLUSION AND FUTURE WORK

In this study, a counterfeit coin detection method based on fuzzy association rules was proposed. Fuzzy association rules mining can adequately utilize a large amount of data, and extract important relationships among items. In the proposed framework, an image mining method using ROI technique has been introduced for finding important information from the coin image data set. It is interesting to note that by using fuzzy association rules mining technique, which have been equipped by ROI method, the accuracy of the proposed framework is 98.06%, which is an acceptable level when compared with recent research in this field. In this study, the problem with a general form has been described to provide a common framework for other problems appeared in other domains.

Direction for future work can be stated as follows: (a) using optimization algorithms to tune the fuzzy membership function, (b) using genetic algorithm and particle swarm optimization (PSO) algorithm to determine optimum values of parameters (Min_FS and Min_FC).

ACKNOWLEDGMENT

The authors are grateful to the anonymous referees for their helpful comments and suggestions. This research was supported by Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

Objective Identification of Bullets Based on 3D Pattern Matching and Line Counting Scores

Danny Roberge, Ph.D.  
Research Department  
Ultra Electronics Forensic Technology  
Montréal, Canada  
danny.roberge@ultra-ft.com

Alain Beauchamp, Ph.D.  
Research Department  
Ultra Electronics Forensic Technology  
Montréal, Canada  
alain.beauchamp@ultra-ft.com

Serge Lévesque, Ph.D.  
Research Department  
Ultra Electronics Forensic Technology  
Montréal, Canada  
serge.levesque@ultra-ft.com

Abstract—The goals of this study were to develop an objective identification method for bullets fired from conventionally rifled barrels, and to test this method on public and proprietary bullet 3D image datasets. Two newly developed bullet identification scores, the Line Counting Score and the Pattern Matching Score, computed on 3D topographic images of a realistic dataset, enabled us to define a discriminative line at a false match rate of 1/10000 on a two-dimensional plot that shows both identification scores for matches and nonmatches. A likelihood function was also computed from a linear combination of both scores.

Keywords—forensic science, firearm identification, bullet, topography measurements, consecutive matching striae (CMS), false match rate, likelihood ratio

I. INTRODUCTION

A. Literature Review

With the development of high-resolution 3D sensor technologies, the topography of toolmarks can be measured at a vertical resolution of less than one micron; this advancement has made research into 3D forensic firearm identification possible. Projects in this field focus on the development of algorithms that extract data relevant to firearm identification from raw topographic measurements. Significant progress has been made for bullets [1–3] and cartridge cases [4–9].

Research on bullet toolmarks is hampered by the challenge of capturing the topography of the bullets due to their approximate cylindrical and sometimes arbitrary shape. Results generated by bullet identification algorithms [1–3] typically follow three main research avenues: profile similarity measures, consecutive matching striae (CMS) method criteria, and machine learning (ML) strategies.

The CMS method is a line counting method that registers and counts groups of CMS when comparing two striated toolmarks. Biasotti and Murdock [10] stated their quantitative CMS method criteria for the objective confirmation of the common source for three-dimensional toolmarks (which include striae on fired bullets). There is an identification “when at least two different groups of at least three consecutive matching striae appear in the same relative position, or one group of six consecutive matching striae are in agreement in an evidence toolmark compared to a test toolmark”.

Chu et al. [3] implemented a CMS algorithm that used high-resolution topography measurements taken by a confocal microscope equipped with an objective lens with 20X magnification and a pixel resolution of 1.5625 μm. The input data were 3D topographic images from one of the Hamby-Brundage sets, consisting of 35 bullets fired from 10 consecutively manufactured rifled Ruger P85 pistol barrels, grouped as 10 pairs of known training bullets and 15 unknowns for comparison [11]. With this algorithm, the 10 known bullet match pairs were objectively identified.

Hare et al. [12] used random forests, a specific ML algorithm, to successfully classify a Hamby-Brundage set of bullets from the public NIST Ballistics Toolmark Research Database [13]. The authors found that the two most important features for classification of the matches and nonmatches are the cross correlation function similarity measure (CCF) and the total number of matching lines in a land impression pair comparison.

B. Objective Identification

Objective identification methods provide meaningful scores associated with a probability measure that can reinforce the legal admissibility of expert conclusions. The most relevant probability measure for objective identification is the likelihood ratio [14]. In the context of forensic firearm identification, the likelihood ratio (LR) is the probability of being a match divided by the probability of being a nonmatch, based on a set of meaningful scores obtained from the comparison of a pair of toolmarks.

When comparing two particular bullets (herein called the reference bullet and the test bullet), the strict application of the LR definition imposes severe constraints on the population used to build the match and nonmatch score distributions. In particular, the match score distribution must be the result of comparisons between the reference bullet and a sufficiently large set of test bullets fired from the same gun as the reference bullet. For a study that involves only small groups of known matches (pairs, triplets), the match score distribution of each firearm cannot be used to draw statistically valid
conclusions. The remaining option is to study the “typical” behavior of the LR by combining all the reference bullet match scores into a single average match score distribution (and doing so for the nonmatch scores).

A modest and conservative alternative is to adopt a false match rate (FMR) and define a boundary that discriminates the matching and nonmatching regions in the score space in a way that is consistent with this FMR, a methodology based solely on the nonmatch score distribution. In the context of firearm identification, the FMR is the probability that two bullets (or cartridge cases) will be erroneously identified as having been fired from the same firearm.

II. METHODS

A. Strategy

The goals of this study were (i) to develop an objective identification method for bullets fired from conventionally rifled barrels (i.e., excluding polygonal rifling and unrifled barrels), that yields results that can be displayed using a visualization tool and, (ii) to test this method on public and proprietary bullet 3D image datasets.

We adopted the strict approach of defining scores in a two-dimensional space. With pattern matching and the CMS method being two different, but successful, viewpoints, it seemed reasonable to adopt a pattern matching score and a line counting score. While imposing a two-dimensional score vector is a severe constraint, it can be relaxed by allowing each score to be a function of several complementary measures.

B. Data

The optimal range in the lateral and depth resolution of the 3D images used for automated objective identification is yet to be determined but, intuitively, a higher resolution carries more information and should be preferred. This is critical for a fully automated system, which requires advanced surface tracking methods coupled with controlled rotation and translation motors images in order to build the full 360° band of a pristine bullet or the arbitrary shape of a fragment.

In practice, higher lateral resolution is obtained at the cost of a reduced field of view, which translates into a need to capture a larger number of images, thus increasing acquisition time. The resulting band is also narrower, which limits the area available for firearm identification. Furthermore, a smaller field of view negatively affects the automated image acquisition and bullet-tracking processes, since the base of some pristine bullets is far from being flat, and deformed bullets and fragments can be much more difficult to manage.

The data used in this study consisted of a dataset of pristine bullets captured using BULLETTRAX® acquisition station of the IBIS®-TRAX-HD3D® system. Its 3D confocal microscope is equipped with a 10X objective at 3.125 μm/pixel resolution (twice the value of [3, 12]). Bullet images are 360° bands built by stitching together several images. The 9 mm bullets of the dataset (six grooves, right twist) were fired from 136 firearms (from pairs to 5-tuples, for a total of 406 bullets) from different firearm manufacturers that are representative of the population of hand guns in an urban U.S. environment. The topographic measurements of a pristine bullet and of a bullet fragment (not used in this study) are shown in Fig. 1.

Before any computations were performed, every known match pair was inspected using a 3D virtual microscope, and was labeled as visually matching or not. In this study, we define the subjective visual matching criterion as a minimum of two distinct land impression pair comparisons in a common phase1 where sufficient agreement is observed. The fraction of confirmed visually matching pairs, called the Visual Matching Ratio (VMR) in this paper, is 55% for this dataset. The purpose of this labeling was to focus on reasonably good matches when training the algorithms.

C. Profile Extraction and Comparison

The process that we used to extract a profile from our bullet 3D images is similar in essence to some of those used in previously published studies [3, 12], the difference being that the process starts with the topography a 360° band for bullets captured using BULLETTRAX. The bullet’s local curvature is removed from the band, yielding a roughness image. A straightening algorithm determines the dominant orientation of the toolmarks and applies a geometric transformation to align them perpendicularly to the direction of the band (Fig. 2). Another algorithm automatically detects the shoulders and splits the roughness image into individual land and groove images.2

In most cases, the majority of toolmarks are concentrated very near the base of the bullet. The relevant areas for objective

1 The concept of phase will be defined in section 3.1.A.

2 These are Forensic Technology’s proprietary algorithms
identification are automatically emphasized using a binary mask based on a measure of local coherence. A profile representation of the topography within the mask is computed. At this stage, the comparison process is reduced to computing a similarity measure between such profiles.

Two similarity measures were evaluated. One is the global maximum of the cross-correlation function (CCF), which has proven its usefulness for objective identification of bullet profiles [1] and cartridge case areas [5, 6, 9]:

$$\text{CCF}_{\text{MAX}} = \max \left\{ \sum_{i} \frac{X_{i} - \mu_{X}}{\sigma_{X}} \left( \frac{Y_{i} - \mu_{Y}}{\sigma_{Y}} \right) \right\} ,$$

(1)

where $N$ is the number of elements in the X and Y profiles, $\mu_{X}$ is the average, and $\sigma_{X}$ is the standard deviation of the X profile (and similarly for the Y profile). The final CCF score is the highest score among the set of CCF values computed over several horizontal translations ($\Delta$) of one vector with respect to the other.

The CCF is invariant under a global change of the (vertical) scale. Other similarity measures are not scale invariant, and might then complement the CCF. We selected the Absolute Normalized Difference (AND):

$$\text{AND} = 1 - \frac{\sum_{i} \left| X_{i} - \mu_{X} \right| - \left( Y_{i} - \mu_{Y} \right) |}{\sum_{i} \left| X_{i} - \mu_{X} \right| + \left( Y_{i} - \mu_{Y} \right) |} ,$$

(2)

where $\Delta$ is determined from the CCF\text{MAX} measure.

**D. Line Extraction and Comparison**

In this study, we were interested in finding a single score based on line counting, that would then be computed independently for peaks and valleys, and combined. The definition of the peak/valley of a profile must be part of the training process of the pattern matching algorithm. Similar to Chu et al. [3], we defined four parameters for peaks: horizontal location of the maximum, width, left height, and right height. The peak detection criteria based on these values were fine-tuned from the data in order to optimize objective identification performance. The valleys are characterized by the same set of parameters, computed from the reversed profile.

The first step in the line counting process (for peaks) is to align the reference profile and the test profile according to the relative position found by the pattern matching algorithm (CCF). Each profile is then converted into an idealized binary profile. Binary AND and OR operations are applied to the pair of profiles, and a binary vector $P$ (for position) is created, where values of 1 indicate matching lines and values of 0 indicate nonmatching lines which are present in either one of the input profiles but not in both. Here is an example of such vector:

$$P = (0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1).$$

(3)

Line counting analysis is easily performed from this vector. A CMS group corresponds to a set of consecutive 1s that is interrupted by a 0 on its right side or at the end of the vector. The whole comparison is then summarized by the number of groups of different lengths, such as, in the above example: 1 single line, one group of four CMS and one group of six. Some other parameters, computed from the P-vector, will be used in the next section. These are the total number of matching striae, $T$ (= 11 from the example above); the number of consecutive pairs, $T_{2}$ (8); the number of CMS groups $N_{G}$ (3).

**III. RESULTS AND DISCUSSION**

**A. Bullet Comparison Score Strategy**

Each land of a firearm barrel is like a unique tool, independent of the other lands, leaving distinctive toolmarks, called land engraved areas (LEAs), on the fired bullet’s surface. The comparison of two pristine bullets showing N LEAs (i.e., not fragments) leads to the possibility of $N^{2}$ LEA-to-LEA comparisons. However, since the sequence of the LEAs is fixed inside the barrel, these $N^{2}$ possibilities can be arranged into $N$ groups, called phases, of $N$ LEA-to-LEA comparisons with consistent ordering.

Finding the correct phase, by considering either the phase with the best overall LEA-to-LEA comparison or the phase which shows the best average agreement, is usually the first step in bullet identification. This step relies on pattern matching techniques; the CMS method is being used only as a quantitative measure of the pattern agreement after the two compared LEAs have been properly aligned. In this study, the best average CCF value was used to find the correct phase, and the profile alignment (i.e., the $\Delta$ value) that yielded the CCF\text{MAX} score for each LEA-to-LEA comparison was used for all line counting type scores.
Once the phase was found by the algorithms, we were then faced with the challenge of selecting the optimal similarity measure for the bullet pair “as a whole”. Our choice must consider two elements: (i) firearm examiners, in practice, rarely draw final conclusions based only on one LEA-LEA comparison (they look for similarities over the whole circumference of the bullet), and (ii) bullet fragments from crime scenes can have missing LEAs. Consequently, developing a similarity measure based on the comparison of all possible LEA-LEA comparisons at the best phase from datasets of pristine bullets can be of no practical value for the objective identification of crime evidence.

As will be discussed below, the average of the best two LEA-LEA comparisons offers a good compromise. Therefore, for the new types of scores being presented in this paper (i.e., the newly developed Pattern Matching Score (PMS) and Line Counting Score (LCS)), the score value represents the average of the best two LEA-LEA comparisons within the selected phase. The selection of the LEAs to be part of the score relies on the PMS values. The LCS is computed for those LEA-to-LEA comparisons with optimal alignment obtained from the CCF. This best two-score approach is also consistent with our visual matching criterion described earlier.

B. Pattern matching score (PMS)

Our PMS is defined as the combination of the CCF and the AND that optimizes the separation between the match and nonmatch distributions for our dataset. The representative value of the CCF and the AND scores is computed as the respective average value over the LEAs that yield the two highest CCF scores at the best phase. We found that the score distribution of the visually confirmed matches lies along a straight line crossing approximately the origin (correlation coefficient = 0.96) in the CCF-AND plane while the nonmatch is an approximate Gaussian distribution with significant variance along the direction perpendicular to its main axis. It is then natural to define the PMS as a linear combination of the CCF and the AND, with their respective weight computed from the eigenvector of the main principal component of the match distribution.

C. Line Counting Score (LCS)

The simplest measure is the total number of matching striiae, that is, the sum of the binary elements in the \( P \) vector, as suggested by [12]. We found that the two-dimensional (peak and valley) visually confirmed match and nonmatch \( T \) distributions strongly overlap, which makes \( T \) alone unsuited for objective identification. However, dividing \( T \) by \( N_p \), the number of elements in the \( P \) vector, significantly reduces the overlap. This binary similarity measure is the Jaccard coefficient, which always lies between 0 and 1.

Since the total number of striiae does not consider the potentially positive contribution from consecutiveness, we tested different linear combinations of \( T \) and a new term, \( T_2 \) (defined in the previous section). The optimal combination was found to be their sum normalized by \( N_p \).

Since any monotone increasing function of a score cannot change performance, we adopted the following LCS for peaks and valleys,

\[
LCS = \sqrt{\frac{1}{2} \left( \frac{T + T_2}{N_p} \right)} = \sqrt{\frac{T}{N_p} \left( 1 - \frac{N_G}{2T} \right)} \tag{4}
\]

where the \( \frac{1}{2} \) factor yields an upper bound near unity and the square root ensures that the threshold segregating matches and nonmatches is about 0.5; it can be shown that both mathematical expressions in (4) are equivalent. The rightmost one can be interpreted as the (square root) product of two contributions: the normalized number of matching striiae and a correction term which penalizes a high number of CMS groups, \( N_G \), for a given \( T \).

Following the same reasoning as for the PMS, the final LCS is defined as a weighted sum of contributions from peaks and valleys, with equal weights in this case.

D. PMS and LCS: False Match Rate

The PMS and the LCS were computed for every pair of bullets in our dataset. Fig. 3 shows the scores for all known matching and nonmatching bullet pairs, that is, 81793 nonmatch scores and 422 match scores, among which 235 are for the visually confirmed match pairs. The PMS and the LCS are highly correlated for the visually confirmed known matching pairs, and follow a straight line that approximately crosses the origin. The line that best fits the match scores distribution is found with PCA analysis.

An FMR function is obtained by performing an orthogonal projection of the nonmatch two-dimensional scores on the match scores’ best-fitting line, in order to obtain a distribution of the projected distance \( D \) between the projected points and a fixed point on the line. In this process, we are in fact defining a single score \( D \) that linearly combines the PMS and the LCS. Since this score cannot be interpreted easily by firearm experts, we propose to keep the two-dimensional representation for ease of interpretation, but use \( D \) for the computation of error rates. The \( D \) score match and nonmatch probability densities are shown in Fig. 4.

The FMR associated with a given projected distance \( D \) is the area of the nonmatch distribution for distances larger than \( D \). A linear decision boundary corresponding to an FMR of 1/10000 is shown in Fig. 6 (discussed in the next section). The corresponding sensitivity (i.e., the proportion of the visually confirmed known matches that satisfy the criterion) is 97.7%.

As a reminder, these results were obtained by averaging the LCSs and PMSs over the best two LEA-to-LEA comparisons. This analysis was repeated by considering only the single best
LEA-to-LEA comparison. Very similar results were obtained, with a slightly lower sensitivity of 92.6%, thus supporting the choice of the two-score averaging process.

E. Likelihood Ratio Function and Extrapolation of the Nonmatch Distribution

An LR function characterizing our dataset can be computed as the ratio of the match and nonmatch probability densities of the projected distance score D. For the D values of interest, the nonmatch distribution must be extrapolated beyond the highest available score values. Visual inspection of this distribution highly suggests that the nonmatch distribution has a Gaussian shape (Fig. 4); a reasonable approximation then seems to fit a normal distribution over the whole set of nonmatch scores and use the resulting analytical function for extrapolating over high scores.

However, the right wing of the nonmatch distribution and the Gaussian that best fits the whole distribution diverges for the 1% highest scores (800 scores); the logarithm of the complementary cumulative distribution (or, equivalently, of FMR(D), the false match rate function of D) becomes linear while the Gaussian function has an approximate quadratic asymptotic form (Fig. 5).

An alternative to global fitting is to perform statistical modeling of the highest scores of the experimental distribution, which is the object of Extreme Value Theory. For univariate distributions, three families of extreme value distributions are commonly used [15]: the Gumbel, the Fréchet, and the Weibull distributions. The Weibull distribution has been found to fit the highest scores produced by multiple recognition algorithms in computer vision [16]. The observed linear behavior of \( \log_{10}(\text{FMR}(D)) \) is consistent with a Weibull distribution with a shape parameter of 1.

The LR function is computed by approximating the nonmatch distribution by two analytical functions: the Gaussian fit for D values smaller than 0.55 and the PDF associated with the linear extrapolation previously discussed, for larger D values. The Parzen-window technique is used to smooth the match distribution and takes the whole set of matches (including those that cannot be visually confirmed) into account. The LR value corresponding to FMR=1/10000 is \( 10^{1.95} \approx 90 \). Using only the Gaussian fit would overestimate the LR of the visual matches by several orders of magnitude.

The linear boundary with FMR = 1/10000 in the PMS-LCS plane and the LR(D) function could be a valuable quantitative criterion for case work in forensic laboratories or proficiency tests. Fig. 6 summarizes the output of the method on the PMS-LCS plane for one of the 15 bullets considered as “unknown” in the Hamby-Brundage test (similar results are obtained from every bullet of the set). This bullet is compared with 10 known matching pairs. The twenty two-dimensional score vectors are shown as black dots, among which two are above the discriminative line; this pair and the tested bullet are connected to the same barrel [17]. All nonmatches are concentrated below the line. The figure also displays the empirical distribution of nonmatches and visually confirmed matches from our dataset, to be used as a visual reference. Isolines of \( \log_{10}(\text{likelihood ratio}) \) computed from our dataset provide a quantitative assessment of the similarity between the matches.
The result of a bullet pair comparison can be displayed using a visualization tool that shows the isolines of likelihood ratios, with a discriminative line at FMR = 1/10000, and with representative distributions of match and nonmatch score values.

IV. CONCLUSION

Using line counting and pattern matching methods, two new objective identification scores have been developed based on high-resolution 3D topography for bullets fired from conventionally rifled barrels. The Pattern Matching Score is a linear combination of the cross-correlation function and the Absolute Normalized Difference similarity measure; the Line Counting Score is the arithmetic average of a quantity calculated separately for peaks and valleys, and is based on the normalized number of matching striations and a correction term of order unity sensitive to consecutiveness.

Experiments done on a realistic set allowed us to define a discriminative line at the FMR = 1/10000 level in a two-dimensional plot that shows both objective identification scores for matches and nonmatches. The proportion of the visually confirmed known matches that satisfy the criterion is 97.7% (sensitivity).

A linear combination of the PMS and the LCS yields a more abstract univariate score. Its highest 1% of scores fit a Weibull distribution with a shape parameter equal to unity. This Weibull fitting was used to compute the likelihood ratio (LR) for high scores not sampled by the experimental nonmatch distribution. The likelihood function yields values larger than $10^{1.95}$ ($\sim 90$) for most match pairs that were confirmed visually prior to the numerical analysis.

The result of a bullet pair comparison can be displayed using a visualization tool that shows the strength of the pair’s Pattern Matching and Line Counting Scores relative to the isolines of likelihood ratios, with a discriminative line at FMR = 1/10000, and with representative distributions of match and nonmatch score values.

ACKNOWLEDGMENT

We are grateful to the Allegheny County Forensic Laboratory Division, for providing the bullets for the large dataset in this study; to Jim Hamby, who provided the physical bullets from a Hamby-Brundage set; to Michel Paradis and Marlene Reed, for reviewing the manuscript and providing valuable suggestions; and to Debra Desrosiers, who performed all bullet 3D digital image captures.

REFERENCES


Visual Detection of Fake Coins Using Fisher Vectors
Li Liu¹,², Yue Lu² and Ching Y. Suen³
1. School of Information Engineering, Nanchang University, Nanchang, China
2. Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University, Shanghai, China
3. Faculty of Engineering and Computer Science, Concordia University, Montreal, Canada
liuli_033@163.com

Abstract—In this paper, we propose a visual appearance-based approach for fake coin detection. The keypoints are extracted from the coin image that are invariant under image transformations. We then divide each coin into several regions to investigate the roles that they play in detecting fake coins. Afterwards, the keypoints from each region are encoded as a Fisher vector. The coin image is finally represented by concatenating the Fisher vectors of all the regions in a weighted fashion. The proposed approach is evaluated on three datasets and promising results have been achieved. The results show that the closer to the rim of the coin the region is, the more important it is for fake coin detection.

Keywords—fake coin detection, keypoints, region, Fisher vector

I. Introduction
Coins are widely used in everyday life. Compared with banknotes, coins are less susceptible to abrasion and thus can be circulated for a much longer period of time. In addition to being used as currency, many people enjoy collecting coins, since they are often of artistic value, through which one can gain a glimpse of the culture at the time when the coin is struck. However, production of fake coins on a large scale has been reported across all over the world [1-2], which causes great harm to the society. Therefore, it is imperative to detect fake coins, and it is also one of the major concerns of numismatics.

A. Related work
In order to distinguish between genuine and fake coins, the coins’ acoustical, electromagnetic properties and visual appearances have been exploited [3-9]. In this paper, we give a brief review of the fake coin detection methods employing the visual appearances which are more relevant to our study. The common practice for the visual appearance-based approach is to apply pattern recognition techniques to the coin images. In [6], Tresanchez et al. employed template matching to detect two-Euro fake coins. In spite of simplicity, template matching badly suffers from its sensitivity to coin rotation. Wang et al. [7] first aligned the image of coin to be detected with the reference image that is captured from a genuine coin. Afterwards, a back propagation neural network was trained for fake coin detection. They experimented with two types of coins, viz. Japan Meui Year 29 1-Yen silver coin and Taiwan Year 1950 2-Dime aluminum coin and reported promising results, yet on quite a small dataset. There were only 10 genuine and 42 fake coin images for the 1-Yen silver coin. Regarding the 2-Dime aluminum coin, there were 16 genuine and 41 fake images. In [8], Gavrijaseva and Martens used the cross-correlation technique for fake Euro coin detection. However, the details of the dataset they employed such as the number of coin types and the number of genuine and fake coins for each coin type are missing. Sun et al. [9] proposed a fake Danish coin detection approach. Two types of features were
extracted from the coin image, which aimed at characterizing the queen's head in the middle of the coin and the surrounding letters respectively. To be more specific, a feature named angle-distance was presented to describe the queen's head. Five features, i.e. stroke width, roughness, height, width and the relative distance between two adjacent letters, were used to describe the letters. They successfully detected all the fake coins according to their experiments. In our previous work [10], one-class learning was conducted for fake coin detection, considering the fact that the number of fake coins is usually very limited in the real world. So only genuine coins were employed to train the classifier.

In this study, we propose a new approach to detect fake coins using their visual appearances. The keypoints that are invariant under image transformations are first extracted from the coin image. In order to investigate the roles that different coin regions play in detecting fake coins, we divide a coin image into several regions. The keypoints from each region are then encoded as a Fisher vector. Thus a coin image can be represented by concatenating the Fisher vectors of all the regions in a weighted fashion. Afterwards, a linear classifier is trained to distinguish between genuine and fake coins. To validate the proposed approach, we evaluate it on three datasets. The results show that the proposed approach compares favorably with the state-of-the-art fake coin detection methods. Besides, the closer to the coin rim a region is, the more important it is.

II. Proposed approach

In order to segment the coin from the background, the Hough Transform is employed [11]. An example segmentation is shown in Figure 1, where the boundary of the segmented coin is outlined. We then detect keypoints in the coin image. The keypoints are invariant to image transformations such as rotation, scaling, and illumination variations, etc. Each keypoint is subsequently characterized by a \( D \)-dimensional descriptor. Various keypoint detectors have been proposed in the literature such as Difference-of-Gaussian (DOG) [12], Harris-Laplace, Harris-Affine, Hessian-Laplace or Hessian-Affine [13]. Besides, there are many keypoint descriptors, e.g. Scale Invariant Feature Transform (SIFT), Shape context (SC) and gradient location and orientation histogram (GLOH) [14]. We compare the performance of different combinations of keypoint detectors and descriptors in Section III.

Fig. 1. An example coin segmentation.

With a set of descriptors extracted from an image, it is important to encode them as a vector. Some encoding methods have been proposed such as Bag-Of-Visual-Words [15], Fisher vector [16-17], and Vector of Locally Aggregated Descriptors(VLAD) [18]. We employ the Fisher vector encoding approach in this paper, because of its remarkable performance demonstrated in many applications, e.g. image classification [18] and retrieval [19-20].

A. Fisher vector encoding

Let \( X = \{ x_1, x_2, \ldots, x_T \} \) be a set of descriptors, where \( T \) is the number of descriptors and \( x_t (t = 1, 2, \ldots, T) \in \mathbb{R}^D \). Assuming that \( X \) can be modeled by a probabilistic density function \( u_x \) with the parameters \( \lambda \). It can be represented by the gradient vector:
\[ G_\lambda^X = \frac{1}{T} \sum \nabla_\lambda \log u_\lambda (X) \]  

Suppose that the descriptors \( x_i \)'s are generated independently by \( u_\lambda \), thus \( G_\lambda^X = \frac{1}{T} \sum \nabla_\lambda \log u_\lambda (x_i) \). The gradients describe the contribution of the parameters to the generation process. A natural kernel on the gradients is:

\[ K(X, Y) = G_\lambda^X F_\lambda^{-1} G_\lambda^Y \]  

where \( F_\lambda \) is the Fisher information matrix of \( u_\lambda \) and is defined as follows:

\[ F_\lambda = \mathbb{E}_{\lambda-\mu} [\nabla_\lambda \log u_\lambda (X) \nabla_\lambda \log u_\lambda (X)'] \]  

Since \( F_\lambda \) is positive definite and symmetric, it has a Cholesky decomposition \( F_\lambda = L_\lambda L_\lambda' \). So the kernel \( K(X, Y) \) defined in Eq. (2) can be rewritten as the dot product between the normalized gradient vectors:

\[ K(X, Y) = g_\lambda^X g_\lambda^Y \]  

where \( g_\lambda^X \) is called the Fisher vector of \( X \) and is formally defined as:

\[ g_\lambda^X = L_\lambda G_\lambda^X \]  

Following [16-17], the probabilistic density function \( u_\lambda \) is chosen to be the Gaussian mixture model:

\[ u_\lambda(x) = \sum_{i=1}^{K} w_i u_i(x) \]  

where \( K \) is number of components, and the mixture weight \( w_i \geq 0 \) with \( \sum_{i=1}^{K} w_i = 1 \). \( u_i (i = 1, 2, \ldots, K) \) is the \( i \)th Gaussian as defined in the following:

\[ u_i(x) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right\} \]

where \( \mu_i \) is the mean vector and \( \Sigma_i \) is the covariance matrix. Therefore, the parameters of \( u_\lambda \) are denoted as \( \lambda = \{ w_i, \mu_i, \Sigma_i, i = 1, 2, \ldots, K \} \). The covariance matrix \( \Sigma_i \) is assumed to be diagonal, so it can be denoted by the variance vector \( \sigma_i^2 \).

Let \( y_i(i) \) be the soft assignment of descriptor \( x_i \) to the \( i \)th Gaussian and it is defined as:

\[ y_i(i) = \frac{w_i u_i(x_i)}{\sum_{j=1}^{K} w_j u_j(x_i)} \]

As in [17], we assume the Fisher information matrix \( F_\lambda \) to be diagonal. Consequently, the gradient with respect to the mean \( \mu_i \) and the standard deviation \( \sigma_i \) is calculated as:

\[ \frac{\partial g_\lambda^X}{\partial \mu_i} = \frac{1}{T \sqrt{w_i}} \sum_{t=1}^{T} y_t(i)(\frac{x_t - \mu_i}{\sigma_i}) \]

\[ \frac{\partial g_\lambda^X}{\partial \sigma_i} = \frac{1}{T \sqrt{2w_i}} \sum_{t=1}^{T} y_t(i)\left[ \frac{(x_t - \mu_i)^2}{\sigma_i^2} - 1 \right] \]

The Fisher vector \( g_\lambda^X \) is formed by concatenating \( g_{\mu,i}^X \) and \( g_{\sigma,i}^X \) for \( i = 1, 2, \ldots, K \), whose dimensionality is therefore \( 2KD \). \( K \) is empirically set equal to 256 in this paper. We then apply signed square-rooting and L2 Normalization to the Fisher vector, which have been reported to greatly improve the performance [17]. It should be noted that the gradient with respect to the weight \( w_i \) is not considered in our work, since it brings little performance improvement.

### B. Coin image representation

To investigate the roles that different parts of coin play in detecting fake coins, we divide a coin image into \( M \) regions, viz. \( R_1, R_2, \ldots, R_M \). An example is shown in Figure 2, where \( M = 5 \). The keypoints from each region are encoded as a Fisher vector and then normalized as detailed in Section II-A. In order to generate the representation of the coin image, the Fisher vectors from all the regions are concatenated. However, instead of treating all the Fisher vectors equally, a weight \( \omega_i (i = 1, 2, \ldots, M) \) is assigned to each Fisher vector. More formally, \( \omega_i \geq 0 \) and \( \sum_{i=1}^{M} \omega_i = 1 \).

Since the Fisher vectors work well with linear classifiers as proved in [17], we train a linear SVM [22] to detect fake coins.
III. Experiments

To validate the effectiveness of the proposed approach, we test it on three datasets. The performance of the proposed approach under various configurations is reported. We also compare the proposed approach with several state-of-the-art fake coin detection methods.

A. Datasets

We build three coin image datasets, all of which are captured from the obverse side of Danish 20 Kroner, but from different years, viz. 1991, 1996 and 2008. The number of genuine and fake coins in each dataset is given in Table 1. We further split each dataset into three sets: training set, validation set and test set, the details of which are given in Table 2. The validation set is used for the selection of parameters involved in the proposed approach. The best ones are then applied to the test set. It should be noted that we make ten random splits and the average accuracy is reported.

B. Configurations

The performance of the proposed approach under various configurations is compared on the validation set.

- Different combinations of keypoint detectors and descriptors

We test the proposed approach when different combinations of keypoint detectors and descriptors are employed. The results are given in Table 3, where we can see that the combination of DOG detector and SIFT performs the best. As in [17], we reduce the dimension of SIFT descriptor from 128 to 64 using PCA, which greatly improves the performance as shown in Table 4.

<table>
<thead>
<tr>
<th>Detector</th>
<th>Descriptor</th>
<th>Danish1991</th>
<th>Danish1996</th>
<th>Danish2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris-Laplace</td>
<td>SIFT</td>
<td>0.903</td>
<td>0.910</td>
<td>0.892</td>
</tr>
<tr>
<td>Harris-Laplace</td>
<td>GLOH</td>
<td>0.882</td>
<td>0.891</td>
<td>0.867</td>
</tr>
<tr>
<td>Harris-Laplace</td>
<td>SC</td>
<td>0.852</td>
<td>0.861</td>
<td>0.825</td>
</tr>
<tr>
<td>Harris-Affine</td>
<td>SIFT</td>
<td>0.921</td>
<td>0.903</td>
<td>0.898</td>
</tr>
<tr>
<td>Harris-Affine</td>
<td>GLOH</td>
<td>0.911</td>
<td>0.892</td>
<td>0.883</td>
</tr>
</tbody>
</table>

| Table 1: Statistics of Coins in the Three Datasets. |
|----------------|----------------|-------|
| Datasets       | No. of genuine coins | No. of fake coins | Total  |
| Danish1991     | 2,000            | 2,000           | 4,000  |
| Danish1996     | 2,000            | 2,000           | 4,000  |
| Danish2008     | 2,000            | 2,000           | 4,000  |

| Table 2: Statistics of Training, Validation and Test Sets. |
|----------------|----------------|----------------|----------------|
| Datasets       | Training set   | Validation set | Test set       |
|                | No. of genuine coins | No. of fake coins | No. of genuine coins | No. of fake coins |
| Danish1991     | 1,000           | 1,000           | 500             | 500             |
| Danish1996     | 1,100           | 1,100           | 500             | 500             |
| Danish2008     | 1,000           | 1,000           | 500             | 500             |

<p>| Table 3: Accuracy of the Proposed Approach W.R.T. Different Combinations of Detectors and Descriptors. |
|----------------|----------------|-------|-------|-------|
| Detector       | Descriptor     | Danish1991 | Danish1996 | Danish2008 |
| Harris-Laplace | SIFT           | 0.903     | 0.910     | 0.892   |
| Harris-Laplace | GLOH           | 0.882     | 0.891     | 0.867   |
| Harris-Laplace | SC             | 0.852     | 0.861     | 0.825   |
| Harris-Affine  | SIFT           | 0.921     | 0.903     | 0.898   |
| Harris-Affine  | GLOH           | 0.911     | 0.892     | 0.883   |</p>
<table>
<thead>
<tr>
<th>Method</th>
<th>Without PCA</th>
<th>With PCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris-Affine</td>
<td>0.895</td>
<td>0.887</td>
</tr>
<tr>
<td>Hessian-Laplace</td>
<td>0.912</td>
<td>0.910</td>
</tr>
<tr>
<td>Hessian-Laplace</td>
<td>0.923</td>
<td>0.912</td>
</tr>
<tr>
<td>Hessian-Laplace</td>
<td>0.901</td>
<td>0.891</td>
</tr>
<tr>
<td>Hessian-Affine</td>
<td>0.932</td>
<td>0.943</td>
</tr>
<tr>
<td>Hessian-Affine</td>
<td>0.921</td>
<td>0.930</td>
</tr>
<tr>
<td>Hessian-Affine</td>
<td>0.914</td>
<td>0.908</td>
</tr>
<tr>
<td>DOG</td>
<td>0.974</td>
<td>0.969</td>
</tr>
<tr>
<td>DOG</td>
<td>0.936</td>
<td>0.945</td>
</tr>
<tr>
<td>DOG</td>
<td>0.923</td>
<td>0.915</td>
</tr>
</tbody>
</table>

**TABLE 4. ACCURACY OF SIFT DESCRIPTOR WITH AND WITHOUT PCA.**

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Without PCA</th>
<th>With PCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Danish1991</td>
<td>0.974</td>
<td>0.986</td>
</tr>
<tr>
<td>Danish1996</td>
<td>0.969</td>
<td>0.982</td>
</tr>
<tr>
<td>Danish2008</td>
<td>0.958</td>
<td>0.971</td>
</tr>
</tbody>
</table>

**TABLE 5. ACCURACY OF THE PROPOSED APPROACH WHEN ONLY ONE REGION IS CONSIDERED.**

<table>
<thead>
<tr>
<th>Region</th>
<th>Table 1:</th>
<th>Table 2:</th>
<th>Table 3:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_1 = 1$</td>
<td>0.562</td>
<td>0.542</td>
<td>0.501</td>
</tr>
<tr>
<td>$\omega_2 = 1$</td>
<td>0.673</td>
<td>0.601</td>
<td>0.625</td>
</tr>
<tr>
<td>$\omega_3 = 1$</td>
<td>0.723</td>
<td>0.708</td>
<td>0.698</td>
</tr>
<tr>
<td>$\omega_4 = 1$</td>
<td>0.832</td>
<td>0.851</td>
<td>0.891</td>
</tr>
<tr>
<td>$\omega_5 = 1$</td>
<td>0.932</td>
<td>0.928</td>
<td>0.940</td>
</tr>
</tbody>
</table>

**Table 6. COMPARISON WITH STATE-OF-ART METHODS.**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Danish1991</td>
<td>0.968</td>
<td>0.922</td>
<td><strong>0.984</strong></td>
</tr>
<tr>
<td>Danish1996</td>
<td>0.975</td>
<td>0.931</td>
<td><strong>0.985</strong></td>
</tr>
<tr>
<td>Danish2008</td>
<td>0.962</td>
<td>0.921</td>
<td><strong>0.975</strong></td>
</tr>
</tbody>
</table>

**Region weight $\omega_i$**

As stated in Section II-B, we assign a weight $\omega_i (i = 1, 2, \ldots, M)$ to the Fisher vector from each region $R_i$ when generating the coin image representation. $M$ is set to 5 in our study which is a reasonable value according to pilot tests. We first evaluate the performance of the proposed approach when only one region is considered, which means that $\omega_1 = 1$ and $\omega_j = 0 (j \neq i)$. The results are shown in Table 5, where one can observe that region $R_5$ performs the best, while the worst performance is achieved with region $R_1$. Consequently, the closer to the rim of the coin a region is, the more important it is.

When all the regions are considered, different values of $\omega_i$ have been tested and the best performance is achieved with $\omega_1 = 0.1$, $\omega_2 = 0.1$, $\omega_3 = 0.1$, $\omega_4 = 0.3$ and $\omega_5 = 0.4$, obtaining the highest accuracies of 0.986, 0.982 and 0.971 on the three datasets, respectively.

It should be noted that when all the $\omega_i$'s are the same, viz. $\omega_i = 0.2 (i = 1, 2, \ldots, 5)$, the accuracies on the three datasets are: 0.802, 0.793, 0.753, which degrade significantly. Consequently, it is important to assign different weights to different regions when generating the representation of the coin.

**C. Comparison with state-of-the-art methods**

We compare the proposed approach with several state-of-the-art fake coin detection methods. To be more specific, our previous work [10] and Sun et al.’s method [9] are employed for comparison as shown in Table 6.

From the table, the proposed approach in this paper outperforms the other two methods. Sun et al.’s [9]
method works well when the coin images are of good quality. However, when the quality of the image deteriorates, segmenting features is a nontrivial task, which adversely impacts its performance.

IV. Conclusion
A visual appearance-based approach for fake coin detection is proposed in this paper. We first detect keypoints in the coin images. To find out the roles that different parts of the coin play in detecting fake coins, the coin image is divided into several regions. The keypoints from each region are then encoded as a Fisher vector. The coin image is finally represented by concatenating the Fisher vectors of all the regions in a weighted manner. We evaluate the proposed approach on three different datasets and achieve promising results. According to the experiments, the regions that are closer to the rim of the coin play a more important role in fake coin detection.

Acknowledgment
This work is supported by National Natural Science Foundation of China (No. 61603256), the Open Research Fund of Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University, and Natural Sciences and Engineering Research Council of Canada.

References
Oral Session K

Video processing and surveillance

Thursday May 17, 2018, 10:00 AM

Vishal Batchu, Varshit Battu, Dakannagari Mohana Murali Krishna Reddy and Radhika Mamidi
“How to rate a video game?” - A prediction system for video games based on multimodal information

Florent Lefevre, Vincent Bombardier, Nicolas Krommenacker, Patrick Charpentier and Petat Bertrand
Automatic video stream selection method by on-air microphone detection

Minji Lee, Benjamin Baird, Olivia Gosseries, Jaakko Nieminen, Ji-Hoon Jeong, Giulio Tononi and Seong-Whan Lee
Classification of the Levels of Consciousness within Non-Rapid Eye Movement Sleep

Yan-Lin Chou and Daw-Tung Lin
Robust and Adaptive Vehicle Detection System Using Surveillance Videos

Brian Gauch and Richard Peters
Animation Generation with a Low-Dimensional Simplicial Complex
“How to rate a video game?” - A prediction system for video games based on multimodal information

1st Vishal Batchu  
Centre for Visual Information Technology  
International Institute of Information Technology Hyderabad  
Hyderabad, India  
vishal.batchu@students.iiit.ac.in

2nd Varshit Battu  
Language Technologies Research Center  
International Institute of Information Technology Hyderabad  
Hyderabad, India  
battu.varshit@research.iiit.ac.in

3rd Murali Krishna Reddy  
Language Technologies Research Center  
International Institute of Information Technology Hyderabad  
Hyderabad, India  
murali.dakannagari@research.iiit.ac.in

4th Radhika Mamidi  
Language Technologies Research Center  
International Institute of Information Technology Hyderabad  
Hyderabad, India  
radhika.mamidi@iiit.ac.in

Abstract—Video games have become an integral part of most people’s lives in recent times. This led to an abundance of data related to video games being shared online. However, this comes with issues such as incorrect ratings, reviews or anything that is being shared. Recommendation systems are powerful tools that help users by providing them with meaningful recommendations. A straightforward approach would be to predict the scores of video games based on other information related to the game. It could be used as a means to validate user-submitted ratings as well as provide recommendations. This work provides a method to predict the G-Score, that defines how good a video game is, from its trailer (video) and summary (text). We first propose models to predict the G-Score based on the trailer alone (unimodal). Later on, we show that considering information from multiple modalities helps the models perform better compared to using information from videos alone. Since we couldn’t find any suitable multimodal video game dataset, we created our own dataset named VGD (Video Game Dataset) and provide it along with this work. The approach mentioned here can be generalized to other multimodal datasets such as movie trailers and summaries etc. Towards the end, we talk about the shortcomings of the work and some methods to overcome them.

Index Terms—Rating Prediction, Game Trailers, Game Summaries, Deep Learning, Neural Networks

I. INTRODUCTION

Video games are almost everywhere these days, from individual consumers who play video games for fun to serious E-Sports professionals. The video game industry is a billion dollar industry, it was valued at $44.9 billion back in 2007 which rose to $91.5 billion in 2015. The increase in the rate of development of games and the number of people who play these games spiked up hand in hand throughout the world over the recent years. This increase in the sheer number of games marketed required people to rely on a trusted resource that would give them information about these games since it is infeasible for a human to keep the details of every single game ever released in memory. Another trend observed in recent times is that there is an exponential increase in the amount of data shared online. This, however, comes with certain unforeseen consequences such as a reduction in the quality of data present online, the spread of bogus information i.e false information being shared online. Considering the video game industry, people often rely on various sites to provide them with ratings, reviews etc of games before purchase. Since most ratings and reviews are submitted by a wide array of users, maintaining them is hard and hence, we end up having a lot of incorrect/unwanted entries. Another issue we often face with simple methods of input is that users might unknowingly select the wrong option such as an incorrect rating or a genre for a video game. Reviews and descriptions don’t face this issue since textual inputs have lesser tendency to be incorrectly entered, however not many people would be willing to spend their time adding textual information and hence we see a wide use of simple input methods. Recommendation systems are quite popular since they allow us to provide meaningful options to users for various purposes. Deep learning has shown a lot of promise at this task. We define the G-Score of a game as a value that determines how good a game is which is derived from critic and user game ratings. In order to mitigate the issues mentioned earlier and to offer useful recommendations to users, we propose several deep neural network architectures that would predict the G-Score from the trailer and the summary of a video game. We believe that the use of summaries along with the trailers would aid the model to predict the G-Score better compared to the use of trailers alone. This would also aid game developers while creating trailers to see how well they score before a public release since the predicted G-Scores could be used to refine and improve the trailers. In order to train our models, we have created the VGD dataset and provide it along with this work.

II. RELATED WORK

There have been multiple works in areas related to video and text classification, however, they often deal with domain
specific information. Nominal work has been done on video game trailers in the past. We use video game trailers along with reviews to perform a cross-domain analysis in order to predict ratings.

**Video Analysis and Classification** - Zhang et al. [11] propose a supervised learning technique for summarizing videos by automatically selecting key-frames. They use Long-Short-Term Memory to model the variable-range temporal dependency among frames so that both representative and compact video summaries can be generated. Venugopalan et al. [2] look into how linguistic knowledge taken from large text corpus can aid the generation of natural language descriptions of videos. Haninger et al. [5] quantified and characterized the content in video games rated T (for "Teen") and measure how accurate the ESRB-assigned content descriptors displayed on the game box are to the real game. Simonyan et al. [4] investigate architectures of discriminatively trained deep Convolutional Networks for action recognition in videos. Capturing the complementary information from still frames and motion between frames was a challenge they address. Kahou et al. [1] present an approach to learn several specialist models using deep learning techniques. Among these are a convolutional neural network focusing on capturing visual information in detected faces, a deep belief net which focuses on the representation of the audio stream, a K-Means based “bag-of-mouths” model, which extracts visual features around the mouth region and a relational auto-encoder, which addresses spatiotemporal aspects of videos. Le et al. [7] present an unsupervised feature learning as a way to learn features directly from video data. They presented an extension to the Independent Subspace Analysis algorithm to learn invariant spatiotemporal features from unlabeled video data. Zhou et al. [7] formalize multi-instance multi-label learning in which each training example is associated with not only multiple instances but also multiple class labels. They propose algorithms for scene classification based on the relationship between multi-instance and multi-label learning.

**Text Analysis and Classification** - Glorot et al. [8] propose a deep learning approach that learns to extract a meaningful representation for each review in an unsupervised manner. Sentiment classifiers trained with this high-level feature representation clearly outperform state-of-the-art methods. Zhang et al. [9] show empirical exploration on the use of character-level convolutional networks (ConvNets) for text classification. They built large-scale datasets to show that character-level convolutional networks can achieve state-of-the-art results. Iyyer et al. [10] present a simple deep neural network that competes with and sometimes outperforms models on sentiment analysis and factoid question answering tasks by taking only a fraction of the training time. Baker et al. [11] describe the application of Distributional Clustering to document classification. Their approach clusters words into groups based on the distribution of class labels of each word. Unlike techniques such as Latent Semantic Indexing, they were able to compress the feature space, while maintaining the classification accuracy. Poria et al. [12] use the extracted features in multimodal sentiment analysis of short video clips representing one sentence each. They use the combined feature vectors of textual, visual, and audio modalities to train a classifier which is based on multiple kernel learning, which is known to be good at heterogeneous data. Zhang et al. [13] mention that this learning problem is addressed by using a method called $M_{LNB}$ which adapts the traditional naive Bayes classifiers to deal with multi-label instances. Feature selection mechanisms are incorporated into $M_{LNB}$ to improve its performance.

### III. Dataset

We have created a dataset named VGD that consists of the trailer, summary, developer, age rating, user-score, critic-score and genre of 1,950 video games. The data was collected from metacritic.com. The dataset along with the code used can be found at [https://goo.gl/7SbN1s](https://goo.gl/7SbN1s) for replicability and future use. This is the first dataset of its kind and we believe it would be quite helpful to the research community.

<table>
<thead>
<tr>
<th>Genre</th>
<th>Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role-Playing</td>
<td>522</td>
</tr>
<tr>
<td>Strategy</td>
<td>329</td>
</tr>
<tr>
<td>Action</td>
<td>734</td>
</tr>
<tr>
<td>Sports</td>
<td>200</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>165</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Age Rating</th>
<th>Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mature</td>
<td>449</td>
</tr>
<tr>
<td>Adults Only</td>
<td>2</td>
</tr>
<tr>
<td>Everyone 10+</td>
<td>365</td>
</tr>
<tr>
<td>Teen</td>
<td>704</td>
</tr>
<tr>
<td>Everyone</td>
<td>409</td>
</tr>
<tr>
<td>Other</td>
<td>522</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>G-Score (S)</th>
<th>Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-10</td>
<td>0</td>
</tr>
<tr>
<td>11-20</td>
<td>3</td>
</tr>
<tr>
<td>21-30</td>
<td>7</td>
</tr>
<tr>
<td>31-40</td>
<td>19</td>
</tr>
<tr>
<td>41-50</td>
<td>89</td>
</tr>
<tr>
<td>51-60</td>
<td>263</td>
</tr>
<tr>
<td>61-70</td>
<td>433</td>
</tr>
<tr>
<td>71-80</td>
<td>693</td>
</tr>
<tr>
<td>81-90</td>
<td>416</td>
</tr>
<tr>
<td>91-100</td>
<td>27</td>
</tr>
</tbody>
</table>

**TABLE I**

**DISTRIBUTION OF VGD ACCORDING TO GENRE CLASSES, G-SCORE CLASSES AND AGE RATINGs**

### A. Preprocessing

The first step in preprocessing involved the removal of certain games that had missing details (a lot of games did not have trailers).

**Trailers** - The trailers extracted from the website had a resolution of 720p (with a few exceptions). We reduced the resolution to 360p since 720p required more space and mostly consisted of redundant information from the view of a neural network. We put an upper limit of 3 minutes for each trailer, trimming trailers that were larger to the 3-minute mark.

**Summaries** - We remove non-ASCII characters from the summaries since some of the summaries had terms from other languages like Japanese, Korean, French etc. However, since they are quite small in number, including them would not provide much value in terms of generalizability of the approach.

The final dataset consists of 1,950 video game trailers and summaries.

B. Statistics

Various statistics related to the dataset are provided, that show the diversity of the dataset. Video games were collected from a wide range of over 730 developers. Games span across various age ratings from E (Everyone) to M (Mature) which provides us a wide collection of games.

Genres - We cluster the genres into 5 groups based on similarity as specified in Table II and present the number of games belonging to each group in sub-tables in Table I.

<table>
<thead>
<tr>
<th>Genre-Class</th>
<th>Genres</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role-Playing</td>
<td>Adventure, First-Person, Third-Person, Role-Playing</td>
</tr>
<tr>
<td>Strategy</td>
<td>Turn-Based, Strategy, War-Game, Puzzle, Platformer</td>
</tr>
<tr>
<td>Action</td>
<td>Action</td>
</tr>
<tr>
<td>Sports</td>
<td>Fighting, Sports, Racing, Wrestling</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>Simulation, Flight, Party, Real-Time</td>
</tr>
</tbody>
</table>

Table II
Our proposed grouping of genres into 5 classes based on similarity

Game scores - We define the G-Score of a game as an average of critic and user ratings, details are specified in Section IV. We observe that most games have G-Scores above 40 and only a small fraction of games have a G-Score below 40 as shown in Table II. This results in some inter-class bias. The main reason for this is that most video games that would potentially have a bad G-Score would either not have trailers or not have any associated critic/user ratings since most people would not play the game in the first place and hence would not be present in the dataset.

IV. SCORE PREDICTION

Each video game has a user rating \( R_u \) and critic rating \( R_c \) associated with it. We define the G-Score of a game (\( S \)) as follows,

\[
S = \frac{R_u + R_c}{2}
\]

and aim to predict this G-Score. The G-Score essentially represents how good a game is.

Critic ratings are collected from a large number of critics and a weighted average is computed to form the final critic rating \( R_c \). The weights of individual critics depend on the overall stature of the critic. Formally, if \( R_c^i \) corresponds to the rating of critic \( i \) and \( \alpha^i \) is the weight associated with critic \( i \) and there are \( M \) critics then,

\[
R_c = \sum_{i=1}^{M} \alpha^i \cdot R_c^i
\]

The number of critics that review games vary from game to game since popular games often get a larger number of ratings as compared to others that are not so popular. Critic weights \( \alpha^i \) are based on how well the critics performed in the past (well written, insightful reviews etc). This is determined by Metacritic staff who handle the website from where we collected our data.

The user rating \( R_u \) is computed as an average of all user ratings submitted for the game. Formally, if \( R_u^i \) corresponds to the rating of user \( i \) and there are \( N \) users then,

\[
R_u = \sum_{i=1}^{N} R_u^i
\]

Regardless of the number of users and critics that rate a game, we compute the final score as mentioned in Equation 1. We consider both trailers and summaries as inputs in order to predict this G-Score using our proposed model. We quantize the G-Scores to 10 classes since predicting the G-Score directly is a regression problem which is harder to tackle compared to classification problems.

A. Trailers

Each video game is associated with a trailer that we use in order to predict the G-Score.

Trailer frame selection - Since videos are captured with a frame-rate of 24 fps, it is infeasible to use them as they are since the sheer number of frames are too many. Hence, we propose a method to pick frames in a certain manner that would allow us to maximize the information we obtain from game trailers. Firstly, we reduce the frame-rate to 4 fps while extracting the frames from the video. We then follow the frame selection algorithm mentioned in Algorithm I in order to select frames. This allows us to capture important information at various parts of trailers. The reason we skip frames is that most trailers have a sequence of events that go on for a while before transitioning to the next sequence of events. Upon observation, we use a skip of 150 frames as a good approximation. We skip the first 50 frames since most trailers have textual information during the start of the video such as the developer titles, age ratings etc.

Algorithm I Frame selection for trailers:

1. Consider we have a set of \( N \) frames \( F_1, F_2, \ldots, F_N \)
2. \( f_{\text{start}} = 50 \)
3. while \( f_{\text{start}} < N \) do
4.    for \( j = 0, j++, \) while \( j < 10 \) do
5.        if \( f_{\text{start}} + j <= N \) then
6.            Select frame \( F_{f_{\text{start}}+j} \)
7.        else
8.            Break
9.    \( f_{\text{start}}++ = 150 \)

Trailer features - We use the pre-trained Inception-V3 [13] model to extract features from each of the frames selected in the previous step. The model was pre-trained on ImageNet [13] and hence generalizes well to a wide range of images. We extract the features of the \textit{Avg Pool} layer (the penultimate layer in the network) which gives us a feature representation of 2048 elements per frame. Considering all the frames, we get a
Fig. 1. An overview of our pipeline corresponding to Model-1. We start with the trailers and summaries as inputs and predict the G-Score classes as outputs. The Inception-V3 pre-trained model is used to extract features of video frames. ConvPoolBlocks and ConvBlocks are described in Figure 2. The output sizes at each of the layers are mentioned in the figure.

vector having dimensions \((M, 2048)\) where \(M\) is the number of frames we selected by the frame selection algorithm 1 as our final trailer features.

C. Method

**Model-1** - We propose a deep learning based architecture as outlined in Figure 1 that uses a combination of recurrent and convolution based networks which allows us to process both trailer features and summaries in order to predict the G-Score of a video game. The frame features are fed to multiple levels of LSTMs [16] that finally output a vector of size 512. The summaries are fed to an embedding layer that dynamically generates embeddings having a size of 300. These embeddings are then fed to a convolution-based network as depicted in Figures 1 and 2 that outputs a vector of size 512. Finally, these vectors are concatenated and passed along to a linear layer that outputs the G-Score class. We also perform experiments on multiple other model architectures, however, this gives us the best results.

**Model-2** - We use a time distributed CNN over extracted frame features to generate a small embedding for each frame which are then concatenated and fed to a fully connected layer in order to produce the final output vector. Summaries are passed through a CNN, similar to what was done earlier. The outputs from both the LSTM and CNN are concatenated and a linear layer is applied to predict the final output. One significant advantage of this approach was that the model had a very small number of parameters since the time distributed CNN shares weights across time. This would be an ideal model to use in memory constrained scenarios such as mobile computing.

**Model-3** - We use a 3D CNN [17] over the frames to generate an output embedding for the trailer. Summaries are passed through a CNN, similar to what was done earlier. The outputs are then concatenated and passed to a linear layer to predict the final G-Score class. We also tried generating sentence embeddings using Doc2Vec [18] for each of the summaries but they didn’t give

---

**B. Summaries**

Considering all the summaries we have, we create a dictionary where each word is given an index. We then go through each of the summaries replacing words with their corresponding indexes. Finally, we resize the summaries to a size of 100 by trimming the summaries if they are larger and padding them with zeros if they are smaller.

---

**Fig. 2.** ConvPoolBlocks and ConvBlocks are used in order to process the summaries in our proposed models. A ConvPoolBlock consist of Convolution, Tanh, MaxPool and Dropout layers. A ConvBlock consists of Convolution, Tanh and Dropout layers.
Fig. 3. Qualitative examples where the model predicts the correct G-Score class for Super Mario Odyssey (left) and the incorrect G-Score class for Gran Turismo 6 (right). The true G-Score for Super Mario Odyssey is 93 and the true G-Score for Gran Turismo 6 is 81.

us the best results and hence we stuck to dynamic embeddings as mentioned earlier. The three models mentioned here consider both the trailer and summary as inputs. In order to validate our claim that the use of summaries gives us accuracy improvements, we also perform the same experiments without considering summaries on each of the proposed models and report accuracies in Table III. This shows that using summaries along with trailers gives us significant improvements of over 5%.

D. Implementation Details

We implemented the proposed models in the Keras [19] framework over the Tensorflow [20] backend. We use the Cross-entropy loss at the output of our model and use the Adam optimizer with a learning rate of 1e-4 and a decay of 1e-6 in order to train the model. We use tanh activations instead of ReLU throughout the model as it helps us achieve better accuracies. We also include multiple Dropout [21] layers to allow the model to generalize well.

To evaluate our model, we perform 10-fold cross-validation and provide results. Further details on Model-1 (our best model) can be found in the code submitted along with this work at https://goo.gl/IYiEfq.

V. RESULTS

On each of our proposed models, we perform 10-fold cross-validation and consider the mean as our final accuracy. We observe a significant increase in accuracy with the inclusion of summaries as inputs along with the trailers. Model-1 gives us the best results in terms of accuracy. We believe the main reason for this is that Inception-V3 is trained on ImageNet which is a huge dataset of more than 1M images. Hence, it provides us with feature representations that are rich and meaningful.

Model-2 has a very small number of parameters which is why it is well suited for use in portable devices and memory constrained situations such as mobile processing. This, however, comes at a cost that the accuracy is lower than Model-1.

A. Qualitative Analysis

A few qualitative results have been provided where the network performs well in one case but fails at the other as shown in Figure 3. Gran Turismo 6 has a true G-Score of 81 but we predict a G-Score class of 40-50. The main reason this fails is that the trailer has multiple overlay texts and game-play irrelevant clips. A simple solution to frames containing overlay text is to ignore them before feeding them to model. We could also process these frames separately, extracting the text from...
them and using them as inputs along with the summaries of games. Since non game-play scenes do not contribute any significant information when scoring a game, the model would misinterpret this information hence resulting in incorrect G-Scores. In the example provided, refer Figure 3, the frame containing the person would get a feature representation from Inception-V3 that has no relevance to the game and would ultimately contribute to noise. Handling non-game-play scenes in trailers is an issue that is hard to tackle and is one of the shortcomings of this work. An approach towards this would be to train a model that takes a frame as an input and predicts if the frame is a game-play scene or not given the video as a reference.

B. Empirical Validation

We validate our claim that summaries provide information that is quite useful while predicting the G-Score of a video game. Hence, using both the trailers and summaries allows us to predict with a much better accuracy. We conduct a significance test where we perform experiments on predicting the G-Score of a video game based on the trailer alone and show that we gain significant accuracy improvements of over 5% when we use both the summaries and trailers in order to predict the G-Score as mentioned in Table III. Most of the times, we have the summary at our disposal along with the trailers of games and hence, using information from multiple modalities helps us develop models that perform better.

VI. Conclusion and Future Work

In this work, we show how valuable multimodal knowledge is at performing a task at hand. In most real-life scenarios we would have multimodal information available which could be utilized to train better models. We also provide a new VGD dataset that is a dataset on video games, a first of its kind. We propose multiple models that work under different scenarios such as memory constrained settings etc. We plan to apply our approach to movie trailers and summaries in order to show the generalizability of our approach. We plan to take care of overlay texts that occur in trailers by processing them separately in order to produce better results. Finally, we also plan to include audio in order to improve our prediction accuracies.

References

Automatic video stream selection method by on-air microphone detection

Lefevre Florent *, †, Bombardier Vincent *, Krommenacker Nicolas *, Charpentier Patrick * and Petat Bertrand †

*Université de Lorraine
CRAN CNRS UMR 7039,
F-54506 Vandoeuvre-les-nancy
firstname.lastname@univ-lorraine.fr
†CitizenCam
132 rue André Bissiaux
54320 Maxéville
(flefevre, bpetat)@citizencam.eu

Abstract—This article presents an automatic video editing method for video stream selection in a multi-camera environment. The specific context of this study is council meetings recording and broadcasting. In order to offer the best view to spectator our method is based on a speaker detection, to select the right camera. Since no sound information is available, the proposed method is based on the detection of the change in the visual state of the microphones LED in image sequences, in order to automatically and efficiently select the camera where the speaker is. Studies about the suitable size of the used sliding window and about the relevant features' selection for the verification of microphones' activation are also presented. We have selected seven features to effectively train one classifier, which can be used on different cameras. The feasibility of this approach is shown by the experimentation on councils' videos where the proposed method allows a very efficient detection of the speaker in real-time.

Index Terms—Automatic video editing, Speaker Detection, Feature selection, LED detection

I. INTRODUCTION

Automatic video editing allows small events to be available to a much larger audience. Indeed, many events cannot be broadcasted because of the cost of the human production crew and equipment. By Automatic video editing, we mean automatic selection of the best viewing angle in a multi-camera system, in order to provide to the spectator the video stream where the action take place. CitizenCam † is a French company which offers multi-camera automatic recording solutions in order to retransmit on the web every type of event. Their goal is to reduce costs by automating recording and broadcasting while using IP cameras. This in order to be affordable to the greatest number of people. In the context of council meetings, around ten cameras are filming the potential speakers from the center of the room. Searching in the ten proposed views, the one where the speaker is located is tedious for the spectator. In addition, live broadcasting of an event requires selecting the most relevant view for bandwidth saving reasons. That’s why automatic view selection is mandatory in order to improve the user experience.

In [1], the authors describe autonomous camera systems as a system which have to solve three simultaneous problems: find out how the camera should be oriented to capture the action, define how the camera should move to film the subject and finally select the right camera which should be on air. In our case study, we use fixed camera so our problem is to find out if there is a person speaking on each camera to select the most relevant camera.

In literature, speaker detection or localization method generally need audio systems to confirm the presence of a talking person. Some methods [2]–[5] only use microphones array in order to localize the person talking in a room. Some other methods [6]–[11], use both video and audio in order to retrieve the person which is currently speaking. In [12], the authors propose an automating camera management to record lectures. The system automatically select the view where a person is talking: either the professor, or someone from the audience. In order to do that, they use a two-microphone array to estimate sound source location.

In our context, many cameras are filming the different speakers. However the audio systems used in most of the council meetings don’t allow the isolation of each microphone, so we can’t use this kind of technique. Changing the microphone system will greatly increase costs, making broadcasting impossible.

Lip movement analysis [13], [14] could be an alternative to detect current speaker in a video stream. However, these methods require a close-up and frontal view of the speakers, which is not possible due to the large number of potential speakers. In addition, it is common for people to talk to each other, although they are not the expected speakers.

In order to detect the speakers, we offer to detect the light emitted by the microphone when a person speaks. Indeed, most of the French municipality is equipped with microphone systems where a LED lights up on the microphone when some-
one speaks. The detection of this light allows the detection of the speaker without changing the actual system. Even if those research have been made for traffic light detection, thus treat a different environment than the one we’re interested in, light source’s recognition is regularly the base of the work. The offered algorithms rely on the extraction of candidate regions using color thresholding methods. In this methods, [15], [16] are using the HSV color model, whereas [17], [18] are working on normalized RGB color models.

The first part of this paper explains the methodology used for the speaker detection. We detail the method proposed and make a study of the features used in the microphone verification step. Our second step will be presenting the videos used to validate our method and the results we obtain. Finally, we’ll introduce how we can improve this method.

II. SPEAKER DETECTION METHOD

The method we propose is therefore based on the search for a light source in the image, symbolizing the activation of a microphone. Due to a number of light disturbances, it is necessary to check whether the received light is that of a microphone. One of the objectives being the generalization of the method to be used in different conditions, we will look for the features allowing the best characterization of microphones.

The proposed speaker detection method is divided in four principal parts and is described in the first paragraph. The second defines the choosing principle of the size of the sliding windows for the microphone detection. The third part details the choice of attributes used for the validation step of the speaker presence allowing a generalization of the method for different viewing angles.

A. Method description

The method presented in this paper detects light from the microphone, in order to find the people who is speaking. The principle of our method is shown in fig. 1 where each process is executed sequentially. In order to use this method in every situation, we introduce an initialization step to manually select the limits of the research region and the HSV threshold.

Selection of the Region of Interest (ROI): The first step of our method is the selection of the ROI involving a reduction of computation’s time and the risk of wrong detections. Since microphones are located on the table in front of every eventual speaker, it is appropriate to look for microphones between the table and the top of the head of speakers. Because the microphones are not fixed on the tables, we need to define a large area in case the speakers move the microphones. An example of ROI selection is shown in Fig. 2 from the video "Sight 1", where the research zone measures 1800 x 340 pxl (cf Table I).

Color space selection: The second step of the method is to prepare for the light detection stage. To do this, it is necessary to change the color representation. Different color spaces were tested (RGB, HSV [19], CIE L*a*b* [20]). Those color spaces, frequently used in colorimetry, allow an efficient representation of the luminosity in the image, especially with the Intensity component of the HSV color space or the Luminance component of the L*a*b* color space. The choose of the HSV color space was made in line with the selection of relevant features (II-C).

Detection of light sources: In the first place, we want to localize microphones which are currently used. In other words, we want to find light sources emitted by active microphones. The HSV color model and especially the V component allows to efficiently find them. We execute thresholding to get candidate regions, as we can see in the following equation:

$$C(x, y) = \begin{cases} 1 & \text{if } ((S(x, y) \geq T_{s1}) \cap (S(x, y) \leq T_{s2}) \cap (V(x, y) \geq T_{v1}) \cap (V(x, y) \leq T_{v2})) \\ 0 & \text{else} \end{cases} \quad (1)$$

Where $C(x, y)$ is the result of the thresholding operation, $S(x, y)$ and $V(x, y)$ are the saturation and intensity values. The four threshold values ($T_{s1}$, $T_{s2}$, $T_{v1}$ and $T_{v2}$) are empirically defined.

The lighting condition may change during the videos, therefore many reflections may occur, resulting in small lighting sources in the thresholded image (see Fig. 3). We apply a connected component analysis [21] in order to execute a dimensional thresholding. The light sources which are inferior (like reflections) or superior (like lighting) to the light source from a microphone are ignored.
TABLE I
CHARACTERISTICS OF VIDEOS USED

<table>
<thead>
<tr>
<th>Name</th>
<th># images</th>
<th>Research zone</th>
<th>Area of microphone</th>
<th>pixel per mm</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sight 1</td>
<td>9 957</td>
<td>1800 x 340</td>
<td>345 pxl</td>
<td>0.71</td>
<td>Front view in close-up</td>
</tr>
<tr>
<td>Sight 2</td>
<td>22 459</td>
<td>830 x 230</td>
<td>190 pxl</td>
<td>0.23</td>
<td>Front view, wide shot</td>
</tr>
<tr>
<td>Sight 4</td>
<td>15 737</td>
<td>1000 x 300</td>
<td>120 pxl</td>
<td>0.21</td>
<td>Side view, wide shot</td>
</tr>
<tr>
<td>Sight 6</td>
<td>32 508</td>
<td>1000 x 220</td>
<td>210 pxl</td>
<td>0.30</td>
<td>Side view, wide shot</td>
</tr>
</tbody>
</table>

Speaker verification: In order to check that the light sources are from microphones’ LED, we use a classification tree [22] and a sliding windows techniques to separate zones that contain one active microphone from another. For each light sources, a windows of size 19x19 (see II-B), is swept across the candidate regions. The features calculated (see II-C) in each window location are then tested with a classifier. The classification tree is created during the initialization step of our system. This kind of classifier is used because of its computational fastness allowing to respect the real-time constraints.

B. Windows’ size selection

The selection of the size of the window for feature calculation is an important step for the characterization of the microphone state. In order to determine the best size to use, three windows were tested: a small one of 3x3 pixels, a medium one of 9x9 pixels and a big one of 19x19 pixels, as shown in fig. 4.

The use of a small window doesn’t permit to separate efficiently the two classes in every situation. Indeed, a small window doesn’t represent all the structure of the microphone, that may explains the results obtained with images from sight 4 and sight 6. The medium and the big windows permit a better separation of those classes. The windows’ size 19x19 are used for the microphone’s presence check, thanks to its stability in orientation changes.

C. Feature selection

A preliminary step in the microphone’s LED detection is to find the right measure to characterize them in the right way. In order to find those features, a great number of statistical measures were calculated for the three components of the tested color spaces. The most relevant were selected in order to determine those having the biggest impact on the detection of scenes where there is an active microphone from those with none. The following features were calculated in every components histograms:

- The mean (m), mode (mo), variance (σ²) and standard deviation (σ).
- The Khi-2 (K2).
- The third central moment (skewness - sk), a measure of the asymmetry, as well as his third root (3√sk) for scaling reasons.
- The fourth central moment (kurtosis - ku), a measure of the flattening coefficient, as well as his fourth root (4√ku).

These characteristics have been selected in order to respect the real-time constraints. The treatments are based on the statistical analysis of the histogram and not on the calculation of the morphometric characteristics of the microphones which are not relevant. The calculation of these 81 features is however extremely time-consuming. Several feature selection methods were used in order to only keep those having a great discriminatory power.

We can identify three types of features:

- Complementary: those which, combined, allow a better differentiation of classes

The use of a small window doesn’t permit to separate efficiently the two classes in every situation. Indeed, a small window doesn’t represent all the structure of the microphone, that may explains the results obtained with images from sight 4 and sight 6. The medium and the big windows permit a better separation of those classes. The windows’ size 19x19 are used for the microphone’s presence check, thanks to its stability in orientation changes.

TABLE II
INFLUENCE OF THE WINDOWS’ SIZE ON CLASSIFICATION

<table>
<thead>
<tr>
<th>Size</th>
<th>3x3</th>
<th>9x9</th>
<th>19x19</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sight 1</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Sight 2</td>
<td>97.9%</td>
<td>99.6%</td>
<td>100%</td>
</tr>
<tr>
<td>Sight 4</td>
<td>88.6%</td>
<td>95.6%</td>
<td>98.2%</td>
</tr>
<tr>
<td>Sight 6</td>
<td>87.7%</td>
<td>96.8%</td>
<td>98.1%</td>
</tr>
</tbody>
</table>

In order to check the windows size, a specific image database has been created. For each size of windows, 500 images from active microphones and 500 images from other parts were extracted from each the four videos explained in Table I. Classification trees were trained using the features (presented in II-C), extracted from these 4000 images. We use a k-fold cross-validation [23] with k=3 in order to prevent overfitting. Each trained classifier was next tested on each video, which will be presented in detail in III-A. The table II sum up the obtained results in term of accuracy.
The features in the HSV color space, especially the Hue, are important for the characterization of active microphones. The results show that the selection of characteristics applied on the images extracted from the 4 videos of the dataset. The results show that the features in the HSV color space, especially the Hue are important for the characterization of active microphones. The low number of selected features of component L is due to the fact that the information is redundant with that of component V.

The seven recurring features to each picture library have been selected:

- Mean, variance and third central moment of the H component
- Mean of the S component,
- Mean, variance and the V component’s Kurtosis fourth root.

These are the features that best differentiate an on-air microphone from an off microphone. These results show the importance of taking into account the average and dispersion of the microphone’s hue and luminance for class separation.

These characteristics are thus used during the creation and during the use of the decisional tree for microphone’s presence check.

### Table III

<table>
<thead>
<tr>
<th>Methods</th>
<th>RELIEFF</th>
<th>SFS</th>
<th>SFFS</th>
<th>SBS</th>
<th>SBFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sight 1</td>
<td>$\sqrt{s}H; mV; \sigma^2V; mS; \sigma^2V$</td>
<td>mV; mH; mH; $\sqrt{s}H; mS$</td>
<td>mV; mH; mH; $\sqrt{s}H; mS$</td>
<td>$\sigma^2H; mV; mS; \sigma^2V$</td>
<td>$\sigma^2H; \sqrt{sk}V; mV; \sigma^2V$</td>
</tr>
<tr>
<td>Sight 2</td>
<td>mH; skH; $\sqrt{sk}V; mV; mH; mH; mS$</td>
<td>mH; mH; mH; $\sqrt{s}H; mS$</td>
<td>mH; mH; $\sqrt{sk}V; mS$</td>
<td>mH; skH; mS; $\sigma^2V; \sqrt{sk}V$</td>
<td>mH; mH; mH; mV; $\sigma^2V; mV$</td>
</tr>
<tr>
<td>Sight 4</td>
<td>mH; $\sigma^2H; skH; mS; moS; \sigma^2V$</td>
<td>mH; $\sqrt{sk}V; mS; \sigma^2V; \sqrt{sk}V$</td>
<td>mH; $\sqrt{sk}V; mS; \sigma^2V; \sqrt{sk}V$</td>
<td>mH; skH; mS; $\sigma^2V; \sqrt{sk}V$</td>
<td>mH; mH; mH; $\sigma^2V; \sqrt{sk}V$</td>
</tr>
<tr>
<td>Sight 6</td>
<td>moV; $\sigma^2H; skH; mS; \sqrt{sk}V$</td>
<td>mH; $\sigma^2H; skH; \sigma^2V; \sqrt{sk}V$</td>
<td>mH; $\sigma^2H; skH; \sigma^2V; \sqrt{sk}V$</td>
<td>mH; mH; $\sqrt{sk}V; mS; \sigma^2V$</td>
<td>mH; mH; mH; mV; $\sigma^2V; \sqrt{sk}V$</td>
</tr>
</tbody>
</table>

- Redundant: those which bring identical informations
- Antagonistic: those which bring contradictory informations as for the separation of the classes.

The removal of antagonistic features allows to obtain a greater recognition rate. The suppression of redundant features allows to decrease computational cost.

Images from test sequences have been extracted (2 images per second) and reduced zones (19 x 19 pixels) around the microphones were selected, as well as images derived from zones with inactive microphones. Every reduced images was then annotated "positive" or "negative". In feature selection, filters methods are used when a large number of samples and features to be selected are available [24]. Since we have a reduced number of samples, we can use "wrapper" [25] methods. Moreover, these feature selection methods include a classification step, needing a labeled database. They provide more suitable selected feature set. The following reference methods are used:

- ReliefF [26]
- SFS : Sequential Feature Selection [27]
- SBS : Sequential Backward Selection [28]
- SFFS : Sequential Forward Floating Selection [29]
- SBFS: Sequential Backward Floating Selection [29]

For each dataset, by combining the results given by the algorithms of selection, a subset of 10 features was extracted. This subset gathers the most discriminating features for the separation of "switched on microphone" classes (positive images) and "rubbish" classes (negative images). The table III shows the five best features selected by each method for each video from our databases. The features from the HSV color space are those which characterize efficiently the microphones’ LED. The interest of using many methods is that the features selected are confirmed by all of them, as shown in table III.

For each dataset, by combining the results given by the algorithms of selection, a subset of 10 features was extracted. This subset gathers the most discriminating features for the separation of "switched on microphone" classes (positive images) and "rubbish" classes (negative images). The table III shows the five best features selected by each method for each video from our databases. The features from the HSV color space are those which characterize efficiently the microphones’ LED. The interest of using many methods is that the features selected are confirmed by all of them, as shown in table III.

The figure 5 summarizes the percentage of appearance of each characteristic, video dataset and algorithms of selection altogether. In other words, it is the number of occurrences of each characteristic in the results given by the 5 algorithms of selection of characteristics applied on the images extracted from the 4 videos of the dataset. The results show that the features in the HSV color space, especially the Hue are important for the characterization of active microphones. The

![Image](312x566)

Fig. 5. Percentage of appearance of the features from the 5 features selection algorithms apply on the 4 datasets. In green are represented the selected features.

2Features in green in the figure 5.
III. Validation

The evaluation of the proposed method was performed on a computer equipped with an Intel Core i7-5557U processor (Base Frequency 3.1 GHz) and 8GB RAM. The method was implemented in Python, using functions from OpenCV library.3

A. Videos database

The evaluation of the method was performed on videos from a city council, that was filmed in Villers-Lès-Nancy in France. Table I presents different videos that were used. Original images have a size of 1920x1080 pixels for the video "Sight 1" and 1080x720 pixels for the others. Since videos 2, 4 and 6 have similar characteristics, we can group them under the name DB2. The size of the zone corresponding to the region of interest is manually defined during the initialization step. Calculating times are really impacted by the definition of these zones. The microphone’s area is the average number of pixels in the reduced area (19x19 pixels) around the microphone: 361 pixels maximum. The properties are each sighting’s features. These test sequences were manually annotated to serve as the ground truth in order to estimate the efficiency of the method.

B. Classifiers

As we can see in I, there are two different types of views. The first one, “Sight 1”, is a camera directed at the mayor, with a higher resolution. The other cameras film advisors and have lower resolution. That’s why we use two different decision trees trained in an initialization step. The first one is trained from 500 images from the camera 1 and is then applied on the entire sequence of “Sight 1”. The second one is trained from 500 images from the camera 2 and applied on the other sequences: sight 2 , 4 and 6 (DB2).

C. Results

The table IV sums up the results obtained by using our method. The results are obtained thanks to the confusion matrix and are expressed in terms of precision, recall and accuracy [30].

The processing time per image (P. T. I.) expresses the average time (in milliseconds) needed to detect the presence of an active microphone in an image. This time depends on the search area defined during initialization.

The results obtained show the performance of the proposed method on the selected videos. The precision and the accuracy are in a 99% range and the recall about 98%. These results confirm the effectiveness of the features’ selection presented in part II-C. The processing time per image shows the ability to make it work in real time. In addition, the results obtained on DB2 show that it is possible to generate a general model, from one camera, for processing multiple cameras. The majority of false positives obtained are caused by light disturbance (reflections, smartphones,...). A longer learning time could maybe reduce those errors. False negatives are caused by a total occlusion of the microphone (when the speaker holds the microphone by its LED).

The presented results are obtained by using the proposed detection method on each image, without consideration of the video sequence’s dynamics. In the context of automatic editing, detection and loss of detection under 300ms are ignored. By doing so, occlusions and false positives are not felt by the user.

IV. Conclusion

The automatic video selection method offered is reliable in a council context and the consideration of the dynamics of the scene must improve the obtained results. We have selected seven features (mean, variance and third central moment of the H component, mean of the S component, as well as the mean, variance and the V component’s Kurtosis fourth root) that make it possible to obtain a microphone model that can be used for different cameras but have close viewing characteristics. This method is compatible with a real-time utilization and the selection of the stream of interest is then possible. Each time a new person speaks, we can change the stream being relayed in order to show this new speaker.

However we’re planning several solutions to improve it.

An upper-body detector should allow to look for microphones only where the eventual speakers are. Instead of a global research zone, we could obtain one zone for each speaker. It would result in a diminution of the computational cost and a decrease of false detection.

Taking into account the phenomena of scale should make it possible to generalize for all shots taken within a council, or even different installations.

The use of microphones’s LED as an informative medium [31] is also considered. The microphone will become an active component of the system, communicating an identification signal, owned by each speaker, that will be captured by the cameras. It will allow the verification of the detection as well as the identification of the speaker.

References


Classification of Sleep Consciousness based on Filter Bank Common Spatial Pattern

Minji Lee  
Dept. of Brain and Cognitive Engineering  
Korea University  
Seoul, Korea  
minjilee@korea.ac.kr

Benjamin Baird  
Dept. of Psychiatry  
University of Wisconsin  
Madison, USA  
bairdb@wisc.edu

Olivia Gosseries  
Dept. of GIGA-Research & Neurology  
Hospital of Liege  
Liege, Belgium  
ogosseries@ulg.ac.be

Jaakko O. Nieminen  
Dept. of Neuroscience and Biomedical Engineering  
Aalto University School of Science  
Espoo, Finland  
jaakko.nieminen@aalto.fi

Ji-Hoon Jeong  
Dept. of Brain and Cognitive Engineering  
Korea University  
Seoul, Korea  
jh_jeong@korea.ac.kr

Giulio Tononi  
Dept. of Psychiatry  
University of Wisconsin  
Madison, USA  
gtononi@wisc.edu

Seong-Whan Lee  
Dept. of Brain and Cognitive Engineering  
Korea University  
Seoul, Korea  
sw.lee@korea.ac.kr

Abstract—The aim of this research is to classify the level of consciousness within non-rapid eye movement sleep from electroencephalography data. We defined three categories of conscious experience based on subjective reports: dream experience, dream experience without recall of content, and no conscious experience. The filter-bank common spatial pattern and shrinkage linear discriminant analysis were used to classify sleep consciousness. We also evaluated conscious experience using 10-fold cross-validation. Classification accuracy was 88.8% for the three classes. In particular, the sensitivity for no conscious experience was 0.99, and the sensitivity for dream experience with and without recall of content was 0.86 and 0.73, respectively. These results suggest that it is easy to predict the brain dynamics between conscious and no conscious experience, but to distinguish dream experiences with and without the recall of content is difficult. These findings could help predict the levels of consciousness during sleep and may help in developing a passive biomarker of the levels of consciousness in humans.

Keywords—consciousness, sleep, classification, transcranial magnetic stimulation, electroencephalography, filter-bank common spatial pattern, shrinkage linear discriminant analysis

I. INTRODUCTION

Sleep has a profound impact on our health. Although its exact function remains to be determined, sleep is closely associated with brain [1]. Therefore, monitoring sleep stages has been very important in medical practice and research [2]. Machine learning methods have been developed for automatic sleep classification [3, 4]. Many studies have focused on the classification based on sleep stage in accordance with American Academy of Sleep Medicine (AASM) [5]. Until now, dreaming has been considered to occur mostly in rapid eye movement (REM) sleep, and it might have seemed plausible to classify consciousness based on sleep stage. However, recent studies have reported that dreaming occurs also during non-rapid eye movement (NREM) sleep [6].

A serial-awakening paradigm can be used to study conscious experiences within the same sleep stage [7]. The classification of conscious experiences within the same stage is crucial because physiological variables that differ between stages could otherwise act as confounding factors. With this in mind, a recent study explored prediction of conscious experience specifically within NREM sleep using electroencephalography (EEG) data [8]. The predictive performance between individual dream experience and the absence of dream experiences was 86.2%. In particular, the mean accuracy of dreaming was 91.6%, whereas the mean accuracy of no experience was 80.7% during NREM sleep. This predictive algorithm used a real-time bispectral threshold of neural activity in posterior cortical regions during NREM sleep. However, the study did not consider dream experiences without a recall of content.

Transcranial magnetic stimulation with high-density EEG (TMS/hd-EEG) is a useful method to measure brain complexity in functional integration and causality [9]. Based on direct perturbation of the thalamocortical system, the perturbational complexity index (PCI) is one empirical marker that can distinguish between consciousness and unconsciousness [10, 11]. PCI indexes the spatiotemporal pattern of the EEG response triggered by TMS during pharmacological, physiological, and pathological loss of consciousness. However, this measure has not yet been used to discriminate consciousness within the same vigilance states.

In this study, we classified the levels of consciousness based on machine learning and TMS/hd-EEG. We used the filter-bank common spatial filter and shrinkage linear discriminant analysis. To predict conscious experience with or without recall of content within NREM sleep, subjects underwent a serial-awakening paradigm. This study suggests a new direction for the sleep classification based on the conscious experience itself.

This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (No. 2017-0-00451). Development of BCI based Brain and Cognitive Computing Technology for Recognizing User’s Intentions using Deep Learning) and the Academy of Finland.
II. METHODS

A. Data Description and Preprocessing

The TMS/hd-EEG data are published in Ref. [12]. In the experiment, six healthy subjects (5 males, age 23.7 ± 3.2 years) performed a serial-awakening paradigm (four to five nights per subject). The EEG data were scored in accordance with the AASM criteria for sleep stage [13]. When NREM sleep had lasted over 3 minutes, TMS was applied to the medial superior parietal cortex approximately every 2 seconds. After the stimulation sequence ended, participants were awakened and asked whether before the awakening they were having dream experience (DE), dream experience without the recall of content (DEWR), or no conscious experience (NCE) (Fig. 1). Table I shows the individual number of awakenings of each class.

The signals were measured using a 60-channel TMS-compatible EEG amplifier (Nexstim eXimia, Nexstim Plc) with a 1450-Hz sampling rate. EEG analysis was performed using MATLAB 2011b and BCILAB toolbox [14]. Artifacts induced by TMS were removed by linearly interpolating the first 15 ms of the data post-TMS. The data were down-sampled to 362.5 Hz and filtered from 1.5 to 50 Hz. After rejecting bad trials, we used the trials occurring within 30 s before the awakenings. The EEG signals were baseline-corrected from ~400 to 0 ms and average-referenced.

B. Feature Extraction

To extract features, we used the filter bank common spatial pattern (FBCSP) algorithm. This algorithm is an extension of the common spatial pattern (CSP) algorithm, which is commonly used to optimize the discriminability of EEG signals. The filters are learned from a training dataset, and then they exploit the EEG signals to an informative low-dimensional spatial-subspace so that the variance of the time series is suitable for discrimination. The log-variance features from those projections are constructed [15]. CSP filter maximizes the variance of the EEG signals of one condition and minimizes that of the other condition.

The FBCSP algorithm finds the optimal spatio-spectral filter by constructing a filter bank [16]. The first step is spectral filtering. This algorithm divides the broadband frequency range into predefined small frequency bands. We used five frequency bands: delta (1.5–4 Hz), theta (4–8 Hz), alpha (8–13 Hz), beta (13–30 Hz), and gamma (30–45 Hz) band. The second step is spatial filtering. The EEG data in each frequency band are spatially filtered according to the CSP algorithm and log-variance feature extraction. The third step is feature extraction and selection [17]. The optimal filter is selected using maximal mutual information criterion [18]. As a result, this algorithm is very useful for exploring the subject-specific frequency band based on CSP. The features were extracted from 1-s epochs of non-averaged EEG signals post-TMS. Figure 2 illustrates the FBCSP algorithm.

C. Classification

We classified three classes based on conscious experience: DE, DEWR, and NCE. Since the FBCSP algorithm is optimized for binary-class classification, we, applied the pair-wise (PW) approach, which distinguishes every pair of classes and conjoins all comparisons as a popular multi-class classification method [19, 20].

<table>
<thead>
<tr>
<th>Subject</th>
<th>Classes</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DE</td>
<td>DEWR</td>
<td>NCE</td>
</tr>
<tr>
<td>Sub1</td>
<td>160</td>
<td>30</td>
<td>96</td>
</tr>
<tr>
<td>Sub2</td>
<td>45</td>
<td>185</td>
<td>243</td>
</tr>
<tr>
<td>Sub3</td>
<td>65</td>
<td>25</td>
<td>175</td>
</tr>
<tr>
<td>Sub4</td>
<td>114</td>
<td>19</td>
<td>163</td>
</tr>
<tr>
<td>Sub5</td>
<td>214</td>
<td>71</td>
<td>104</td>
</tr>
<tr>
<td>Sub6</td>
<td>176</td>
<td>82</td>
<td>142</td>
</tr>
</tbody>
</table>
Linear discriminant analysis (LDA) is a common machine learning method when features are extracted [20]. The log-variance feature vector from EEG signals was calculated using FBCSP filters. A shrinkage estimator of the covariance matrix was used to solve a small training set [21]. Two LDAs were generated to classify each feature. LDA projected high-dimensional data into a one-dimensional space to classify the binary classes. The projections maximized the distance between the binary classes and minimized the variance within each class [22].

To discriminate the binary classes, a weight vector \( \mathbf{w}_i \) and an offset vector \( w_0 \) were obtained using distance \( D \). For every block \( i \) and every sampling point \( t \) were as follows:

\[
D^{(i)}_t = \mathbf{w}^T_t \cdot \mathbf{d}^{(i)}_t - w_0
\]

where \( T \) denotes the transpose of the vector and \( \mathbf{d}^{(i)}_t \) is a feature vector. A measured distance \( D^{(i)}_t > 0 \) and \( D^{(i)}_t < 0 \) mean that block \( i \) is classified as binary features [23].

D. Performance Evaluation

The classification performance for conscious experience was evaluated via 10-fold cross-validation for each subject. We calculated the average accuracy. The precision and sensitivity in each class based on the confusion matrix were also computed. The precision indicates the positive labels given by the classifier, whereas the sensitivity means the effectiveness of a classifier to identify positive labels [24].

III. RESULTS

We classified the conscious experience within NREM sleep using the FBCSP algorithm. The classification performance indicators such as precision, sensitivity, and accuracy were considered here. Table II indicates the individual performance for the levels of consciousness within NREM sleep from 10-fold cross-validation. The maximum classification accuracy ± standard deviation was 96.6 ± 4.6\% (Sub4) and the minimum classification accuracy was 78.9 ± 8.7\% (Sub5) for 10-fold cross-validation. The averaged classification accuracy was 88.8 ± 6.9\%.

The confusion matrix for conscious experience is presented in Figure 3. The performance for NCE was 98.9\%, but the true positive value for DE and DEWR was 86.4\% and 73.2\%, respectively. The sensitivity with NCE was 0.99, and the sensitivity with DE and DEWR was 0.86 and 0.73, respectively. The precision within NREM sleep was similar to sensitivity (Table III). These results show that discrimination between conscious experience and no conscious experience was clear, but dream experiences with and without recall of content were more difficult to distinguish from each other.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sub1</td>
<td>92.6 ± 4.5</td>
</tr>
<tr>
<td>Sub2</td>
<td>89.5 ± 12.7</td>
</tr>
<tr>
<td>Sub3</td>
<td>93.2 ± 3.9</td>
</tr>
<tr>
<td>Sub4</td>
<td>96.6 ± 4.6</td>
</tr>
<tr>
<td>Sub5</td>
<td>78.9 ± 8.7</td>
</tr>
<tr>
<td>Sub6</td>
<td>82.0 ± 7.5</td>
</tr>
<tr>
<td>Mean ± SD</td>
<td>88.8 ± 6.9</td>
</tr>
</tbody>
</table>

TABLE III. PERFORMANCE ACROSS THREE CLASSES

<table>
<thead>
<tr>
<th>Class</th>
<th>Precision</th>
<th>Sensitivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE</td>
<td>0.85</td>
<td>0.86</td>
</tr>
<tr>
<td>DEWR</td>
<td>0.75</td>
<td>0.73</td>
</tr>
<tr>
<td>NCE</td>
<td>0.99</td>
<td>0.99</td>
</tr>
</tbody>
</table>
In this paper, we classified the level of consciousness using the FBCSP algorithm. The predictive classification was 88.8% for three classes. In particular, although we included the dream experience without the recall of content, we had a 2.6 percentage points higher classification performance than a previous two-class study. Our classification performance was high for the presence or absence of the conscious experience, but lower for dream recall. Our study has the limitation of imbalance-class problem. However, this is the first attempt to classify conscious experience for DE, DEWR, and NCE based on machine learning.

We used the spatio-spectral pattern of EEG signals based on the FBCSP algorithm, whereas PCI uses the spatiotemporal pattern of EEG signals. The spectral information is an important factor for neural correlates of consciousness [25]. Delta waves are a hallmark of unconsciousness and are associated with bistability of cortical neurons [26], which prevents causal interactions [27]. On the other hand, beta and gamma oscillations are linked to neuronal firing [28] and may play a role in memory processing during sleep [29]. In this regard, the FBCSP algorithm, which uses spatio-spectral features, is potentially well-suited for classifying conscious experiences.

In conclusion, this study explored the three-class classification of consciousness within NREM sleep. Future research should consider attempting a multi-class sleep classification of conscious experience, including not only EEG patterns within the same stage but also other stages such as REM sleep. In addition, the use of spatio-spectral and temporal domains would be an interesting direction for high-performance classification of the levels of consciousness. These results are potentially useful for online dream detection during sleep and may be useful in developing measures for improved prognosis of patients with disorders of consciousness.

ACKNOWLEDGMENT

The authors thank M. Boly for useful discussions.

REFERENCES


Robust and Adaptive Vehicle Detection System Using Surveillance Videos

1st Yan-Lin Chou
Department of Computer Science
National Taipei University
New Taipei City, Taiwan
minipopy0314@gmail.com

2nd Daw-Tung Lin*
Department of Computer Science
National Taipei University
New Taipei City, Taiwan
dalton@mail.ntpu.edu.tw

Abstract—Many automobile repair shops and parking lots are often equipped with a sensing coil to detect when a vehicle enters. However, the single sensing coil often produces false alarms (e.g., when motorcycles pass over the sensor or vehicles reverse from or stop at the gates). To overcome these problems, surveillance camera equipment is utilized and installed near the entrance gate to detect the incoming vehicles. Given that image-processing technology is now more developed, we propose using a camera to perform the task of a sensor and improve the accuracy of vehicle detection. This system is empowered such that it can adapt to the different factors of an outdoor environment. The background subtraction method is utilized to extract foreground information of a moving object from the field of view of the camera. Optical flow is also applied to determine the direction of the incoming vehicle. A foreground energy-map analysis is conducted to improve the reliability of vehicle detection. Finally, a 20-day test of the system was conducted at an automobile repair shop. The experimental results show a highly promising detection performance with a hit rate as high as 95.58%. The camera system is designed to replace the sensing coils for reducing the hardware costs and increasing the detection efficiency.

Index Terms—computer vision, vehicle detection, background subtraction, optical flow, energy-map analysis

I. INTRODUCTION

With advancement in technology, the traditional traffic management systems have evolved into an intelligent automated traffic management systems, which are capable of detecting vehicles with a high accuracy in an all weather conditions. Sina et al. [1] extracted raw traffic video from CCTV footage and used it to identify a specific vehicle and count the total numbers of vehicles per unit time. Pan et al. [2] combined background subtraction and edge detection algorithms to detect a vehicle. Moreover, they proposed a method that can automatically generate an adaptive region of interest (ROI) based on the height of a vehicle computed from the observed width of the road. Guo et al. [3] presented a vehicle detection method that is not sensitive to the time of day and can therefore be effective at night. Miller et al. [4] designed a vehicle detection method based on hidden Markov models (HMMs). They applied a modified Viterbi algorithm to the HMM sequential estimation framework to initialize and track vehicles. Moreover, they used the AdaBoost method to identify the vehicles. Seenouvong et al. [5] utilized the background subtraction method to extract information of a moving object within an ROI of the road to detect a vehicle. Meany et al. [6] employed background subtraction and machine learning algorithms to perform automatic vehicle counting by using remote video cameras. Choudhury et al. [7] applied Haar-like features for vehicle detection. Mehboob et al. [8] analyzed the vehicle trajectories in the video frame and tracked the trajectories over time. Vehicle counting is conducted by estimating the trajectories and comparing the trajectories with a Hungarian tracker.

Recently, many entrance gates have been equipped with sensing coils. Sensors are time-consuming to install and often trigger a false alarm when a non-vehicle object passes over them. Given that image-processing technology is now more developed [9], we propose using a camera to perform the task of a sensor. This would increase the value of the surveillance equipment, reduce equipment expenditure, and improve the accuracy of vehicle detection. This system is developed such that it can adapt to the different factors of an outdoor environment. The background subtraction method is utilized to extract foreground information of a moving object from the field of view (FOV) of a camera. Optical flow is also applied to determine the direction of the incoming vehicle. A foreground energy-map analysis is conducted to improve the reliability of vehicle detection. The concept of applying energy maps is very novel in the field of computer vision. Yi et al. [10] suggested that a crowd-movement path is similar to that of fluid movement and used this concept to establish a normal energy map. A higher energy represents the presence of more people in the region, while a lower energy represents the presence of fewer objects such as obstacles. Based on these factors, we can establish various customized energy maps. Thus, we suggest that the concept of energy map can also be applied for the detection of traffic flow.

II. SYSTEM DESIGN AND CONSIDERATIONS

A. Vehicle Detection System Configuration

Vehicle detection system reliability is affected by the position, angle, and FOV of a camera. Therefore, an algorithm is required to adjust these parameters to increase the reliability.
of the system. In this work, a general type camera is mounted at a high position to detect incoming vehicles, as shown in Fig. 2.

B. Main Issues and Considerations

When a sensing is deployed for automatic vehicle detection, false alarms are commonly triggered when a vehicle is either moving backward or parking temporarily. When an image-processing algorithm is incorporated into a camera, some of the problems encountered by the induction coil in the camera can be eliminated. However, the following issues persist: sensitivity to lighting change, non-target-vehicle object identification (pedestrian or motorcycle), false alarms for vehicles moving backward, and false alarms for vehicles parking temporarily. These situations are listed in Fig. 3 and are addressed as follows:

- Operation in all weather conditions: Because the vehicle detection system is installed in an outdoor environment, the outdoor lighting can affect the results of image-processing algorithms. Therefore, we utilized an adaptive background subtraction method (described in Section III-A) to automatically adapt to the variable light conditions. Figure 4 shows the three main lighting conditions that affect detection when using the camera.

- Pedestrians and motorcycles: An entrance-gate is usually an open space. Therefore, not only target vehicles but also other moving objects such as pedestrians and motorcycles appear in the FOV of the mounted camera. We constructed an algorithm that accurately identifies and excludes these non-target objects.

- Wrong way and backward movement: An automobile repair shop entrance gate is an open space; therefore, there is no specific path that a vehicle must follow to enter into the shop. Due to these liberal constraints, there are two main vehicle movement patterns that must be accurately identified: (1) the vehicle leaving the entrance gate in a reverse direction (reverse event) and (2) the vehicle leaving the entrance gate in the wrong direction (wrong-way event).

- Temporary parking: Occasionally, an incoming vehicle must wait for guidance and instructions from the service personnel concerning how and when the vehicle should enter. Thus, an incoming vehicle might enter the FOV of the camera but not leave the FOV of the camera for a brief period of time because the vehicle might have temporarily parked, as depicted in Fig. 5. To resolve this problem, we compare the initial and final positions of the vehicle to identify a temporarily parked vehicle, as discussed in Section III-C.

- License plate missing in the snapshot: To perform license plate recognition, the system must capture a snapshot of the vehicle. However, if the front of the vehicle has passed through the FOV of the camera after the vehicle detection algorithm is triggered, the license plate cannot be recognized. In Section III-D, we propose a foreground energy-map concept and an associated method to ensure that the license plate is visible in the captured snapshot.

![Fig. 1. Actual automobile repair shop where the solution was tested in this study.](image1)

![Fig. 2. Video camera configuration for vehicles detection.](image2)

![Fig. 3. Situations and events that are considered in this study.](image3)

![Fig. 4. Different lighting conditions: (a) normal daylight; (b) strong daylight; (c) night.](image4)

![Fig. 5. Temporary parking problem.](image5)

![Fig. 6. System flow chart.](image6)
III. THE PROPOSED VEHICLE DETECTION SYSTEM

We constructed and integrated a series of image-processing algorithms that are reliable and can detect vehicles in a complex environment under various weather conditions. Figure 6 shows a flow chart of the proposed vehicle detection system. First, an image is captured using the camera and pre-processed. The motion direction configuration is pre-defined to detect a specific direction of an incoming vehicle. Second, moving objects are extracted with the Gaussian Mixture and Gamma (GMG) background model segmentation method. Furthermore, pedestrians, motorcycles, and other non-target objects are filtered out by examining the object's width and area. Optical-flow method is then applied to determine the direction of the moving object in order to exclude reverse and wrong-way events. The center point of each moving object is recorded to exclude events in which a vehicle is temporarily parked. Then, to capture a snapshot of the license plate accurately, we utilize the energy-map method to determine the image that provides the clearest view of the license plate. Finally, the system counts the number of vehicles entering.

A. Background Subtraction by Using the GMG Model

A fixed-background model is affected by changes in the lighting conditions and weather. Thus, we utilize the robust and adaptive GMG background subtraction algorithm [11]. The GMG model can adapt to changes in the lighting conditions and extract foreground information in all weather conditions. The GMG background model classifies the foreground and background of an image on the basis of Bayesian inference. As shown in Fig. 7(f), if the background obtained from the GMG model is stable, then no hole appears in the foreground. Then, a morphological filtering operation can be applied to remove noise and fill small gaps to obtain a complete foreground object. In the GMG model, the foreground information is promptly forgotten, thus allowing the model to adapt to environmental changes such as lighting conditions and object shadow. We also evaluated the Visual Background Extractor (VIBE), Mixture of Gaussian (MOG), and MOG2 methods and observed that the foreground integrity of the GMG method is higher than that of these algorithms, as illustrated in Fig. 7.

B. Optical Flow

In Section II-B, we mention that when reverse and wrong-way events occur, a false alarm is triggered by the vehicle detection system. Thus, we adopt optical flow to detect the direction of the vehicle. After obtaining the direction, we exclude reverse and wrong-way events. The system then calculates a tolerance range of approximately $45^\circ$ to determine if the direction of the vehicle is appropriate. Reverse and wrong-way events are excluded if the angles exceed the range. Figure 8 shows an example of calculation results.

C. Bounding Box and Center Allocation

In Section II-B, we introduce a situation of an automobile repair shop in which a vehicle often has to stop temporarily to await maintenance. To avoid repeated counting and false alarms after the vehicle restarts, we compare the center point of the vehicle when it stops and when it is moving. If the difference is sufficiently small, the system determines the vehicle to be the same one. The center-point formula is expressed as $G_{\text{new}}G_{\text{pre}} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$, where $G_{\text{new}} = (x_1, y_1)$ is the current center-point location, and $G_{\text{pre}} = (x_2, y_2)$ is the previous center-point location. If $G_{\text{new}}G_{\text{pre}}$ is greater than a certain threshold, then the vehicle is identified as a new vehicle; otherwise, the vehicle is identified as the same one, as shown in Fig. 9.

Figure 10 shows an image sequence of a vehicle entering. In frame #1, the vehicle detection system sets a default point $G_{\text{pre}}$, which is placed far away from the vehicle entry position. In frame #2, if the foreground area of the vehicle is larger than the threshold, the system initiates the calculation of the vectors center position $G_{\text{new}}$. If $G_{\text{new}}G_{\text{pre}}$ is greater than a certain threshold, then the vehicle is considered as a new vehicle. However, if the vehicle is moving forward as appears in frame #10 and $G_{\text{new}}G_{\text{pre}}$ is smaller than a certain threshold, then the vehicle is considered the same vehicle, temporarily parked in the FOV of the camera. In Image frame #13, if the foreground area of the vehicle is smaller than a certain threshold, $G_{\text{new}}$ is replaced by $G_{\text{pre}}$. For example, if a vehicle halts in the middle of the FOV of the camera and the vehicles center position $G_{\text{new}}$ is replaced by $G_{\text{pre}}$, then $G_{\text{pre}}$ is in the middle of the FOV of the camera. If the vehicle restarts and $G_{\text{pre}}$ is smaller than the threshold, then this event is determined as a temporary parking event.

D. Foreground Energy-Map Analysis

Once the vehicle image has been analyzed through the aforementioned procedure, we capture a snapshot of the incoming vehicle for license plate recognition. To ensure that an appropriate image is selected from a series of snapshots (e.g., Fig. 11(a) and (d)), we proposed the foreground energy-map analysis method and integrated the method into the vehicle detection system. The energy map represents the probability of a specific event occurring in a region and displays the energy levels in pseudo color. Yi et al. [10] applied this concept to demonstrate the probability that a pedestrian pass through a specific region in a station hall. Subsequently, a scoring factor is calculated on the basis of the vehicle foreground area on the energy map. Finally, the snapshot with the largest score is chosen. Algorithm 1 describes the proposed foreground energy-map analysis procedure.

First, we build an energy map based on the probability that vehicles appear in a region. The region is colored in red if the probability is high. Otherwise, the region is colored in cool colors such as yellow and blue. In the scenario considered here, the best timing to capture a snapshot is when the vehicle is entering the automobile repair shop (i.e., direction from top to bottom of the FOV of the camera). Thus, as shown in Fig. 11(c) and (f), the ROI is plotted in red at the top and in blue at the bottom of the energy map. The energy map is divided into $n \times n = m$ blocks. Each block is specified by a weighting coefficient $w(i), 1 \leq i \leq m$. Red blocks indicate the
possible vehicle entry region and thus have high weights. The yellow blocks represent the region close to the edge of the image and have lower weights. The blue blocks around the lower edge of the energy map are given the lowest weights because a vehicle is not expected to enter from this location.

Once the energy map is completed, the foreground image can be used as an input for the energy-map analysis. For example, both foreground images in Fig. 11(b) and (e) were used to perform the energy-map analysis to determine the image that has the highest chance of containing the complete license plate. Let $Q_i$ be the sum of the foreground pixels gray level in each cell block $i$; then, the block flag $q_i$ is set as follows.

$$q_i = \begin{cases} 
1 & \text{if } Q_i \geq E \\
0 & \text{if } Q_i < E 
\end{cases} \quad (1)$$

**Algorithm 1** Foreground Energy Map Analysis

1. Build the energy map based on the probability of vehicles appear in the regions.
2. Divide the energy map into $n \times n = m$ blocks and set weighting coefficient $w(i), 1 \leq i \leq m$.
3. Compute $Q_i$ (the sum of foreground pixels grey level in each block $i$) and set the block flag $q_i$.
4. Calculate $W = \sum_{i=1}^{m} q_i \times w_i$ for each block $i$.
5. **if** $W$ is high **then**
6. keep the snapshot image.
7. **end if**

**IV. EXPERIMENTAL RESULTS**

**A. Dataset**

To evaluate the deployed system, we collected a general type surveillance camera video from an automobile repair shop for a period of 20 days. Based on the timeline, we divided the test video into four time slots: (a) 7:00 a.m.~11:00 a.m., with soft daylight; (b) 11:00 a.m.~2:00 p.m., with stronger daylight; (c) 2:00 p.m.~7:00 p.m., with dim daylight; and (d) 7:00 p.m.~10:00 p.m., at night. Figure 12 presents the time schedule and environmental conditions of the recorded videos. We found that weather conditions varied in the 20-day video data, including sunny, cloudy, and rainy days. As shown in...
Fig. 11. (a) An image contains a license plate at time $t$; (b) foreground image; (c) the energy-map score $W=2140$; (d) when the detection proceeds at time $t+a$, the vehicle deviating from the FOV of the camera; (e) foreground of (d); (f) the energy-map score $W=290$.

Fig. 12. Time schedule and weather conditions of recorded video data.

Fig. 12, two vertical black lines denote the business hours of the automobile repair shop - from 7:00 a.m. to 19:00 p.m. Finally, the ground truth is manually labeled. Therefore, we divided data into two scenarios, one with and one without night records, and tested the reliability of the system. Scenario 1 was from 7:00 a.m. to 7:00 p.m. (business hours of the automobile repair shop), and scenario 2 was from 7:00 a.m. to 10:00 p.m. (before the system was shut down).

B. Analysis of the Experimental Results

The proposed vehicle detection system was implemented on a personal computer with an Intel i5-760 quad-core CPU, 8 GB RAM, and 128 GB SSD. The performance was evaluated using the entire video dataset. Based on the two scenarios specified in Section IV-A, the test results are presented in Table I. For scenario 1, 1360 moving events were detected, with 1289 events being appropriate vehicle detections. The successful hit rate was approximately 95.58%. Sixty false alarm events occurred in total, representing a false alarm rate of approximately 4.41%. 60 vehicles were not detected in scenario 1; we classified them as missed detection events. The total missed detection rate was approximately 4.41%. We also noticed that the successful hit rate was lower in scenario 2 (93.60%). We observed a lower successful hit rate at night.

In addition, we reorganized the data for each time slot and present the detailed test results in Table II. During the business hours, the system had a successful hit rate of over 95% for the time slot from 7:00 a.m. to 5:00 p.m. Due to the lack of a proper light source in the time slot of 7:0022:00 p.m., the hit rate decreased to 74.45%. The results demonstrate that the performance of the detection system is highly affected by light. Pedestrians generated a total of 39 false alarms, with a false alarm rate of 2.86%. Motorcycles generated a single false alarm, with a rate of 0.07%. Temporarily parked vehicles generated a total of 20 false alarms, with a rate of 1.46%. We conclude that pedestrians cause the highest number of false alarms.

C. Performance Comparison

Our algorithm is implemented in all weather conditions and complex environments. However, most of the other algorithms are specifically designed for a highway and normal road environment, where false alarms do not occur. Five vehicle detection methods were compared in this study. Zheng et al. [12] used convolutional neural network (CNN) to construct a virtual coil and make the counting decisions. Yuan et al. [13] proposed neighborhood gradient prediction for extracting a feature and applying this feature to detect a vehicle. Tourani et al. [14] utilized edge detection, motion detection, and a Kalman filter to detect a vehicle. Song et al. [15] proposed a vehicle counting method that applies background subtraction and considers the ROI. As shown in Table III, our algorithm exhibits promising performance in a complex environment and various lighting conditions.

V. CONCLUSION

This paper presents a complete solution for outdoor vehicle detection in an open area for an entrance gate of an automobile repair shop. The background subtraction method is utilized to extract foreground information of a moving object. Optical flow is then applied to determine the direction of an incoming vehicle. A foreground energy-map analysis is proposed to improve the reliability of the snapshot captured during vehicle detection. The proposed system was implemented and tested for 20 days at an automobile repair shop in all weather conditions and complex environments. The experimental results show a highly promising detection performance with a hit rate as high

TABLE I

<table>
<thead>
<tr>
<th>Scenario 1</th>
<th>Scenario 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Ground Truth</td>
<td>1360</td>
</tr>
<tr>
<td># of Detected Object</td>
<td>1300</td>
</tr>
<tr>
<td>Hit</td>
<td>1289</td>
</tr>
<tr>
<td>Snapshot containing License Plate</td>
<td>60</td>
</tr>
<tr>
<td>Miss</td>
<td>60</td>
</tr>
<tr>
<td>False-alarm</td>
<td>95.58%</td>
</tr>
<tr>
<td>Hit Rate(%)</td>
<td>99.16%</td>
</tr>
<tr>
<td>Snapshot Hit Rate(%)</td>
<td>4.41%</td>
</tr>
<tr>
<td>Miss Rate(%)</td>
<td>4.41%</td>
</tr>
<tr>
<td>False Alarm Rate(%)</td>
<td>4.41%</td>
</tr>
</tbody>
</table>
as 95.58%. We conclude that the proposed computer vision system successfully excludes many specific unwanted events and has reliable vehicle detection performance. It can replace sensors, thus improving its acceptance by industry. In our energy-map analysis, the energy map is generated manually. In the future, the energy map could be automatically generated using deep learning algorithms, increasing the adaptability of the system.

REFERENCES


Animation Generation with a Low-Dimensional Simplicial Complex

1st Brian Gauch
EECS
Vanderbilt University
Nashville, TN
brian.gauch@vanderbilt.edu

2nd Alan Peters
EECS
Vanderbilt University
Nashville, TN
alan.peters@vanderbilt.edu

Abstract—Given motion capture training data, we control an animated character by reducing the dimensionality of the pose space, finding a simplicial complex using Delaunay triangulation, creating a motion graph using the simplicial complex, and then using the motion graph and simplicial complex for distance estimates and interpolation respectively. Using a simple pathfinding algorithm, we compare the above model to a simpler model where k-NN is used to induce a motion graph and interpolate.

Index Terms—animation, motion graph, manifold, Delaunay triangulation, simplicial complex

I. INTRODUCTION

Realistic animation of human motion is of crucial importance to the film and video game industries, and has applications elsewhere such as robotics and biomechanics. Animation of human motion is particularly challenging because the models have a fairly large number of joints and because viewers can easily detect unrealistic motions. Due to these difficulties, this area has received much research attention within Computer Graphics [1] [2].

When creating animations, a large portion of artist effort is put into so-called “inbetweening”, where start and end poses are known and the frames in between need to be generated. This has been an important area of Computer Graphics for some time [3]. At its most basic, inbetweening can be done with a linear interpolation, but these kinematic models do not lead to very realistic motions. More recently, animators incorporate a dynamics model that may be learned from motion capture [4].

Our goal is to generate realistic human motion animations given a small number of artist-generated or motion captured training motions. To do this we synthesize the intermediate skeletal positions by blending neighboring motions. Unlike the distance metric approaches used by some algorithms to find neighboring points in the state space, e.g. k-NN, we consider two poses to be neighbors if they are both vertices of the same simplex of the Delaunay triangulation.

II. RELATED WORK

A. Dimensionality Reduction

1) Dimensionality Reduction Motivation: There has been significant prior work in modeling and generating motions, on which our work builds. Many motion modeling algorithms’

 speeds depend not only on the number of data points n, but also on the dimensionality of the state space. Examples of areas in which dimensionality is critical include clustering, modeling and controlling a dynamical system, and machine learning. While adding dimensions to the input data ideally provides useful information, the reduction in speed associated with higher dimensionality is typically referred to as the “curse of dimensionality”. This creates a temptation to hand pick the input features, limiting them to those that seem likely to be important in some sense. Dimensionality reduction techniques aim to eliminate the need for hand-picking features, by pre-processing input data and selecting those dimensions which are deemed important programmatically. These new dimensions need not be (and indeed, rarely are) a subset of the original dimensions. Classical techniques like Principle Component Analysis limit new features to be linear combinations of input features, but many newer techniques do not.

2) Dimensionality Reduction Methods: Perhaps the best known dimensionality reduction technique is Principle Component Analysis [5] (PCA), because it was among the first dimensionality reduction methods, and is still one of the fastest. While naïve PCA is \(O(n^3 + D^3)\), it may be as fast as \(O(n d D)\) where \(D\) is the dimensionality of the data space and \(d\) is the dimensionality of the latent space [6]. Most other common dimensionality reduction methods like Multi-Dimensional Scaling [7] (MDS), ISOMAP [8], local linear embedding [9] (LLE), and Laplacian eigenmaps [10] are \(O(n^3)\) because they involve an eigendecomposition of an \(n\)-by-\(n\) matrix [11] [12]. Other methods include local tangent space alignment [13], Gaussian Process Latent Variable Models (GPLVM) [14], and Gaussian Process Dynamical Models (GPDM) [15].

B. Motion Models

Regardless of how we reduce dimensionality, the key to animation generation is the motion model. In general we may wish to allow exaggerated or cartoon-like physics [16]. Some human motion tracking algorithms compute a maximum likelihood estimator (MLE) of the next pose given a sequence of prior poses. Because our work focuses on animation generation, we are interested in the probabilities of pose sequences. Some animation generation algorithms build motion models from a discrete graphical representation of
training data. Arikan et al. [17] give a good introduction to motion graph models and how to generate animations from them. The model is then augmented by adding edges to the motion graph, \( G \), interpolating between nodes to create new nodes, or both. Ren et al. [18] add edges to \( G \) to minimize the lengths of shortest paths, and ask the user to verify the quality of resulting animations periodically. Zhao et al. [19] interpolate between whole sequences at a time, to produce a set of candidate nodes and edges. They then use a threshold based on dynamical probability to determine which to add to \( G \). Levine et al. [20] also add nodes and edges to \( G \). They simulate many user inputs when creating their controller, and create new nodes in regions that are frequently traveled. They add edges such that each node is connected to its k-NN in the latent space.

1) Motion interpolation and generation: Motion interpolation can be done while each training sequence is considered separately, by e.g., finding a smooth function that passes through each of the training data points. Motion blending means interpolating between two or more training sequences, e.g., creating a sequence that is a piecewise linear combination of two training sequences. Shin et al. [21] do motion blending in the latent spaces derived by various dimensionality reduction methods.

a) : Motion interpolation and motion blending are both relevant to animation in the film industry, but in the video game industry there is more interest in animations as a function of pose and user control signal. Such a policy is called a character controller. A controller should cause the character to respond quickly to a change of control signal. However, searching the motion graph for candidate responses to the control signal [17] and evaluating both the motion quality and control quality [22] of each candidate motion is generally too slow. Therefore, significant precomputation is usually needed to achieve the desired runtime speed, so that at runtime the controller does little more than a table lookup. Note that the size of the space of control signals must be small for the creation of such a table to be feasible. Typically the control signal is one direction (e.g., walking direction) or the location of the “end effector”, i.e. one joint of interest, e.g., a hand. Control of an end effector is the well-studied problem of inverse kinematics, used in robotics. Agrawal and van de Panne [23] presented a method to get foot positions from higher-level task-specific plans, then use those foot positions to generate a complete animation.

b) : There is a tradeoff in the creation of a motion graph for an animation or character controller between motion quality and control quality [22]. If control quality is to be emphasized, as Mccann et al. suggest for the case of video game character control [22], then many edges should be added to the motion graph. If motion quality is to be emphasized, then the motion graph should only have a few, high-quality edges beyond the edges in the training data. The tradeoff typically takes the form of a threshold for edge motion quality.

c) : Given a motion graph \( G \), creating a controller can be relatively simple. A separate Markov Decision Process (MDP) is applied to \( G \) for every possible control signal, or for a dense sampling of possible control signals if there is a distance metric in the control space. Mccann et al. [22] compare a MDP controller to other simpler, faster controllers.

d) : Levine et al. [20] also use an MDP-based controller. After the expected rewards of the MDP have finished propagating on the finite \( G \), Their algorithm interpolates reward between nodes, allowing their controller to go to points in the latent space which are not on \( G \).

C. Manifold reconstruction

1) n-Dimensional Delaunay Triangulation: In the motion models above, candidate edges for \( G \) are typically found using a distance metric, whether in the original space or in the latent space. Similarly, neighbors for interpolation can be found via k-NN. We instead use a simplicial complex [24], where each simplex can be thought of as a neighborhood. We find a simplicial complex using Delaunay triangulation.

Delaunay triangulations were originally defined in [25]. We first consider the Delaunay triangulation of a set of points in a two-dimensional space. We are interested in the following properties:

1) Given a set of two-dimensional points, a Delaunay triangulation is a set of triangles that covers the region in the plane bounded by the convex hull of the points.
2) The set of triangle vertices and the set of input points are equal.
3) Among the coverings that satisfy the two above conditions, a Delaunay triangulation is a covering that maximizes the minimum interior angle of its triangles.

This last property makes the Delaunay triangulation unique (i.e., it does not depend on the order of input points) under certain assumptions. Delaunay “triangulation” is a bit of a misnomer when it is generalized to higher dimensions, in which case a Delaunay triangulation produces a simplicial complex [26]. Variations of Delaunay triangulations have been used in Computer Graphics to generate triangular meshes [27] [28], from sets of points [29]. More recently, Delaunay triangulations have been used in manifold learning [30] [31].

III. APPROACH

Our approach to generating animations that start at an arbitrary point \( a \) in the pose space comprises the following steps.

a) Reduce the dimensionality of the pose space to produce a lower-dimensional latent space. Project point \( a \) from the latent space to point \( a' \) in the latent space.

b) Create a graph of the training data in the latent space using either k-NN or Delaunay triangulation.

c) Use the graph to define a neighborhood function over the latent space.

d) Find the vectors in the latent space which are projected directional derivatives of the training data.

e) Interpolate the latent vectors to define a vector field.

f) Find a path as follows: Sample the vector field at \( a' \), move with the same direction and magnitude as that vector, and
A. Dimensionality Reduction

We use PCA for dimensionality reduction because it is fast, and because it is straightforward to reproject back into to pose space where we eventually want to see our animations.

B. Motion Graph

We use a collection of training sequences to define our motion graph $G$. The first step is to add nodes and directed edges corresponding to each of the “clips” in our training data. We then extend this graph by finding the Delaunay triangulation of these nodes, or by adding edges using k-NN.

We have implemented n-dimensional Delaunay triangulation using the algorithm described by Watson [26]. One step of Watson’s algorithm (step B) is to find the circumspheres of new simplices; However Watson does not go into detail about how this done. We solved this subproblem using an algorithm described by Daly [32]. We generalize their “Center of Circumscribed Sphere” algorithm to $d$ dimensions. However, for the results presented in this paper, we always reduce dimension to 2, so a simpler implementation of Delaunay triangulation which only works for 2 dimensions would have sufficed.

Using k-NN, we add an undirected edge between points $p_1$ and $p_2$ if $p_2$ is one of $p_1$’s k nearest neighbors or vice versa. Using the Delaunay triangulation, there is an undirected edge between points $p_1$ and $p_2$ if there is at least one simplex of the Delaunay triangulation such that $p_1$ and $p_2$ are vertices of that simplex.

The next step is to direct the edges added by Delaunay or k-NN so the edge points in roughly the same direction as the training data. To do this, consider an undirected edge between training data points $p_1$ and $p_2$, with associated velocities $v_1$ and $v_2$. Let $\mathbf{v}_\text{target} = \frac{(v_1 + v_2)}{2}$ represent a vector that is in the desired direction. Take the dot product of $\mathbf{v}_\text{target}$ with the vectors ($p_2-p_1$) and ($p_1-p_2$). Keep whichever edge has a positive dot product with $\mathbf{v}_\text{target}$. We call the resulting directed graphs $G'_{\text{Delaunay}}$ and $G'_{k-NN}$, or simply $G'$ when the method of adding edges need not be specified. Examples of $G'_{k-NN}$ and $G'_{\text{Delaunay}}$ can be seen in Figure 1 and Figure 2 respectively.

Ideally, we would like to be able to find a transition from any animation frame in the training data to any other. Since such a transition amounts to a path on $G'$, all pairwise graphical distances need to be calculated, which requires that $G'$ be a single connected component.

If $G'$ instead has undirected edges, The graph $G'_{\text{Delaunay}}$ induced by the Delaunay triangulation would be a single connected component, because the simplices of a Delaunay triangulation cover the convex hull of the training data. $G'_{\text{Delaunay}}$ has $O(n \cdot d)$ edges where $d$ is the dimensionality of the latent space. The graph $G'_{k-NN}$ induced by k-NN is not necessarily a single connected component for $k \leq \frac{n-1}{2}$. To see this, consider the case where our data is divided into two distinct clusters, where the distance between the clusters greater than the distance between any two members of the same cluster. Some algorithms using k-NN get around this issue by adding edges to connect the graph [33].

If $G'$ has directed edges then it has a single component only if its undirected version is likewise one component. For the directed graph to be connected, we also need the data to be cyclical. More precisely, we need the following: Consider the set of time series $T_1, T_2, ... , T_r$ that make up $X$. Consider the directed bipartite graph $H$ such that for each $T_i$ there is a node in the left set of $H$ corresponding to the first element of $T_i$ and a node in the right set of $H$ corresponding to the last element of $T_i$. Let there be an edge from $a$ to $b$ in $H$ iff there is a path from $a$ to $b$ in $G'$. Note that this implies an edge from the beginning of each $T_i$ to its end. Then we claim that $G'$ has a single connected component iff $H$ has a single connected component.

Proof: Suppose $H$ is connected (i.e., has a single connected component) and $G'$ has at least two connected components. Then there is some pair of nodes $a, b \in G'$ such that there is no path from $a$ to $b$. Let $a \in T_a, b \in T_b$. Because $G'$ includes a path graph for each $T_i$, $\forall t \in T_i$ there is a path in $G'$ from $t$ to the $t_F$, the last element of $T_i$. $\forall t \in T_i$ there is a path in $G'$ from $t_0$, the first element of $T_i$, to $t$. Therefore, there is a
path in $G'$ from $a$ to the $a_F$, last element of $T_a$, and there is a path in $G'$ from $b_0$, the first element of $T_b$, to $t$. Because $H$ is connected, there is a path in $H$ from $a_F$ to $b_0$. Because the edges of $H$ are a subset of the transitive edges of $G'$, there is a path between nodes in $H$ only if there is a path between those nodes in $G'$. Therefore, there is a path in $G'$ from $a_F$ to $b_0$. Finally, there is a path from $a$ to $b$: $a \rightarrow a_F \rightarrow b_0 \rightarrow b$. This is a contradiction to our assumption that $G'$ has at least two connected components. Therefore, $H$ connected implies $G'$ is also connected.

To make the data more cyclical, we remove the root coordinates from the motion capture data, so that, e.g., a walk cycle stays in place. That is, we expect $H$ to have more edges if each time series ends close to where it starts. However, removing the root coordinates does not guarantee that $H$ is connected.

C. Neighborhoods

After constructing $G'$, the next step is to use it to define a function which, given a latent point, returns a set of neighbors which are elements of $G'$. Within each neighborhood we treat a point as a weighted combination of its neighbors.

For $G_{k-NN}'$, the neighborhood of a point is the k-NN of that point. For $G_{Delaunay}'$, the neighborhood of a point is the set of vertices of its containing simplex. In the case of $G_{Delaunay}'$, the neighborhoods returned by the neighborhood function form a topology. Specifically, the simplicial complex returned by Delaunay triangulation is the “geometric realization” [34] of an abstract simplicial complex, which has a topology. oNote that for $G_{Delaunay}'$, this function is only defined over the convex hull of the training data. To calculate weights within a neighborhood, we use inverse distance weighting. Specifically, we calculate the Euclidean distances from an arbitrary point $p$ to the $N$ points in the neighborhood, and the weight for each neighbor is given by $1/D^2$ normalized such that weights sum to 1.

If we use $G_{Delaunay}'$, our neighborhood function has discontinuities along the boundaries between simplices. If we use $G_{k-NN}'$, our neighborhood function has discontinuities along the boundaries between the cells of the corresponding $k_{th}$-order Voronoi diagram [35]. In general, we expect the neighborhood function induced by $G_{k-NN}'$ to have more discontinuities, but we expect the discontinuities in the neighborhood function induced by $G_{Delaunay}'$ to be more pronounced, because with fewer neighbors, the weight of the neighbor being dropped in the distance-weighted interpolation is usually higher.

D. Training Data Vectors

The next step is to use the training data to generate a vector field so new motions can be estimated and generated. We start by associating a vector $v_{x_i}$ with each training data point $x_i$ in a time series $X$, which is a discrete approximation of the derivative of $X$ at that point. We use the following: $v_{x_i} = \frac{x_{i+1} - x_i}{2}$ where $x_{i+1}$ exists and $v_{x_i} = \frac{x_i - x_{i-1}}{2}$ where it does not (at the end of a time series).
walking animation reprojected back into the original data space, where animations are much prettier.

**IV. Evaluation**

In order to compare the abilities of the $G'_{k-NN}$ and $G'_{Delaunay}$ models at finding new, plausible animations, we perform a cross-fold validation of sorts.

**A. Data sets and Metrics**

We use data from CMU Graphics Lab Motion Capture Database, a collection of motion capture data. In particular, we collected 16 time series and broke them into four subsets, each containing four related time series. Two were labeled walking (w1 and w2), and two were labeled running (r1 and r2); w2 and r2 include turning behavior. More details about the sequences used appear in Table I.

We define two metrics to evaluate the similarity of a generated animation to a “true” animation, assuming that both animations share the same starting point. Error F is the distance from the generated animation to the final point of the true animation during the closest approach. Error A is the average distance from the true animation to the (closest point on the) generated animation, and vice versa, i.e., average distance from the generated animation to the (closest point on the) true animation. Error A is intuitively similar to the area between the two curves.

**B. Experimental design**

Our cross-fold validation of animations works as follows: For each of our four subsets, we train on all but one of the sequences (three sequences) to learn $G'_{k-NN}$ and $G'_{Delaunay}$ and the corresponding and vector fields. We then generate a path in the latent space by starting at the same point as the fold that was left out of training and following the vector field. On the rare occasion that our path takes us outside of the convex hull of the training data, and the vector field induced by Delaunay triangulation is therefore undefined, we use k-NN as a fallback. Finally, for each subset, for each method (Delaunay or k-NN), for each of the four folds, we measure Error A and Error F of the generated path where the left out fold is consider the true animation.

**C. Results**

Table I shows the results of our experiments. It reports the medians of both metrics from each of the 4-fold cross-validations. You can see that the $G'_{Delaunay}$ model gave better results than $G'_{k-NN}$ overall, with medians of 0.068 vs 0.079 (Error A) and 0.041 vs 0.042 (Error F) across all subsets. $G'_{Delaunay}$ also performed better than $G'_{k-NN}$ within five of the eight combinations of subset and metric.

**V. Discussion**

We present a new method of creating a motion graph from motion capture data, based on Delaunay triangulation. Our method first finds the Delaunay triangulation of the training data points, then defines a vector field over the convex hull.
TABLE II

<table>
<thead>
<tr>
<th></th>
<th>w1</th>
<th>w2</th>
<th>r1</th>
<th>r2</th>
<th>all data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error A Delaunay</td>
<td>0.023</td>
<td>0.073</td>
<td>0.749</td>
<td>0.068</td>
<td>0.068</td>
</tr>
<tr>
<td>k-NN (k=16)</td>
<td>0.022</td>
<td>0.105</td>
<td>0.099</td>
<td>0.118</td>
<td>0.079</td>
</tr>
<tr>
<td>Error F Delaunay</td>
<td>0.02</td>
<td>0.043</td>
<td>0.038</td>
<td>0.067</td>
<td>0.041</td>
</tr>
<tr>
<td>k-NN (k=16)</td>
<td>0.057</td>
<td>0.114</td>
<td>0.044</td>
<td>0.024</td>
<td>0.042</td>
</tr>
</tbody>
</table>

of the training data by interpolating among vertices of the containing simplex.

Finally, given a start point, it generates an animation by following the vector field. Our experiments showed that our approach outperformed k-NN, as measured by Error A and Error F.

A. Future Work

In future, we plan to extend our Delaunay triangulation method to higher dimensions, and to investigate better pathfinding methods. In particular, we are interested in the use of all-pairs shortest paths to find paths towards some target point, such as the final point of a fold in cross-fold validation. We also plan to consider combining the k-NN and Delaunay models, perhaps by defining the neighborhood of k-Delaunay to be the union of the neighborhoods of Delaunay and k-NN.

ACKNOWLEDGMENT

The data used in this project was obtained from mocap.cs.cmu.edu. The database was created with funding from NSF EIA-0196217.

REFERENCES


Poster Sessions
Monday May 14, 2018, 2:30 PM
Wednesday May 16, 2018, 2:30 PM

Chunhua Feng  
The existence of oscillatory solutions for a complex-valued neural network model with delay  
Rajesh Loganathan, Bhooopathy K and Abdur Rahman Mohamed Ismail  
Distance Estimation Based Energy Efficient Protocol for Wireless Sensor Network  
Yibin Hou and Jin Wang  
LS- SVM's no-reference video quality assessment model considering the network packet loss under the internet of things  
Hamza Djebli and Samy Ait-Aoudia  
SAF-AFIS: SIFT-based Alignment Free Approach for an Automatic Fingerprint Identification System  
Dirk Meijer, Mitchell Kesteloo and Arno Knobbe  
Unsupervised Anomaly Detection in Sewer Images with a PCA-based Framework  
Suban Krishnamoorthy and Ching Suen  
Survey of Issues with Text to Speech Synthesis of Multilingual Indian Texts  
Tamarafinde Dittimi, Ching Suen and Ali Hmood  
Mobile Based Assistive Technologies for Authentication of Banknotes  
Yea-Shuan Huang and Charles Djimy Slot  
A Robust Palmprint Matching Method  
Elham Vahdati and Ching Y. Suen  
A novel female facial beauty predictor  
Saeed Khazaee, Maryam Sharifi Rad and Ching Y. Suen  
Restoring height-map images of shiny coins using spline approximation to detect counterfeit coins  
Hongjun Li and Wei Hu  
Automatic High-Speed Compressive Tracking with Motion Prediction  
Ning Bi, Haoli Sun and Tan Jun  
Off-line Handwritten Chinese Recognition and Improvement Based on Caffe Framework  
Xin Lin, Jiahao Chen, Jun Tan and Ning Bi  
Attention-Based Recurrent Neural Networks in Chinese Short Text Classification  
Yulong Wang, Yuan Yan Tang, Cuiming Zou, Lina Yang and Patrick Wang  
Quaternion Linear Regression for Color Face Recognition  
Clement Bouvier, Cédric Clouchoux, Nicolas Souedet, Anne-Sophie Hérard, Zhenzhen You, Caroline Jan, Philippe Hantraye, Gilles Mergoil, B. Helena Rodriguez and Thierry Delzescaux  
Computational Optimization for Fast and Robust Automatic Segmentation in Virtual Microscopy using Brute-force-based Feature Selection  
Sarada Dakua, Gajendra Mourya, D. Bhatia, Julien Abinahed and Abdulla Al-Ansari  
Novel Graph-based Image Segmentation: Application to Medical Imaging  
Lina Yang, Pu Wei, Lin Bai, Cheng Zhong, Yuan Yan Tang and Jianjia Pan  
Fractal-windowed based Empirical Mode Decomposition Scheme for Protein Sequence Analysis  
Donato Impedovo, Giuseppe Pirlo and Vincenzo Dentamaro  
A new ConvNet architecture for heartbeat classification  
Mahdi Barat Zadeh Joveini, Javad Sadri and Hoda Alavi Khoushhal  
Fractal Modeling of Big Data Networks  
Hossein Izadi and Javad Sadri  
Application of Pattern Recognition in Mineral Segmentation and Identification  
Woo-Hyuck Song, Hong-Gyu Jung and Seong-Whan Lee  
Sequential Transformation with Geometric Constraints for Matching Oblique Aerial Images  
Florian Particic, Christian Hofmann, Markus Hiller, Lucila Patino-Studencki and Jórrn Thielecke  
Stochastic Modeling of Camera Errors for Stereo Image Processing  
Hamza Qassoud, Miodrag Bolic and Sreeraman Rajan  
Posture and Fall Detection System Using 3D Motion Sensors  
Hongjun Li, Yupeng Ding, Chaobo Li and Ze Zhou
Fall Detection Based on Deep Saliency Images
Yuanyuan Shen, Xu-Yao Zhang and Cheng-Lin Liu

Online Semi-Supervised Learning with Adaptive Vector Quantization
Julio Valdes, Zachary Baird, Sreraman Rajan and Miodrag Bolic

William Brendel and Luis Marujo

Sequential Minimal Optimization Extended to General Quadratic Programming
Yantao Wei, Peng Zhang, Huang Yao, Jiazhen Xu and Xinge You

Stacked Kernel Extreme Learning Machine for Hyperspectral Image Classification
Jeong-Soo Kim, Geon-Tae Ahn and Soon-Yong Park

A Block-based Path Recognition of Slag Removal Using Convolutional Neural Network
Srisupang Thewsuwan and Keichii Horio

Local Binary Pattern Mapping on Graph-based Image Representation for Texture Classification
José A. Reyes-Ortiz, Maricela Bravo and Leonardo Sánchez

Ontology-driven Acquisition of Verbal and Nominalization Patterns for Criminal Events
Mireya Tovar Vidal, Gerardo Flores, Azucena Montes, Meliza Contreras González and Ana Patricia Cervantes

Classification of Keyphrases using Random Forest and Latent Semantic Analysis
Farnoush Zohourian, Jan Siegemund, Mirko Meuter and Josef Pauli

Efficient fine-grained road segmentation using superpixel-based CNN and CRF models
Hiroki Ogihara and Masayuki Mukunoki

Image Classification using Collaborative Mean Attraction with Sparse Optimization
Phutphalla Kong, Matei Mancas, Seng Kheang and Bernard Gosselin

Saliency and Object Detection
Justin Spencer, Deborah Lawrence, Kaushik Roy, Prosenjit Chatterjee, Albert Esterline and Jung-Hee Kim

Presentation Attack Detection using Convolutional Neural Networks and Local Binary Patterns
Zeinab Sedighi, Hossein Ebrahimpour-Komleh and Ayoub Bagheri

SDRN: Scalable Deep Rectifier Network for opinion spam detection
Vitaliy Tayanov, Adam Krzyzak and Ching Suen

Learning classifier predictions: is this advantageous?
Muneera Alsaedi, Thomas Fevens, Adam Krzyzak and Lukasz Jelen

Hybrid RUSBoost Versus Data Sampling to Address Data Imbalance for Breast Cancer Cytological Malignancy Grading
A. Raji

A generalized unified discrete linear method for edge detection by antisymmetric FIR kernels
Vinay Kaushik and Brejesh Lall

Fast Hierarchical depth map computation from stereo
Serguei Mokhov, Joey Paquet and Arash Khodadadi

Fast Context-Annotated Classification of Different Types of Web Service Descriptions
Zachary Baird, Julio Valdés, Sreraman Rajan and Miodrag Bolic

Classification of Human Activity Level Using Single Channel CW Doppler Radar
Temitope T Adeyemo, Adebola O Olowoye, Temilola M Adepoju, Elijah O Omidiora and Stephen O Olabiyisi

Comparative Study of a Shape-Based and a Texture-Based Feature Extraction Technique for Mass Classification in Digital Mammograms
Adebola O. Olowoye, Stephen O. Olabiyisi, Elijah O. Omidiora, Temitayo M. Fagbola and Temitope T. Adeyemo

Comparative Study of Iterative Back Projection and Discrete Algebraic Reconstruction Techniques for Reconstruction of Low Resolution Images

The existence of oscillatory solutions for a complex-valued neural network model with delay

Chunhua Feng
College of STEM
Alabama State University
Montgomery, USA
cfeng@alasu.edu

Abstract—In this paper, a complex-valued neural network model with delay is investigated. Some sufficient conditions to guarantee the existence of oscillatory solutions for the model are obtained. Computer simulation is provided to demonstrate the correctness of the proposed results.

Index Terms—complex-valued network, delay, limit cycle, oscillation

I. INTRODUCTION

In the past decades, the dynamic properties of complex-valued neural networks with delays have attracted great attention of many researchers. Because complex-valued neural networks can be used to solve some problems which cannot be solved with their real-valued counterparts. For example, by means of a complex-valued neural network one can solve the XOR problem and the detection of symmetry problem which cannot be solved with a single real-valued neuron [1, 2]. Thus, many properties of complex valued neural network models such as boundedness, stability, including exponential stability and global asymptotic stability of the equilibrium point, Hopf bifurcation have been studied [3-20]. Recently, Zhang et al. have considered a complex value delayed Hopfield network model which has a ring topology consists of two coupling unidirectional rings, each with four oscillators as follows [3]:

\[
\begin{align*}
\frac{dz_1(t)}{dt} &= (a_0 + ib_0)z_1(t) + (c_0 + id_0)f(z_2(t)) + (m_0 + in_0)f(w_1(t - \tau)), \\
\frac{dz_2(t)}{dt} &= (a_0 + ib_0)z_2(t) + (c_0 + id_0)f(z_3(t)) + (m_0 + in_0)f(w_2(t - \tau)), \\
\frac{dz_3(t)}{dt} &= (a_0 + ib_0)z_3(t) + (c_0 + id_0)f(z_4(t)) + (m_0 + in_0)f(w_3(t - \tau)), \\
\frac{dz_4(t)}{dt} &= (a_0 + ib_0)z_4(t) + (c_0 + id_0)f(z_1(t)) + (m_0 + in_0)f(z_2(t - \tau)), \\
\frac{dw_1(t)}{dt} &= (a_0 + ib_0)w_1(t) + (c_0 + id_0)f(w_3(t)) + (m_0 + in_0)f(z_3(t - \tau)), \\
\frac{dw_2(t)}{dt} &= (a_0 + ib_0)w_2(t) + (c_0 + id_0)f(w_4(t)) + (m_0 + in_0)f(z_4(t - \tau)), \\
\frac{dw_3(t)}{dt} &= (a_0 + ib_0)w_3(t) + (c_0 + id_0)f(w_1(t)) + (m_0 + in_0)f(z_2(t - \tau)), \\
\frac{dw_4(t)}{dt} &= (a_0 + ib_0)w_4(t) + (c_0 + id_0)f(w_2(t)) + (m_0 + in_0)f(z_3(t - \tau)),
\end{align*}
\]

(1)

where \(\tau \geq 0\) is the time delay and the coefficients \(a_0, b_0, c_0, d_0, m_0\), and \(n_0\) are all real values. Activation function \(f(z) = [\tanh(u) + \tanh(v)] + i[\tanh(u) + \tanh(v)](z = u + iv)\) is common to all neurons. By using the symmetric bifurcation theory of delay differential equations combined with representation theory of Lie groups, the existence of multiple branches of bifurcating periodic solution has been obtained. However, for the practical point of view, the coefficients \(a_0, b_0, c_0, d_0, m_0\), and \(n_0\) may be different for each neuron. Therefore, in this paper, we extend model (1) to the following:

\[
\begin{align*}
z_1'(t) &= (a_1 + ib_1)z_1(t) + (c_1 + id_1)f(z_2(t)) + (m_1 + in_1)f(w_1(t - \tau)), \\
z_2'(t) &= (a_2 + ib_2)z_2(t) + (c_2 + id_2)f(z_3(t)) + (m_2 + in_2)f(w_2(t - \tau)), \\
z_3'(t) &= (a_3 + ib_3)z_3(t) + (c_3 + id_3)f(z_4(t)) + (m_3 + in_3)f(w_3(t - \tau)), \\
z_4'(t) &= (a_4 + ib_4)z_4(t) + (c_4 + id_4)f(z_1(t)) + (m_4 + in_4)f(w_4(t - \tau)), \\
w_1'(t) &= (a_1 + ib_1)w_1(t) + (c_1 + id_1)f(w_2(t)) + (m_1 + in_1)f(z_2(t - \tau)), \\
w_2'(t) &= (a_2 + ib_2)w_2(t) + (c_2 + id_2)f(w_3(t)) + (m_2 + in_2)f(z_3(t - \tau)), \\
w_3'(t) &= (a_3 + ib_3)w_3(t) + (c_3 + id_3)f(w_4(t)) + (m_3 + in_3)f(z_4(t - \tau)), \\
w_4'(t) &= (a_4 + ib_4)w_4(t) + (c_4 + id_4)f(w_1(t)) + (m_4 + in_4)f(z_2(t - \tau))
\end{align*}
\]

(2)

where the coefficients \(a_i, b_i, c_i, d_i, m_i\), and \(n_i (i = 1, 2, 3, 4)\) are all real values. Let \(z_k(t) = u_k(t) + iv_k(t), w_k(t) = p_k(t) + iq_k(t) (k = 1, 2, 3, 4)\), by taking the real and imaginary parts from (2), we have

\[
\begin{align*}
u_1'(t) &= a_1u_1(t) - b_1v_1(t) + c_1[tanh(u_1(t) + tanh(v_1(t))] \\
&- d_1[tanh(u_1(t) + tanh(v_1(t))] + m_1[tanh(p_1(t) - \tau)] + tanh(q_1(t - \tau))] - n_1[tanh(p_1(t) - \tau)] + tanh(q_1(t - \tau))),
\end{align*}
\]

(3)

This shows that system (3) is equivalent to system (2). Clearly, system (3) is a real differential equation with a delay. Noting that \(v_1(t) = 0\). It is clear that \((u_1, v_1, p_1, q_1) = (0, 0, 0, 0) (i = 1, 2, 3, 4)\) is an equilibrium point of system (3). Since \(v_1(t) \bigg|_{\tau = 0} = 1 - tanh^2(0) = 1\), the linearization of system (3) at the origin leads to

\[
\begin{align*}
u_1'(t) &= a_1u_1(t) - b_1v_1(t) + c_1[u_1(t) + v_1(t)] - d_1[u_1(t) + v_1(t)] \\
&+ m_1[u_1(t) + v_1(t)] + q_1(t - \tau) - n_1[u_1(t) + v_1(t)] + q_1(t - \tau),
\end{align*}
\]

(4)
For convenience, let \( x_1(t) = u_1(t), x_{4+i}(t) = v_i(t), x_{12+i}(t) = q_i(t) \) (i = 1, 2, 3, 4), then system (4) can be written as the following:

\[
\begin{align*}
\begin{cases}
 x'_1(t) &= a_1 x_1(t) - b_2 x_2(t) + c_1 x_2(t) - d_1 x_1(t) - n_1 x_1(t - \tau) + n_2 x_2(t - \tau) + x_3(t - \tau) \\
 x'_2(t) &= a_2 x_2(t) - b_3 x_3(t) + c_2 x_3(t) - d_2 x_2(t) - n_2 x_2(t - \tau) + n_3 x_3(t - \tau) + x_4(t - \tau) \\
 x'_3(t) &= a_3 x_3(t) - b_4 x_4(t) + c_3 x_4(t) - d_3 x_3(t) - n_3 x_3(t - \tau) + n_4 x_4(t - \tau) + x_5(t - \tau) \\
 x'_4(t) &= a_4 x_4(t) - b_5 x_5(t) + c_4 x_5(t) - d_4 x_4(t) - n_4 x_4(t - \tau) + n_5 x_5(t - \tau) + x_6(t - \tau) \\
 x'_5(t) &= a_5 x_5(t) - b_6 x_6(t) + c_5 x_6(t) - d_5 x_5(t) - n_5 x_5(t - \tau) + n_6 x_6(t - \tau) + x_7(t - \tau) \\
\end{cases}
\end{align*}
\]

(5)

The trivial solution of system (5) is called \( \tanh(0) = 0 \) then system (4) can be written as the following:

\[
\begin{align*}
 X'(t) = AX(t) + BX(t - \tau)
\end{align*}
\]

(6)

where \( X(t) = (x_1(t), x_2(t), \ldots, x_{16}(t))^T \), \( X(t - \tau) = (x_1(t - \tau), x_2(t - \tau), \ldots, x_{16}(t - \tau))^T \), both \( A = (a_{ij})_{16 \times 16} \) and \( B = (b_{ij})_{16 \times 16} \) are 16 by 16 matrices.

In order to discuss the existence of permanent oscillation for system (3) we will adopt Chafee’s criterion [21]: For a class time delay system which has a unique unstable equilibrium point, and all solutions of the system are bounded, this particular instability of the unique equilibrium point and the boundedness of the solutions will force the system to generate a limit cycle, namely, a permanent oscillatory solution. System (3) is in keeping with the requirements of Chafee’s criterion we refer the reader to [22, appendix].

II. Preliminaries

In this paper we adopt the following norms of vectors and matrices [23]: \( \| x(t) \| = \max_{1 \leq j \leq 16} \sum_{i=1}^{16} |x_i(t)| \), \( \| A \| = \max_{1 \leq j \leq 16} \sum_{i=1}^{16} |a_{ij}| \), the measure \( \mu(A) \) of a matrix \( A \) is defined by \( \mu(A) = \lim_{\theta \to 0} \frac{1}{\theta} \| I + \theta A \| - 1 \), for which the chosen norms reduces to \( \mu(A) = \max_{1 \leq j \leq 16} \sum_{i=1}^{16} |a_{ij}| \).

Definition 1 The trivial solution of system (5) is called unstable if there exists at least one component is unstable.

Lemma 1 Assume that the matrix \( R = A + B \) is a nonsingular matrix, then system (5) (or (6)) has a unique equilibrium, implying that system (3) has a unique equilibrium.

Proof An equilibrium point \( x^* = [x_1^*, x_2^*, \ldots, x_{16}^*]^T \) of system (5) (or (6)) is a constant solution of the following algebraic equation

\[
Ax^* + Bx^* = (A + B)x^* = Rx^* = 0.
\]

(7)

Noting that \( R = A + B \) is a 16 by 16 matrix. According to the linear algebraic knowledge, if \( R \) is a singular matrix, Eq. (7) may have infinitely many solutions. However, if \( R \) is a nonsingular matrix, Eq. (7) has only one solution, namely, the trivial solution which implies that system (5) has a unique equilibrium. For system (3), the activation function \( \tanh(y) \) is a monotone function, if and only if \( \tanh(0) = 0 \). Therefore, system (5) has a unique equilibrium point suggests that system (3) has a unique equilibrium point, it is exactly the zero point.

Lemma 2 If parameters \( a_i (1 \leq i \leq 4) \) are negative numbers, then all solutions of system (3) are bounded.

Proof To prove the boundedness of the solutions in system (3), we construct a Lyapunov function \( v(t) = \sum_{i=1}^{4} \frac{1}{2} (a_i^2(t) + v_i^2(t) + p_i^2(t) + q_i^2(t)) \). Noting that \( a_i < 0 (1 \leq i \leq 4) \) and \( |\tanh(y)| \leq 1 \). Calculating the derivative of \( V(t) \) through system (3) we get

\[
V'(t)(3) = \sum_{i=1}^{4} \left[ u_i'(t)u_i(t) + v_i'(t)v_i(t) + p_i'(t)p_i(t) + q_i'(t)q_i(t) \right]
\]

\[
\leq \sum_{i=1}^{4} \left[ a_i [u_i^2(t) + v_i^2(t) + p_i^2(t) + q_i^2(t)] + 2(|c_i| + |d_i| + |m_i| + |n_i|)|u_i(t)| + |v_i(t)| + |p_i(t)| + |q_i(t)| \right]
\]

\[
= \sum_{i=1}^{4} \left[ a_i [u_i^2(t) + v_i^2(t) + p_i^2(t) + q_i^2(t)] + K(u_i(t) + v_i(t)) + |p_i(t)| + |q_i(t)| \right]
\]

where \( K = 2(|c_i| + |d_i| + |m_i| + |n_i|) \). Obviously, when \( u_i(t), v_i(t), p_i(t), q_i(t) (i = 1, 2, 3, 4) \) tend to infinity, \( u_i^2(t), v_i^2(t), p_i^2(t), q_i^2(t) \) are higher order infinity than \( K(u_i(t), v_i(t), k_i, q_i(t)) \), respectively.

Therefore, there exists suitably large \( L > 0 \) such that \( V'(t)(3) < 0 \) when \( |u_i| > L, |v_i| > L, |p_i| > L, |q_i| > L \) since \( a_i < 0 \). This means that all solutions of system (3) are bounded.

III. Existence of Oscillatory Solutions

Obviously, the instability of trivial solution of system (5) implies that the trivial solution of system (3) is unstable. Therefore, in the following we only consider the instability of trivial solution of system (5).

Theorem 1 Assume that system (5) has a unique equilibrium point for given parameters. If the following condition holds

\[
\| B \| \| \exp(-|\tau|\mu(A)) > 1. \]

(9)

Then the unique equilibrium point of system (5) is unstable, implying that system (3) generates a limit cycle, namely, a permanent oscillatory solution.

Proof We shall prove that the unique equilibrium point of system (5) which is exactly the zero point is unstable. From (5), when each \( x_i(t) > 0 \) we have

\[
\frac{d|X(t)|}{dt} = AX(t) + BX(t - \tau)
\]

(10)

when each \( x_i(t) < 0 \) one can obtain

\[
\frac{d|X(t)|}{dt} = A(-X(t)) + B(-X(t - \tau))
\]

(11)
Therefore we have
\[
\frac{d}{dt} \left( \sum_{i=1}^{16} |x_i(t)| \right) \leq \mu(A) \sum_{i=1}^{16} |x_i(t)| + \| B \| \sum_{i=1}^{16} |x_i(t - \tau)|
\] (12)

Specifically, for the scalar time delay differential equation
\[
\frac{dy(t)}{dt} = \mu(A)y(t) + \| B \| y(t - \tau)
\] (13)

if the unique equilibrium point of system (13) is stable, then the characteristic equation associated with (13) given by
\[
\lambda = \mu(A) + \| B \| e^{-\lambda \tau}
\] (14)

will have a real negative root say \(\lambda_0\), and we have from (14)
\[
|\lambda_0| \geq \| B \| e^{\| A \| \tau} - |\mu(A)|
\] (15)

Using the formula \(e^{\tau} \geq e x\) for \(x \geq 0\) one can get
\[
1 \geq \left( \frac{\| B \| e^{\| A \| \tau} + |\lambda_0|}{|\mu(A)| + |\lambda_0|} \right) = \left( \frac{\| B \| \tau e^{-\| A \| \tau} e^{\| A \| \tau} e^{\| A \| \tau}}{(\| A \| + |\lambda_0|)\tau} \right)
\]
\[
\geq (\| B \| e^{\tau} e^{-\| A \| \tau})
\] (16)

The last inequality contradicts the equation (9). Hence, our claim regarding the instability of the equilibrium point of system (13) is valid. Based on the comparison theorem of differential equation [24] we have \(\sum_{i=1}^{16} |x_i(t)| \leq y(t)\). Since the equilibrium point of system (13) is unstable, therefore, there exists at least one \(x_i(t), i \in \{1, 2, \cdots, 16\}\) is unstable. According to the definition 1, the instability of the component \(x_i(t)\) implies that the trivial solution of (5) is unstable, which implied the instability of the trivial solution of system (3).

Since all solutions of system (3) are bounded, the instability of the unique equilibrium point together with the boundedness of the solutions lead system (3) to generate a limit cycle, namely, a permanent oscillatory solution based on [21].

**Theorem 2** Assume that system (3) has a unique equilibrium point for given parameters. Let \(\alpha_1, \alpha_2, \cdots, \alpha_{16}\) represent the eigenvalues of matrix \(A\), and \(\beta_1, \beta_2, \cdots, \beta_{16}\) the eigenvalues of matrix \(B\). Assume that matrix \(A\) has a positive real eigenvalue, or there is at least one eigenvalue, say \(\alpha_j\) which has a positive real part \(Re(\alpha_j) > 0\), satisfying that \(|Re(\beta_j)| + |Im(\beta_j)| < Re(\alpha_j)\), then the unique equilibrium point of system (5) is unstable, which implies that system (3) generates a limit cycle.

**Proof** Consider system (5), the characteristic equation of system (5) is the follows:
\[
det(\lambda I - a_{ij} - b_{ij} e^{-\lambda \tau}) = 0
\] (17)

where \(I\) is an identity matrix. Since the eigenvalues of matrix \(A\) are \(\alpha_1, \alpha_2, \cdots, \alpha_{16}\), and the eigenvalues of matrix \(B\) are \(\beta_1, \beta_2, \cdots, \beta_{16}\), so equation (17) changes to the following
\[
\sum_{i=1}^{16} (\lambda - \alpha_i - \beta_i e^{-\lambda \tau}) = 0
\] (18)

We are led to an investigation of the nature of the roots of the equation:
\[
\lambda = \alpha_i + \beta_i e^{-\lambda \tau}, i = 1, 2, \cdots, 16.
\] (19)

Without loss of generality, assume that \(\alpha_1\) is a complex number which has a positive real part \(Re(\alpha_1)\), then we have
\[
\lambda = \alpha_1 + \beta_1 e^{-\lambda \tau}.
\] (20)

Assume that \(\lambda = \sigma + i\omega, \alpha_1 = \alpha_1 + i\alpha_{12}, \beta_1 = \beta_1 + i\beta_{12}\), where \(\sigma = Re(\lambda), \omega = Im(\lambda), \alpha_{11} = Re(\alpha_1), \alpha_{12} = Im(\alpha_1), \beta_{11} = Re(\beta_1), \beta_{12} = Im(\beta_1)\). From (20) we get
\[
\sigma + i\omega = \alpha_{11} + i\alpha_{12} + (\beta_{11} + i\beta_{12}) e^{-(\sigma + i\omega)\tau}.
\] (21)

Separating the real and imaginary parts, we have
\[
\sigma = \alpha_{11} + \beta_{11} e^{-\sigma \tau} \cos(\omega \tau) + \beta_{12} e^{-\sigma \tau} \sin(\omega \tau).
\] (22)
\[
\omega = \alpha_{12} - \beta_{11} e^{-\sigma \tau} \sin(\omega \tau) + \beta_{12} e^{-\sigma \tau} \cos(\omega \tau).
\] (23)

We show that equation (22) has a positive real root. Let
\[
f(\sigma) = \sigma - \alpha_{11} - \beta_{11} e^{-\sigma \tau} \cos(\omega \tau) - \beta_{12} e^{-\sigma \tau} \sin(\omega \tau).
\] (24)

Obviously, \(f(\sigma)\) is a continuous function of \(\sigma\), and \(f(\sigma) \leq \sigma - \alpha_{11} + (\beta_{11} + \beta_{12}) e^{-\sigma \tau} \). So we have \(f(0) = -\alpha_{11} + (\beta_{11} + \beta_{12}) < 0\). Obviously, there exists a suitably large \(\sigma(> 0)\) such that \(f(\sigma) = \sigma - Re\alpha_1 - Re\beta_1 e^{-\sigma \tau} \cos(\omega \tau) - Im\beta_1 e^{-\sigma \tau} \sin(\omega \tau) > 0\) since \(\lim_{\sigma \to +\infty} e^{-\sigma \tau} = 0\). By means of the Intermediate Value Theorem, there exists a \(\sigma \in (0, \sigma)\) such that \(f(\sigma) = \sigma - \alpha_{11} - \beta_{11} e^{-\sigma \tau} \cos(\omega \tau) - \rho_{12} e^{-\sigma \tau} \sin(\omega \tau) = 0\). This means that the characteristic value \(\lambda\) has a positive real part. The same as we can prove that the characteristic value \(\lambda\) also has a positive real part if \(\alpha_1\) is a positive real number. Therefore, the trivial solution of system (3) is unstable. Based on Chafee’s criterion, there exists a limit cycle of system (3), namely, a permanent oscillatory solution.

**Theorem 3** Assume that system (3) has a unique equilibrium point for given parameters. Let \(\alpha_1, \alpha_2, \cdots, \alpha_{16}\) represent the eigenvalues of matrix \(A\), and \(\beta_1, \beta_2, \cdots, \beta_{16}\) the eigenvalues of matrix \(B\). Assume that matrix \(B\) has a positive real eigenvalue, or there is at least one eigenvalue, say \(\beta_j\) which has a positive real part \(Re(\beta_j) > 0\), satisfying that \(|Re(\alpha_j)| + |Im(\alpha_j)| < Re(\beta_j)\), then the unique equilibrium point of system (5) is unstable, which implies that system (3) generates a limit cycle.

**Proof** The proof is similar to Theorem 2, we omitted the details.

**A. Simulation Results**

System (3) was been used for the simulation which is the equivalent system of (2). We first select the time delay \(\tau = 0.5\), the other parameter values as \(a_1 = -1.5, a_2 = -1.2, a_3 = -1.35, a_4 = -1.45, b_1 = -1.15, b_2 = 1.85, b_3 = 0.48, b_4 = 1.12, c_1 = 0.64, c_2 = 0.75, c_3 = -0.25, c_4 = 0.85, d_1 = 1.15, d_2 = -1.55, d_3 = 1.45, d_4 = 1.25, m_1 = 1.85, m_2 = 0.8, m_3 = -0.25, m_4 = 0.15, n_1 = -1.95, n_2 = 0.85, n_3 = -1.95, n_4 = -1.45, then \(\mu(A) = 2.95, \| B \| = 3.8, \) and \(\| B \| e^{-\tau} \exp(-|\mu(A)|) = 3.8 e^{-0.5 \exp(-0.5 \times 2.95)} = 0.98\)
1.1816 > 1. From theorem 1, system (3) has an oscillatory solution (see Fig.1). When we change $m_1 = 0.6, m_2 = 0.7, n_3 = -0.25, m_4 = 0.75, n_1 = -0.25, n_2 = 0.55, n_3 = -0.95, n_4 = 0.65$, the other parameters are kept as the above, then $\mu(A) = 2.95$, $\|B\| = 1.4$. In this case, the eigenvalues of matrix $A$ are: $-0.2589 \pm i 1.6655, -0.3258 \pm i 1.8143, -1.5866 \pm i 2.2996, -1.7745 \pm i 2.6666, -2.3492 \pm i 1.2091, -2.4303 \pm i 0.8983, -0.2420, -0.2648, -1.6908, -2.3458$. The eigenvalues of matrix $B$ are: $1.5000, -1.5000, 1.4000, -1.4000, 1.2000, -1.2000, 0.5000, -0.5000, 0, 0, 0, 0, 0, 0$. Noting that $| -0.2420 | < 1.5000$. Based on Theorem 3, system (3) has an oscillatory solution (see Fig.2). However, $\| B \| \cdot e^{\tau} \cdot \exp(-\sigma(J(A))) = 1.4 + e^{*} 0.5 \cdot \exp(-0.5 * 2.95) = 0.4353 < 1$, inequality (9) does not hold. This means that the Theorem 1 is only a sufficient condition. In order to see the effect of the time delay, we select $\tau = 0.2$ and $\tau = 0.8$, respectively, the oscillatory solutions still appeared (see Fig.3 and Fig.4). Then we change the activation function from $\tanh(x)$ to $\arctan(x)$, we see that the dynamic behavior is almost the same (see Fig.5). This suggested that the activation functions do not affect the oscillation. Finally we consider different delay values which is more general case for delayed network model. This means in system (6): $X(t - \tau) = (x_1(t - \tau_1), x_2(t - \tau_2), \cdots, x_{16}(t - \tau_{16}))^T$. An oscillatory solution was still generated (see Fig.6). It was emphasized that bifurcating method is hard to deal with this multiple delays case since one delay should be used as a bifurcating parameter.

B. Conclusion

In this paper, we have discussed the existence of permanent oscillatory solutions in a complex-valued neural network model with delay. Some simple criteria to guarantee the existence of oscillatory solutions have been proposed, which is easy to check as compared to finding the regions of bifurcation. By taking the real and imaginary parts from the complex-valued neural network model we obtained a set of real differential equations. This allows only focus on the instability of a unique equilibrium point for such a system. Some specific numerical simulations were provided to demonstrate the result. These simulations also suggested that our Theorems are only sufficient conditions.

REFERENCES


Fig. 1 Oscillatory behavior of the solutions, delay = 0.5. 
$m_1 = 1.85, m_2 = 0.7, m_3 = -0.25, m_4 = 0.15; n_1 = -1.95, n_2 = -0.95, n_3 = -1.45.$

Fig. 2 Oscillatory behavior of the solutions, delay = 0.5. 
$m_1 = 0.6, m_2 = -0.7, m_3 = -0.25, m_4 = 0.75; n_1 = -0.25, n_2 = 0.55, n_3 = -0.65.$

Fig. 3 Oscillatory behavior of the solutions, delay = 0.2.

Fig. 4 Oscillatory behavior of the solutions, delay = 0.8.

Fig. 5 Oscillatory behavior of the solutions, delay = 0.5. activation function: $\arctan(x).$ 
$n_1 = -0.25, n_2 = 0.55, n_3 = -1.95, n_4 = -0.25.$

Fig. 5 Oscillatory behavior of the solutions, delay = 0.8. 
$n_1 = -0.95, n_2 = 0.65, n_3 = -1.45, n_4 = -1.15.$
Fig. 6 Oscillatory behavior of the solutions, delays: [0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.0].

(a) Solid line: $u_1(t)$, dashed line: $u_2(t)$, dotted line: $u_3(t)$, dashdotted line: $u_4(t)$.

(b) Solid line: $v_1(t)$, dashed line: $v_2(t)$, dotted line: $v_3(t)$, dashdotted line: $v_4(t)$.

(c) Solid line: $p_1(t)$, dashed line: $p_2(t)$, dotted line: $p_3(t)$, dashdotted line: $p_4(t)$.

(d) Solid line: $q_1(t)$, dashed line: $q_2(t)$, dotted line: $q_3(t)$, dashdotted line: $q_4(t)$. 

Copyright © 2018 by CENPARMI, Concordia University
Distance Estimation Based Energy Efficient Protocol for Wireless Sensor Network

L. Rajesh, K. Bhoopathy Bagan, Abdur Rahman

Department of Electronics Engineering
MIT Campus, Anna University

Abstract: Recent advancement in Micro-electronics, Networking, and Integrated Digital Electronics led to the development of Wireless Sensor Network (WSN). Usually the sensors are deployed randomly in ad hoc fashion from flights in disaster management operations. In such random deployment of sensors, there are possibilities that different sensors might have same region of coverage. Thus data from those sensors that has same region of coverage are redundant and adds traffic to the network. Identification of redundant sensors can be done by calculating the distance between every sensor by Received Signal Strength Indication (RSSI) Method. Energy efficiency and increase in life time of the network can be achieved by the proposed algorithm that suits the random sensor environment.

Keywords: Wireless Sensor networks, RSSI, LEACH, Cluster head Selection

I. Introduction

Parallel advancements in embedded systems and Communication techniques have led to the proliferation of Wireless Sensor Networks. Wireless Sensors acquire information from the environment and transmit it through wireless channel to the destination. The received information is processed in destination. Most of these sensors are deployed in groups to form networks with different topologies.

A sensor in the network is termed to be sensor node. In addition to transmitters, sensor nodes also require receivers particularly in applications where the sensors change its characteristics based on the received data. On the whole a sensor node usually consists of a transducer to acquire data, transmitter and receiver to transmit and receive information respectively and in some application a processing unit. Wireless Sensor Network has wide range of application that includes Industrial Automation, Home Automation, Medical Device Monitoring, Robot Control, Disaster Management and Rescue Operations.

Most sensor nodes are battery powered, once if the battery gets drained it should be recharged or replaced else the node will die. Therefore energy is a great demand in wireless sensor networks. Ample amount of work that aims to achieve energy efficiency has been done. Wireless Sensor Networks are prominently used in disaster management in rescue management. In such rescue operations sensors are deployed into the disaster prone areas from flight. This deployment of sensor nodes are purely random. On random deployment there are possibilities that same sensor might cover same area, there occurs the redundancy among them. By using a suitable algorithm to identify this redundancy among sensors, energy efficiency can be achieved and life time of the networks can be increased.

II. Related Works

[1] Works have been done to reduce redundancy among sensors and its data. Global Positioning Systems (GPS) can be used during deployment of sensors. By GPS devices attached to the sensor nodes the exact latitude and longitude values of the nodes can be found and the redundancy among sensors is identified. The performance of this method is reflected by the localization errors. Moreover GPS devices equally consume same amount of energy as that of sensor nodes hence the nodes may drain out easily.[2] Low Energy Adaptive Clustering Hierarchy (LEACH) is an energy aware routing protocol for homogeneous sensor networks. In LEACH protocol the nodes are grouped into clusters and a node takes the lead role as cluster head. LEACH protocol has two phases Cluster Formation Phase in which Threshold based cluster selection is done. Every node produce a random number within the range 0 to 1. The number is compared with the threshold value T(n) and if the generated number is less than the threshold value the node is considered to be cluster head. In the Steady State phase the Cluster head receives data from other nodes in the cluster and sends it to sink node.[3] Though LEACH is energy aware protocol, the Cluster head selection is probabilistic in nature, the probability of selection of node as cluster head for both low energy and high energy node is same. ELEACH is an improvement of LEACH protocol in which current energy of the node is considered into account on Cluster Head selection.[4] Data aggregation tree based works have been proposed where sensors are grouped as tree. The whole area is divided into sub regions. When the sub query region continues to grow the sink nodes need to send the query message to root nodes which results in increase in energy consumption.

III. System Model

1. Energy Model
The First order radio energy model is taken into account. The two major components of the system responsible for energy consumption are Power amplifier and Radio circuitry. [5] Energy loss from these components are considered in this model. Energy consumed to transmit $k$ bit data packet for a distance $d$ is given in (1) as 

$$E_{tx} = k \cdot E_{elec} + k \cdot E_{amp} \cdot d^2$$

(1)

Where

$E_{elec}$ is the energy consumption of the radio electronic circuitry.

$E_{amp}$ is the energy consumption of transmitting amplifier.

The considered parametrical values are given in the Table 1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_{elec}$</td>
<td>50 nJ/bit</td>
</tr>
<tr>
<td>$E_{amp}$</td>
<td>100 pJ/bit/m^2</td>
</tr>
<tr>
<td>Total number of bits, $k$</td>
<td>10</td>
</tr>
<tr>
<td>Distance, $d$</td>
<td>10 Km</td>
</tr>
</tbody>
</table>

The value of loss is found to be 0.1 Joule per round for transfer of 10 bits.

Table 1. Parameters of energy model

2. Distance Estimation

As the first step of routing scheme every sensor node sends a packet to all other sensor nodes. This process is termed as flooding. Sensor nodes that receives the packet from other sensors measures the signal strength of the received signal. [6] From the knowledge of the received signal strength, the distance at which the transmitted sensor node is located from receiving sensor node can be estimated by path loss model

$$P = \left( \frac{P_t \cdot G_t \cdot G_r \cdot \lambda}{4\pi f^2 d} \right)$$

(2)

Parameters considered,

$P_t, P_r$ - Power of the Transmitter and Receiver sensors

$G_t, G_r$ - Gain of the Transmitter and Receiver sensors

$\lambda$ - Wavelength

$d$ - Distance between the sensors

Table 2. Parameters of path loss model

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmitter Gain, $G_t$</td>
<td>1</td>
</tr>
<tr>
<td>Receiver Gain, $G_r$</td>
<td>1</td>
</tr>
<tr>
<td>Frequency, $f$</td>
<td>433.3 MHz</td>
</tr>
<tr>
<td>Wavelength, $\lambda$</td>
<td>0.69 m</td>
</tr>
<tr>
<td>Transmitting Power $P_T$</td>
<td>0.1 W</td>
</tr>
</tbody>
</table>

At the end of the flooding process, every sensor will have a database of distance to all other sensors in the given area.

IV. Proposed Work

1. Random Node Deployment

A (10X10) Sq.Km area is considered. 100 Sensor nodes each capable of transmitting and receiving data is randomly deployed in this area. The deployed nodes are shown in the Fig 1.
3. Grouping of Redundant Sensors

In order to have a limitation over the size of the cluster to be formed, there is need to break the cluster though each sensor is redundant with the successive one, else the farthest nodes of the same cluster which is not really redundant will be considered to be in the same redundant cluster. Therefore a limitation is placed by separating the whole area into grids of (1x1) Sq.km in order to avoid the growth of cluster. Only sensors that are within this grid and physically redundant are considered to be redundant sensors. The grouping of sensors is given in the Fig 2.

4. Cluster Head Selection

Instead of sending the packet from all the redundant sensors to base station, Cluster Head is chosen from every clusters. For the selection of cluster head E-LEACH is to be used. E-LEACH is the advanced version of LEACH protocol [6]. In the setup phase of LEACH, Cluster Head is to be chosen with respect to the maximum energy of each of the cluster. In the first step of setup phase, Cluster Head advertisement take place. During this step cluster head sends the advertisement packet to inform the cluster nodes that they have become a cluster head on the basis of the following formula:

\[ T(n) = P(1 - P \mod P^2) \]

\[ T(n) = 0 \]

Where

- \( n \) is a random number between 0 and 1
- \( P \) is the Cluster Head Probability
- \( G \) is set of nodes that weren’t the Cluster Heads for previous rounds

\( T(n) \) is the threshold. Node becomes cluster head for the current round if the number is less than threshold \( T(n) \). Once node is selected as a cluster head then it cannot become cluster head again until all the nodes of the cluster have become cluster head once. This is useful for balancing the energy consumption. As E-LEACH protocol is used in addition to the probability, the remaining energy of the nodes is also taken into account during cluster head selection.

V. Results and Discussions

A real time scenario of transmission of data from sensors is simulated. As the first step the nodes are randomly deployed. The distance between every deployed nodes to the other nodes is estimated by the received signal strength estimation by path loss model. The nodes to nodes and the corresponding distance between them are plotted in Fig 3.

From the estimated distance and grid grouping cluster of redundant nodes are identified, one such cluster is provided in Fig 4.

By redundancy identification based transmission of packets, the energy level of every redundant node has increased and is shown by considering a sample node (Second Node) and its corresponding energy level for both normal transmission of packets and transmission by proposed method is analyzed and is given in Fig 5.
It can be inferred from the Figure 5 that the sample node can withstand till 1500 rounds of data in the proposed method whereas the same node can withstand for only 1000 rounds during normal transmission of data. It could be inferred from the figure that in the proposed method the redundant nodes have remaining additional energy whereas the entire energy is depleted after 1000th node in the usual node. The energy efficiency of the best node and the worst nodes are also analyzed and is depicted in Fig 6 and Fig 7 respectively.

It could be observed in Fig 6 that the number of rounds the best node (Tenth Node) can withstand has increased from 1000 to 3000 rounds by the proposed method.

It can also be inferred from Fig 7 that the number of rounds improvement by the proposed method is 180. The proposed redundancy removal model had made nodes to withstand from 1000 to 1180. The life time analysis of all deployed node is also done. The life time of a node is the number of rounds till the entire energy of that particular node gets depleted. The life time analysis is plotted in the Figure 8.

It could be inferred from the life time analysis depicted in Fig 8 that the lifetime of the entire network has increased from 1000 rounds to 3000 rounds. Moreover the exact percentage increase the proposed algorithm provides depends on the randomness of the deployment of nodes.

V. Conclusion and Future Work

The scenario of sensors are simulated and the redundancy is identified. The proposed algorithm is applied to redundant gro
p of sensors. It can be concluded that energy efficiency can be achieved by the proposed method with parallel increase in lifetime of the network. The percentage increase in amount of efficiency and lifetime of the network depends on the randomness of the deployed node. Advancement in the proposed method can be done by using other path loss model that could suit the environment in which sensors is deployed. Additionally improved grouping schemes can be used with the proposed protocol to enhance the performance.

V. References


LS- SVM’s no-reference video quality assessment model considering the network packet loss under the internet of things

1st Yibin Hou
dep. Department of Information, Beijing University of Technology
School of Software Engineering
Beijing, China
ybhou@bjut.edu.cn (Yibin Hou, phone: 13911114088),
yhou@bjut.edu.cn (Yibin Hou),

Abstract—The Internet of things, including Internet technology, including wired and wireless networks. In order to study the influence of packet loss on QoE and establish the Mapping model of packet loss rate and the Quality of experience when the video transmit in the network, and in order to have a better evaluate of video quality which through the network transmission, building a NS2+ MyEvalvid simulation platform, conducting simulation and evaluation for the video transmit in the network, focus on the influence of packet loss on QoE and establish the Mapping model of packet loss rate and the Quality of experience. And extract features, using Least squares support vector machine method to establish no-reference video quality assessment model considering the network packet loss. Simulation results show that, packet loss has a significant influence on Quality of experience. Packet loss rate and the Quality of experience presents a nonlinear relationship. Therefore, based on the study of packet loss has a significant influence on Quality of experience, using a nonlinear regression analysis method to establish the Mapping model of packet loss rate and the Quality of experience. And extract features, using Least squares support vector machine method to establish no-reference video quality assessment model considering the network packet loss. The experimental results show that, LS-SVM’s training speed is fast, the model is more accurate than the other models.

Keywords— No-reference; Quality assessment model; Network packet loss; Least squares support vector machine; Internet of things;

I. INTRODUCTION (HEADING 1)

At present, the development of international information prompted the international people's exchanges widely, network video business is increasing. In the information industry, network video business has become the most popular application of computer network field, from the international and domestic exchanges to life and entertainment, video penetrated into every aspect of our lives. But the network itself is not perfect, in essence, is a kind of distortion network. More and more people favor HD video. But at the same time, people are doubly worried at that the network video quality which watching from the network terminal that is also the users’ computer can not meet the needs of the users themselves. Therefore, causes the academia and industry professionals think that, what causes the decrease of the quality of the video and how to evaluate the quality of the network video. So, we set up the video quality assessment model to evaluate the quality of video. In order to get a better QoE (Quality of experience) [1, 2], we research on the factors affect the video quality and use them to establish no-reference video quality assessment model [3] considering the network packet loss based on Least squares support vector machine [4, 5]. There are many factors affect the quality of network video, such as the code layer parameters GOP pattern, the quantization parameter Q. The network parameters including the packet size, the packet error rate, the different output link speeds and so on, and in the end, through AWK analysis of the .tr files or through MyEvalvid obtain sd, rd files to gain packet loss, jitter, delay such QoS parameters, can choose the best routing path according to these parameters, in order to improve the resource utilization of the network. The Content layer parameters including different time complexity and spatial complexity and so on.

In related work, the reference [6] mainly proposed scale characteristic of packet loss, scale characteristic mainly refers to in the process of packet loss reflects the long correlation. Long-range dependence with the self-similarity are two equivalent concepts. Because of the long correlation will affect the packet loss and further affect quality of service (QoS), and finally affect the QoE. The packet loss rate is an important measurement parameter in QoS. So study Long-range dependence has an important value and significance. References [7, 8] researches in network traffic, discusses several existing Long-range dependence model, comparing their respective advantages and disadvantages, and predict the network traffic associated short and Long-range dependence into account, is the future direction of the traffic model, and be able better to describe any network traffic development direction. Through the study found the number of superimposed source N, shape parameter, Hurst parameter, the output link speed has impacts on Long-range dependence, and further affects the packet loss rate. Finally, it is concluded that different output link speeds have a significant impact on long-range dependence, and further to affect the packet loss rate. At the same time, this conclusion will be applied to establish the no-reference video quality assessment model considering the network packet loss. Reference [9] using the MPEG4 codec and HD video under MyEvalvid platform research on the influence of packet loss on QoE and set up the mapping model of packet loss rate and the Quality of experience in Matlab environment. Finally concluded that the different content
complexity and different packet loss rate have a significant impact on the user’s quality of experience. At the same time, this conclusion will be applied to the third part of the contents, select content complexity and packet loss rate as parameters to establish the model. Reference [10] presented an adaptive scheme that is the experiential quality driven in order to optimize the content supply and network resource utilization for application of video in wireless networks. Reference [11, 12, 13] consider the packet loss concentration and packet loss rate’s impacts on the quality of video, proposed a no reference video quality assessment model consideration of packet loss characteristics. Reference [14] considering different frame type drop and packet loss rate impacts on video quality, put forward a kind of method don't need to decode video which called no-reference video quality assessment model. Reference [15, 16] puts forward support vector machine (support vector machines, SVM) which has become the current research hotspots in the field of machine learning, it has excellent learning performance. Reference [17] for problems which exists in least squares support vector machine parameter optimization, proposed least squares support vector machine (SVM) parameters self-tuning optimization algorithm which is cross validation. Finally, compare the least squares support vector machine (SVM) and the traditional BP network’s forecast result, the experimental results show that, the model’s prediction accuracy is satisfactory, the proposed method is feasible. On the whole, this paper finally establishes no-reference video quality assessment model considering the network packet loss based on Least squares support vector machine.

II. EXPERIMENT ENVIRONMENT

A. The construction of the experimental environment

Because in the real network, unable to quantify to simulate the network with different degree of damage. Therefore, network simulation environment is built on cygwin+ NS2 at the windows, through the method of changing QoS parameters to simulate different levels of network damage. Then NS2 and Evalvid integration MyEvalvid. The MyEvalvid make the Evalvid function strengthen, the Evalvid through the myEvalvid, my_UDP, my_Evalvid_Sink three interface program to communicate with NS2. Because of the myEvalvid is the core of this environment, so is introduced as follows.

(1) myEvalvid: the interface program’s main job is to learn to read VS procedures after the film log files, the log files of each picture was cut into smaller segments, and in the user in Tc Script set in good time to the section is the bottom of the UDP layer sends out.

(2) my_UDP: Basically my_UDP Agent is the extension of UDP Agent. The new Agent the packet transmission time, packet identification and packet load size recorded in the files which was set up by Tcl Script.

(3) myEvalvid_Sink: It is responsible for the work that receives a packet which is transferred out by my_UDP, and record the receive time, packet identification and packet load size, recorded in the files which was set up by Tcl Script.

B. The principle of using myEvalvid simulation

First of all, under the environment of NS2 conducting video transmission in the network simulation, then use the MyEvalvid for PSNR value assessment. The experiment makes original files YUV encoding into MPEG4 format video files, then under the environment of NS2 for network simulation, finally in MyEvalvid calculates the objective evaluation method’ is PSNR value and give the user's quality of experience. Experiment of principle. First of all, yuv files to m4v files, then m4v files to mp4 files, then extracted information related with every frame, stored in a file which extension name is the st. By running TCL script conducting the NS2 network simulation. After the completion of the simulation produces the sender record files, the receiver record files and video recording files, use these three files and mp4 files generated the reconstruction mp4 files, based on this step make the reconstruction mp4 files back to yuv video files. In the end, calculate the average PSNR value of all frames of the reconstruction video files, give the user’ s Quality of experience. At the same time, can use AWK program to analyze simulation record files to obtain packet loss rate.

III. THE LS-SVM THEORY

A. An experimental environment and the overall architecture figure

In this paper, experimental environment is mainly NS2+ cygwin+ MyEvalvid environment, use this environment get experimental data and finally use Matlab to establish no-reference video quality assessment model considering the network packet loss based on Least squares support vector machine and at the end conducting the comparison of performance.

B. ls-SVM theory

Support vector machine (SVM)’ s basis is the optimal separating hyperplane, as shown in figure 4. Least squares support vector machine (SVM) regression is also known as the LS-SVR, least squares support vector machine is often used for multivariate nonlinear regression analysis, nonlinear simulation and forecasting. Least squares support vector machine is running fast, accuracy is high.

IV. ESTABLISH LS-SVM AND THE ANALYSIS OF EXPERIMENTAL RESULTS

A. Quantization parameter affecting the users’ quality of experience

The use of quantitative parameter Q value is bigger, the video quality QoE will be worse and worse. This is because in the video compression, when the use of quantitative parameters is bigger, the video quality QoE after compression will be worse, the video through the network transmit to the receiving end, it video quality is worse than quantitative parameters which is small, and when use smaller quantitative parameters, it will have a good video effect. At the same time, we can also find, the compression data quantity which use Q value is more, it need more packets to send[18].

This work was partially supported by the National Natural Science Foundation of China (No: 61203377, 60963011).
B. Different output link speeds have a significant impact on long-range dependence

Through the study found the number of superimposed source N, shape parameter, Hurst parameter, the output link speed has impacts on Long-range dependence, and further affects the packet loss rate. Finally it is concluded that different output link speeds have a significant impact on long-range dependence, and further to affect the packet loss rate. At the same time, this conclusion will be applied to establish the no-reference video quality assessment model considering the network packet loss.

For a generalized stationary random process \( \{X_n\} \) set \( X_n \) with constant mean \( \mu = E(X_i) \) and finite variance \( \sigma^2 = E[(X_i - \mu)^2] \), the self correlation coefficient is:

\[
\gamma(k) = E[(X_i - \mu)(X_{i+k} - \mu)]/\sigma^2, \quad (k=0, 1, 2, \ldots)
\]

The self correlation coefficient only concerned with k, the number of network business entity in the k-th unit of time arrival is called \( X_k \). Use \( L_k(k) \) represents a slowly varying function, that is

\[
\lim_{t \to \infty} L_k(k) = 1, \quad \text{and for all } X > 0 \text{ set up } \gamma(k) - k^{-\beta} L_k(k), \quad \text{said the process to satisfy the above conditions called progressive self-similar process}[19].
\]

V. ESTABLISH LS-SVM’S NO-REFERENCE VIDEO QUALITY ASSESSMENT MODEL

A. Topology description and HD video options and the experimental process

Default network includes wired and wireless environment, but in the end, most adopted wired experimental environment, and experimenting under the same topology structure, mainly consider the principle of a single variable. Wired topology structure consists of 4 nodes, between n0 and n1, n2 and n3 are duplex links, link bandwidth is 10 MBPS, delay time is set to 1 ms. Between n1 and n2 is simplex link, the bandwidth is 640 KB, delay time is set to 1 ms. Simplex or duplex queue management mechanism adopts DropTail, and the length of queue is 50 packet size. Establish udp1 Agent on n0, the packet size is set to 1500 bytes, establish a NULL Agent on n3, the packet size is set to 1500 bytes. The simulation time is 10s, run TCL file and generates results after simulation, the results have two records, one is recorded in the simulation log files out.tr file, one is to produce NAM file out.nam. With the rapid development of wireless applications, people look to the wireless network communication. But wireless environment compared with wired environment, usually has a high bit error rate, time delay large, bandwidth low, channel asymmetry and frequent mobile features and so on. Therefor, under the environment of wireless network congestion can no longer be seen as the only reason for the loss of data, there are a large number of data due to the switch, channel fading, disturbances and so on to be discarded, the packet loss due to wireless environment called wireless packet loss. In a word, under the environment of the wired, congestion loss is the only cause of the loss of data, while in the wireless environment, congestion loss and wireless loss can both cause data loss. As a result, the study of the wireless network will be more complicated than the wired network environment. Wireless topology consists of 4 nodes, n0-n3 these four wireless nodes, n0(200, 400), n1(200, 300), n2(400, 300), n3(400, 400). n0 settings for the mobile node, in the simulation time of 5s, from the starting position at a speed of 5 m/s move to the ending position (300, 400), then at the simulation time of 25s, to (200, 400), that is the starting position of n0. Establish udp1 Agent on the n0 node, establish null1 Agent on the n3 node, then set up online above. The simulation time is 50s. The wired experimental topology structure as shown in fig. 1. Wired simulation process NAM demo as shown in fig. 2[9]. Wireless topology structure and NAM as shown in the fig. 3 and 4.
We select the HD video sources to conduct the experiment, HD video sources mainly have 525 series and 625 series. These two sequences are both the video quality experts group VQEG’s test sequence, 525 sequence containing ref and HRC two series, 625 also includes ref and HRC two series. Ref is the reference state sequence, each HRC is a defect of the original reference sequence. Among them, the 525 sequence frequency is 60HZ, The frame size is 1440 x 486 or 699840 bytes per frame, the 625 sequence frequency is 50HZ, the size is 1440 * 576 or to say 829440 bytes per frame. YUV format is 4:2:2 format, a pixel are two bytes. Hd video is divided into two kinds in total, a video is strenuous exercise, video’s content complexity is high, another video is the scenery, video’s content complexity is low. On the basis of fully understand the HD video, select human strenuous exercise, high content complexity video src13_hrc1_525.yuv and the scenery, low content complexity src22_hrc1_525.yuv these two videos. Two selected video src13 as shown in fig. 5, src22 as shown in fig. 6[9].

Here, we take hd video src13_hrc1_525. yuv experiment as an example. Src13 wired and src22 wired and src13 wireless’s scatter plots respectively as shown in figure 23, figure 24, two figures horizontal axis are packet loss rate, vertical axis represents the PSNR values. For src13 video, data points looks less, the reason is in the process of experiment, first has carried on the exploratory experiments, in the process of experiment, because of the need to use a single variable method to modify the QoS parameters, therefore, exploring the parameter values each time how much interval once can not affect obtain the final turning point, and in the case of reduce test times as much as possible to get the packet loss rate’s turning points between 0 and 1, mainly using the vernier caliper’s main ruler and the principle of the vernier. For src22 video, delay time setting has no effects on the user’s quality of experience and packet loss rate. Link speed lower limits are 10 MB and 74 KB. Packet size lower limit is 540kb, upper limit is 2050kb. Quantization parameter ranges from 0.01 to 255. So, packet loss rate very few take between 0.6-1, in the same way, using the basic principle of the vernier that is use as little as datas possible to avoid influencing turning points of data points. Because of using a single variable method, use this way to find how to make the packet loss rate dispersion exists between 0 and 1 and better single variable step size of the data values change, better single variable data values change step, mainly in order to better artificial control the distribution of data positions, improved random parameter combinations leading to the final results before it is the distribution of data uncertainty and no regularity and without purpose, can be more rules more objective for parameter setting and let the results interval can be controlled[9, 20]. As shown in fig. 7, fig. 8.

For src1 wired network environment use LS-SVM method establish no-reference video quality assessment model considering the network packet loss based on Least squares support vector machine as described below. (1)Enter a value for the X[21]. The input sequence is quantitative parameter, packet loss rate, the simplex output link speed, the duplex output link speed, video’s time complexity, and these parameters are of equal weights. X=[X1, X2, X3, X4, X5], X1=[31, 31, 31, 31, 10, 10, 5, 31, ...31]T, X2=[0.553 226, 0.550 704, 0.549 443, 0.549 758, 0.552 805, 0.283 545, 0.540 145, 0.537 262, 0.538 203, 0.539 702, 0.542 737, 0.541 647, 0.542 729, 0.545 425, 0.545 541, 0.547 346, 0.547 459, 0.549 063, 0.548 133, 0.548 719, 0.552 747, 0.550 203, 0.548 882, 0.555 249, 0.549 23, 0.549 73, 0.550 27, 0.551 24, 0.550 99, 0.551 24, 0.551 59, 0.551 81, 0.552 25, 0.552 64, 0.552 64, 0.553 01, 0.549 23, 0.554 3, 0.555 01, 0.557 29, 0.556 4, 0.555 95, 0.558 69, 0.561 01, 0.557 9, 0.560 59, 0.557 84, 0.549 23, 0.558 97, 0.560 698, 0.559 340, 0.561 969, 0.560 797, 0.563 930, 0.560 823, 0.565 552, 0.564 101, 0.563 499, 0.567 126, 0.556 936, 0.087 487]T, X3=[6.4,
It's value
RMSE
so
b x x k
5253.0),
(RBF_kernel', 'preprocess')). Trainlssvm function is LS-SVM, statements
\[\alpha, b\] = trainlssvm({X, Y, type, gam, sig2, '

Therefor, for src22 wired and src13 wireless network environment use LS-SVM method establish no-reference video quality assessment model considering the network packet loss's method is the same as src13 wired network environment[22]. Establishment's results as follows: Src22 wired:
\[f(x) = \sum_{i=1}^{l} \alpha_i k(x, x_i) + b\] so
\[f(x) = \sum_{i=1}^{l} \alpha_i k(x, x_i) + 0.5253\] among them, \(\alpha_i = [-0.016 6, 0.017 5, -1.004 6, -1.083 2, -1.360 2, 0.704 3, 0.017 5, 5.372 5, 28.927 8, 86.043 5, -54.786 1, -48.890 1, -32.544 7, -32.814 8, -26.010 3, -32.050 1, -17.214 1, -15.699 6, -20.574 9, 0.237 4, 0.366 5, 0.725 7, 0.1627, 0.725 7, -0.526 2, 0.366 5, 0.366 5].

B. The analysis of experimental results
The evaluation indexes are mainly R-square, RMSE (Root Mean Square Error, RMSE), SSE, SROCC (Spearman Rank Order Correlation Coefficient, SROCC), Pearson, and these values are between 0 and 1[23].

<table>
<thead>
<tr>
<th>video</th>
<th>R-square</th>
<th>RMSE</th>
<th>SSE</th>
<th>SROCC</th>
<th>Pearson</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS-SVM</td>
<td>0.8889</td>
<td>0.006</td>
<td>0.112</td>
<td>0.998</td>
<td>0.9883</td>
</tr>
<tr>
<td>BNeural network</td>
<td>0.7999</td>
<td>0.66</td>
<td>0.222</td>
<td>0.888</td>
<td>0.45</td>
</tr>
<tr>
<td>PSNR</td>
<td>0.588</td>
<td>0.16</td>
<td>0.2</td>
<td>0.634</td>
<td>0.71</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.666</td>
<td>0.15</td>
<td>0.18</td>
<td>0.815</td>
<td>0.83</td>
</tr>
</tbody>
</table>

The horizonal header is coefficient of each measures, the vertical header is a variety of methods, for these evaluation indexes, RMSE and SSE are the smaller the better, while R-square, SROCC, Pearson are bigger are better. LS-SVM
method suitable for small samples training, BP neural network\textsuperscript{[24]} suitable for big samples training, so here is suitable use the LS-SVM method\textsuperscript{[25, 26]}. The experimental results show that, that is observe Table I we can find, LS-SVM 's performance is better than BP, PSNR and SSIM, and training speed is fast, the model is more accurate.

VI. CONCLUSION

Set up considering network packet loss of video quality evaluation model, this is due to affecting the quality of video is not only packet loss rate, there has other factors, but the packet loss rate occupies the main status, therefore, on the basis of considering packet loss rate consider other factors to establish video quality evaluation model consider the network packet loss, more accurate prediction of user's quality of experience QoE and establish QoS and QoE model is the future work direction. Experiments results show that, the LS-SVM has better generalization ability, and the training speed is faster. Next step is research on the Model of QA system. Algorithm flow and framework. 1. corpus analysis. The corpus is transformed into a vector and the problem is transformed into a vector. The answer vector is computed by vector. Find the sentences in the corpus and return. 2. corpus storage. The corpus is transformed into a vector and stored later. Can be stored in a data table. 3. information segmentation. Divide information into words. 4. extraction of characteristic words. Key words are used to find answers to questions. 5. characteristic word detection. Check the search element is not key words. 6. estimated relevance assessment. Prediction result. 7. shows the evaluation results of the Logistic model. Display results. Use 
\[ p(t) = \frac{k P_0 e^{rt}}{(K + P_0 (e^{rt} - 1))} \]

Among them, \( P_0 \) is Initial value, \( K \) is the final value, \( R \) measuring the speed of curves. It is like a "S" type, so it is also called the sigmoid curve \( S \) curve. Use \( y = -24.3816 \times (0.5 - 1/ (1 + (0.56962 \times (x - 27.49855)))) + 1.9663 \times -2.37071; \) achieve fast and accurate search to demand answers, mainly using the search algorithm to quickly find the right answer.

ACKNOWLEDGMENT (Heading 5)

This work was partially supported by the National Natural Science Foundation of China (No: 61203377, 60963011).

REFERENCES


SAF-AFIS: SIFT-based Alignment Free Approach for an Automatic Fingerprint Identification System

Hamza Djebli, Samy Ait-Aoudia
Ecole Nationale Supérieure d’Informatique - Oued Smar, Algiers, Algeria
{h_djebli, s_ait_aoudia}@esi.dz

Abstract—Automatic fingerprint recognition is a trending topic both in research and industrial communities, and is still challenging. In this work, we propose an automatic identification system based on fingerprints where no prior alignment is required. The proposed architecture is based on Gabor filters and SIFT descriptors, with an identification process inspired from image mosaicking. The systems achieves good results compared to other related systems.

Keywords: Biometrics, automatic fingerprint identification systems, SIFT, Gabor filter, mosaicking

I. INTRODUCTION

Automatic fingerprint identification is a largely used application in privacy protection field and an interestingly growing topic in image processing and pattern recognition research [1, 2, 3]. Regardless of the number of developed systems addressing this application, the problem of automatic person identification is still challenging. Fingerprint identification systems can be classified into two main categories: binarization based systems and grayscale-based systems [4, 5]. In the first category, a binarization of the grayscale image is performed as a first step, whereas the second type uses directly the image data (pixels) to extract salient features called Minutiae. While the second category preserves pixel information from the fingerprint image, they suffer from several skin conditions (moisture, creases, etc.), contrast variations and image noise. In addition to that, a pre-alignment phase is often required to allow pixel comparison for the same fingerprint regions.

In our work we present a new technique for automatic fingerprint identification inspired from image mosaicking, relying on the Scale Invariant Feature Transform (SIFT), which provides highly distinctive and scale invariant local features. For this purpose, we opted for the binarization approach since we need to extract a set of minutiae from the fingerprint image which will be used in the identification step.

The next section presents the proposed global architecture. Section III shows the different parameters used in our algorithm. In Section IV, we present some experimental results and the databases used for testing. Section V presents some related work in the same topic. Section V concludes this paper.

II. RELATED WORKS

Identification of persons relying on the extraction of fingerprint minutiae is a classical topic that has been addressed in several scientific works. A comprehensive survey on existing methods can be found in [4], [5] and [6]. Different types of fingerprint features based on minutiae positions have been proposed: Chaincode [7], run based methods [8], ridge flow and local pixel analysis based methods [9, 10].

To the best of our knowledge, few works in the literature addressed the use of SIFT features and affine transformations for fingerprint identification. In [11], the authors proposed a method based on SIFT, however, the image enhancement step for eliminating the noise is not used. In addition, they begin by extracting the SIFT descriptors on the whole area of the image, which generates a very large number of candidate points, which implies longer execution time and requires a post-processing phase. The method proposed in [12] extracts minutiae by the same method but without skeletonization. It uses a combination of SIFT descriptors for each minutia, and proceeds in two steps for image alignment, using the Hough Transform.

The novelty in our approach is that we optimize the SIFT features detection by providing precise positions of minutiae after extraction. Furthermore, we provide independent feature vector for each minutia. This will allow fingerprint identification even when the set of extracted minutiae is not the same in different fingerprint samples.

III. SYSTEM ARCHITECTURE

Fingerprint images are represented by a set of ridge and valley patterns, where ridges are usually seen in black, and valleys in white. Before proceeding with the minutiae extraction, an initial phase of preprocessing is required to improve the quality of ridges and valleys. To do this, we follow an approach based on anisotropic filters, which make use of the image orientation, namely the Gabor filter [13, 14, 15]. Figure 1 illustrates the overall architecture.

A. Image Normalization and Segmentation

The two main operations needed in the preprocessing are image normalization and segmentation. The normalization goal is to adjust image intensities in order to obtain desired values for mean and a standard deviation. This is done in order to reduce grayscale values variations through the textures of the fingerprint, and to prepare the image for next steps. Then the segmentation of the fingerprint area is carried out by keeping the blocks with a standard deviation greater than a given value, in order to separate the real fingerprint region from the image background, and to filter any noise outside the fingerprint region.
B. Orientation Calculation

Since the fingerprint has a pattern of oriented ridges and valleys, the orientation is calculated at each pixel of the image, using the method described in [16]. The orientation is estimated by blocks of given size (16x16), based on the gradient of the image.

C. Ridge Frequency Estimation

Variation of grayscale levels can be modeled by a sine wave perpendicular to the orientation of ridges. Using the normalized image and the orientation already calculated, the ridge frequency is estimated as described in [9].

This step gives a map of ridge frequency for each pixel inside the fingerprint area.

D. Image Filtering

In order to improve the distinctiveness of ridge areas in the fingerprint, and to eliminate the effect of noise which can occur due to different sampling conditions, we go through an image filtering step. By exploiting the sinusoidal nature of ridges and valleys alternations, and using the orientation already calculated (Step B), a bandpass filter adapted to the direction and the peak frequency is performed. For this we use a Gabor filter, which satisfies the aforementioned conditions, based on its description in [9].

E. Binarization

In this step, we reduce the representation of the filtered image to a binary form, where only zeros and ones are present. We proceed by blocks of 16x16, where each block is filtered according to the average value of pixels within the same block. Pixel values which are below this value are converted to 0. The remainder is set to 1.

F. Skeletonization

The goal here is to reduce the width of ridges obtained from the previous step to 1 pixel. An iterative process is followed, so that at each iteration redundant pixels are marked to be set to 0 in the result image. After a specified number of iterations, or when the image undergoes no change, marked pixels are deleted (reset to the value 0).

G. Minutiae Marking

After skeletonization, marking the minutiae is done by calculating the Crossing Number (CN) for each pixel x of the skeleton image S where S(x) = 1. If CN(x) = 1 we mark a ridge ending. If NC(x) = 3 we mark a bifurcation. The other cases are not considered.

H. Spurious Minutiae Removal

We have included in our system a routine for the detection and elimination of false minutiae which can appear due to several factors (wounds, contrast variation, noise, etc.). The main spurious minutiae types processed are: close pairs (bifurcation-ending, bifurcation-bifurcation, ending-ending), islands, and bridges between ridges. This step reduces the number of minutiae detected in fingerprints, yet it gives more precision in the fingerprints matching step.

The previous steps (A-H) mark the positions and orientations of minutiae points in the fingerprint. The following steps serve to estimate the correspondence between two fingerprints, following an approach similar to image mosaicking.

I. Extracting SIFT descriptors

In this step, we extract the Scale Invariant Feature Transform (SIFT) descriptors [18] at the locations of already marked minutiae. Since SIFT descriptor is invariant to scale and orientation changes, it provides a stable minutiae feature set throughout several fingerprint acquisitions, regardless of translations and rotations changes in acquisition. Due to this SIFT features characteristic, we are able to skip the images pre-alignment problem before correspondence estimation.
We point out that we follow an “indirect” method for extracting SIFT features. Usually, the SIFT algorithm extracts both the descriptors and their locations. In our case, we provide the algorithm with specific locations, formed by the set of marked minutiae, and it calculates the SIFT features related to those locations, regardless of whether they are local maxima or not. As for the orientation, its values are provided by the algorithm in both approaches. We choose a constant scale value for the SIFT descriptors all across the image. The choice is defined empirically using testing samples.

Alignment in this case is implicitly done during the estimation of the transformation function.

We point out that we follow an “indirect” method for extracting SIFT features. Usually, the SIFT algorithm extracts both the descriptors and their locations. In our case, we provide the algorithm with specific locations, formed by the set of marked minutiae, and it calculates the SIFT features related to those locations, regardless of whether they are local maxima or not. As for the orientation, its values are provided by the algorithm in both approaches. We choose a constant scale value for the SIFT descriptors all across the image. The choice is defined empirically using testing samples.

J. Feature Matching

In this step, a correspondence is established between descriptors extracted from the two fingerprints. For each descriptor of the first image, we calculate the first and the second closest descriptors in the second image in terms of Euclidean distance. If the ratio between the two distances is below a given threshold, the first descriptor will be considered as corresponding feature.

K. Transformation Function Estimation

The final step in the correspondence process is the estimation of the geometric transformation function which may exist between the two fingerprints. We used the affine transformation model since the only differences that may exist between the samples are rotations and translations. However, due to the similar nature of minutiae between the fingerprints, we use the RANdom Sample Consensus (RANSAC) method [19] for outliers removal. Over several iterations, a candidate set of corresponding points is chosen each time and is used to estimate the transformation, and the number of inlier points is calculated. A couple of features is considered inliers if the distance between them is below a given threshold. At the end we keep the transformation with the largest number of inliers.

The number of inliers can serve as an acceptance criterion in the process of identification. If the number of inliers is above a specified value, we conclude that the two fingerprints are from the same person, thus giving a positive answer. Otherwise, we consider that the fingerprints are not from the same person, and the identification process gives a negative answer.

IV. System Parameters Selection

Throughout the proposed method, several parameters have been studied which can greatly influence the results of the identification process. So a parameter tuning phase has been carried out to give these parameters convenient values. We used the public FVC2002 database for this purpose. The studied parameters along with their values are shown in the table below.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block size</td>
<td>16x16</td>
</tr>
<tr>
<td>Ridge frequency minimum wavelength</td>
<td>3 pixels</td>
</tr>
<tr>
<td>Standard Deviation Threshold</td>
<td>Mean standard deviation across all image regions</td>
</tr>
<tr>
<td>SIFT Scale</td>
<td>4</td>
</tr>
<tr>
<td>Correspondence Threshold</td>
<td>0.87</td>
</tr>
</tbody>
</table>

V. Tests and Evaluation

For practical testing and parameters tuning of our system, we used the FVC competition databases (2002-2004) [20, 21]. These databases contain data collected by experts using several types of sensors. Each of them is divided into four databases from 4 different sensors. They contain fingerprint...
for a set of 100 individuals, with 8 samples each. A subset of 10 individuals is publicly available on their official sites, which is used in parameter tuning. The rest of the database is used for tests. For each individual in this set, we choose a sample as a reference, and for the remaining samples, we compare them with the reference. The comparison is done in the minutiae domain, represented by their SIFT descriptors, i.e. no transformation is applied to the extracted descriptors.

In terms of quantitative assessment, we used the two indicative measurements widely used in the evaluation of fingerprint identification systems: the False Acceptance Rate (FAR) and False Rejecting Rate (FRR). Both measures vary inversely by changing the acceptance criteria, which is dependent on the application field of the system. We use the Equal Error Rate (EER) as well, which is defined as the value where FAR = FRR.

The following table illustrates the EER values for the different test databases of the FVC2002, and ranks our methods among the published results (total of 31) according to [22]. We included only results from the A datasets (1 to 4) since B ones are only used for parameter tuning.

<table>
<thead>
<tr>
<th>Database</th>
<th>EER</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB1A</td>
<td>2.20%</td>
<td>14 (45%)</td>
</tr>
<tr>
<td>DB2A</td>
<td>0.97%</td>
<td>7 (19%)</td>
</tr>
<tr>
<td>DB3A</td>
<td>4.33%</td>
<td>12 (35%)</td>
</tr>
<tr>
<td>DB4A</td>
<td>2.35%</td>
<td>12 (35%)</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

The designed system thus described is a fully automatic fingerprint recognition system based on Gabor filter and SIFT descriptors. The main advantage of the proposed method is that no prior alignment of the scanned fingerprint images is required. The results may be further improved to obtain better EER values across all FVC databases, which requires dynamic parameter tuning according to the used sensor and the quality of the fingerprint scans.

References

Abstract—We propose a simple framework for unsupervised anomaly detection in aligned image sets, consisting of (i) feature extraction, (ii) PCA decomposition and partial reconstruction, and (iii) a dissimilarity measure comparing reconstructed to extracted features. The basic principle is explained using artificial datasets, and we show the effectiveness in a real-world scenario pertaining to sewer inspection images. We investigate the effectiveness of several features for this specific task, and show that concatenating several features results in superior performance. An analogy is also drawn to convolutional autoencoders and we compare to some simplistic renditions of such networks to our framework.

Anomaly detection, Principal component analysis, Convolutional autoencoders, Unsupervised learning, Image processing, Sewer asset management

I. INTRODUCTION

Sewer inspections need to be performed periodically to ensure performance is up to standards. Image and video data collected by a 'pipe inspection gadget' (PIG) is often manually inspected, leading to subjective, inconsistent, and unreliable deterioration and urgency ratings [1]. The SewerSense project [2] aims to automate such sewer inspections.

Anomaly detection (sometimes outlier or novelty detection) is a problem which aims to detect observations that do not conform to an expected pattern. In the context of image and video data, this relates to finding regions of interest (ROIs), portions of the video or images that have a different appearance and might be of interest. In the context of sewer inspections, automatic ROI detection is the first step of the SewerSense project. At a later stage in the project, classification of the found regions into a taxonomy of defect classes will be considered, but the intermediate result should aid the inspection process in itself.

In this work, we propose a three-part framework to detect anomalies in aligned image sets, such as static camera video or photographs, or registered images. The framework is based on principal component decomposition and partial reconstruction, but accounts for the fact that not all common elements in image sets can be accounted for by a linear model (such as PCA is) by first extracting possibly non-linear features from the image sets. We also foray into the field of deep learning and investigate the possibility of using convolutional autoencoders (CAEs) to fill the role of several parts of the framework.

We would like to emphasize that while this work originated from the need to automatically process sewer pipe images, no assumptions are made specific to this problem. The only requirement is that the images in a set are aligned, so other possible applications include video surveillance, autonomous vehicles and medical image processing.

II. PRELIMINARIES

A. Principal Component Analysis

Principal Component Analysis (PCA) has been around for over a century, after having been introduced in 1901 by Karl Pearson [3]. It is a popular tool in statistics, data science and many other scientific fields, used to reduce the dimensionality of data to facilitate data exploration and the use of algorithms that are sensitive to high dimensionality.

Given a dataset $X$, consisting of $N$ observations with $d$ features each, we express this as an $[N \times d]$ matrix. When using PCA for dimensionality reduction, a dimensionality $\theta \leq d$ is chosen and $X$ is projected on the first $\theta$ eigenvectors. The projected matrix $P$ retains as much variance as is possible in $\theta$ dimensions. This allows researchers to view high-dimensional data in two or three-dimensional plots, or employ algorithms that are not designed for high-dimensional data.

B. Anomaly Detection

In this study, we approach anomaly detection as an unsupervised learning problem. This means the algorithm will learn the structure present in the data, with the caveat that some parts of the data will not adhere to this structure. These anomalies have to be detected without influencing the
We propose a simple three-part framework to detect local anomalies in aligned image sets and videos, as shown in figure 1 and described in more detail in algorithm 1. The three parts are: (i) feature descriptors, (ii) PCA decomposition and partial reconstruction, (iii) a dissimilarity function to compare the PCA reconstructed feature to the extracted features.

a) Feature Descriptors: The choice of feature descriptor depends on the type of anomaly that has to be detected in the images. For example, to detect abnormal texture, we might use a feature that is known to work well in texture classification such as wavelet responses [7]. Or to detect motion in otherwise static camera images, we might calculate the difference between a frame and the previous frame at each position. The simplest choice is an identity function, i.e. the features are the original pixels in the image.

The reason for using feature descriptors instead of simply the images themselves stems from the fact that PCA is a linear model, and the resulting principal components will be combined linearly to reconstruct each image. Extracted features, unlike the images they came from, may have invariances to transformations that makes them more suited to compare images within a certain set than the raw pixel values would.

A feature can be used to describe an entire image, a specific location, or portions of an image, depending on the descriptor used. This determines how ‘localized’ the anomaly detection is. For example, we might calculate a locally windowed greyscale histogram, resulting in as many feature vectors as we have windows for each image in the set. We might want to detect entire images as being anomalous, or we might want to focus on specific regions within the image. When using localized features, we have the option to either treat all resulting feature vectors as if they came from the different locations, or portions of an image, depending on the descriptor used.

III. FRAMEWORK

Algorithm 1: Anomaly Detection Framework

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( x_i \leftarrow F(I_i) \quad \forall i \in [1, N] ) // Extract per-image features</td>
</tr>
<tr>
<td>2</td>
<td>( P \leftarrow \text{PCA}(X) ) // Decompose and project ( X ) onto its PCs</td>
</tr>
<tr>
<td>3</td>
<td>([p_{ij}] \leftarrow 0 \quad \forall i \in [1, N] \quad \forall j \in (\theta, d) ) // Discard contribution of low variance PCs</td>
</tr>
<tr>
<td>4</td>
<td>( \hat{X} \leftarrow \text{PCA}^{-1}(P) ) // Reconstruct to original feature space</td>
</tr>
<tr>
<td>5</td>
<td>( A_i \leftarrow \text{D}(x_i, \hat{x}_i) \quad \forall i \in [1, N] ) // Calculate per-image anomaly score</td>
</tr>
</tbody>
</table>

Output: Anomaly scores \( \{A_1, \ldots, A_n\} \)

Fig. 1. The proposed three-part anomaly detection framework.
A. Proof of Concept

To illustrate our method, we look at the MNIST reference dataset [8], consisting of 70,000 handwritten digits in greyscale images of dimensions \(28 \times 28\). We use the identity function as feature descriptor, so that the feature vector is identical to the pixel vector. This means our feature matrices are shaped \([70000 \times 784]\). When we apply PCA to the MNIST dataset, we obtain 784 principal components, which we can reshape into \(28 \times 28\) images for visual inspection (also known as \textit{eigenimages}), as shown in figure 2 for the first 9 principal components.
This illustrates the basic principle of the framework: the reconstruction error with a limited number of principal components can find anomalies in an image set of otherwise similar appearance. Although no feature descriptors were used for this simple example, the need for this will become clear in the next chapter.

IV. APPLICATION IN SEWER PIPE IMAGES

Dutch sewer engineering company vandervalk+degroot has provided us with a dataset of images from a front-facing camera on a PIG (pipe inspection gadget), from ten different streets within different municipalities in the Netherlands. These images are already spatially aligned, as the inspector has aligned the camera to the center of the pipe before starting. The images contain no labels or annotations though, so a method of verifying that the unsupervised method correctly finds anomalies is required. To this end, we selected two different subsets that are somewhat representative of all the sewer pipes from the different municipalities present in the datasets and hand-labeled 22 images from these sets.

The two subsets correspond to two different types of pipe: (1) smooth concrete and (2) more rough and textured agglomerate. Figure 4 shows an example of both. Henceforth, we will refer to these two image sets as ‘smooth’ and ‘coarse’. The image sets contain 684 and 698 images respectively.

The images are processed by the framework on a per-street basis. The reason for this is that the material used varies for different municipalities and date of installation, as will the effects of age. When using images from a single street, we can be reasonably certain that all images in such a set are of similar appearance, which means that anomalies are more easily detected, because we do not have to account for a possible multimodal distribution in appearance.

The images were divided into 324 patches of $[40 \times 40]$ pixels in the regions of the image that are in the focused portion of the images. Each patch in the 22 validation images was labeled as ‘anomaly’ or ‘normal’, in the context of the rest of the pipe. This includes both actual defects, such as discoloration as a result of leakage, as well as physical features that are simply less common than others, such as pipe joints and refuse.

All the images in the sets from which the labeled images originate are divided into the same 324 patches as the labeled images, and for each patch location features are extracted and PCA is applied to the feature vectors at a specific location. This means the framework is applied 324 times and each patch location across the images is treated as a separate image set. We construct an ROC curve by thresholding the anomaly scores at various levels and obtaining true and false positive rates for our labeled validation images. We report the area under the ROC curve (AUROC) as a measure of how well the resulting anomaly score performs.

Note that the parameter $\theta$, the cutoff value for the number of principal components to use in reconstruction, was chosen to maximize the AUROC. In our experiments, we found that the optimal value for $\theta$ corresponds to approximately 99% explained variance for the smooth image set and 95% explained variance for the coarse image set.

When using pixels as features and the mean absolute difference as a dissimilarity measure, we obtain results as shown in figure 5. The AUROC for the smooth set is 0.942, high enough to be of use, however, the AUROC for the coarse set, 0.774, is quite low. This means that if this method were to see use, human operators would have to sift through a large amount of false positives if the system were calibrated to an acceptable true positive rate.

The reason the framework performs less well on the coarse set when using pixels as features, is the texture present in the surface of the pipe in those images. The variance between pixel values is far greater than it is in the smooth set, where the entire pipe is more or less a single color, and as a result the image are difficult to capture in a linear model such as PCA.

A. Feature Extraction

To alleviate this issue, we extract features that we hope are more robust to textured images. The feature vectors are then decomposed, reconstructed and compared in the same way that the images would be, as shown in the framework in figure 1. In this section, we propose five higher-level features. An overview of each features’ invariances is given in table I.
TABLE I
OVERVIEW OF FEATURE EXTRACTORS INVARIANCES AND TYPICAL USES

<table>
<thead>
<tr>
<th>Feature</th>
<th>Invariances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pixel Values</td>
<td>None</td>
</tr>
<tr>
<td>Color Histogram</td>
<td>Translation, rotation, scaling</td>
</tr>
<tr>
<td>Fourier Transform</td>
<td>Translation (^1)</td>
</tr>
<tr>
<td>Histogram of Oriented Gradients</td>
<td>Translation, rotation</td>
</tr>
<tr>
<td>Local Binary Patterns</td>
<td>Translation, rotation</td>
</tr>
<tr>
<td>Homogeneous Texture Descriptor</td>
<td>Translation, rotation</td>
</tr>
</tbody>
</table>

1) Color Histograms: To illustrate the effectiveness of feature extraction, we choose a simplistic feature: a histogram of the pixel values. The 1600 values in each color channel of a patch are binned into 20 equally sized bins and concatenated to form a feature vector of length 60. These (in comparison) small vectors are decomposed into principal components and reconstructed with fewer than 60 principal components. The histogram is compared to the reconstructed histogram again by mean absolute difference. We see a slight improvement when using the histograms on the coarse set, an AUROC of 0.790, whereas performance on the smooth set is slightly worse with an AUROC of 0.942.

2) Fourier Transform: We perform a 2-dimensional Fourier transform on the [40 x 40] image patches, obtaining the frequency representation of the image patches. We discard the phase component by taking the absolute value and discard half the frequency plane because of symmetry. Again we decompose and try to reconstruct the feature vector, using the mean absolute difference as dissimilarity measure. The Fourier transform does not provide an improvement over using the pixel values, as we obtain an AUROC of 0.928 on the smooth set and 0.715 on the coarse set.

3) Histogram of Oriented Gradients: Often abbreviated as HOG, histograms of oriented gradients [10] describe an image by determining gradient directions at each pixel location, and binning these locally into histograms over a patch of specified size. It is often used for object recognition. It seems that this feature does not suit our purpose too well, as the AUROC for the smooth set becomes 0.886 and for the coarse set becomes 0.588. This can be explained by the fact that this feature is meant for object detection, and our images contain mostly texture.

4) Local Binary Patterns: Local binary patterns are a feature used to describe points as being edges or corners [11]. Each pixel is compared to its neighboring \( n \) pixels (usually \( n = 8 \)) and for each of these neighbors, it assign a 1 or 0 depending on whether the pixel has a higher value than that particular neighbor. The resulting 8-bit numbers are locally binned to summarize the texture of a cell as containing corners, edges, or otherwise. The concatenated histograms are used as a feature vector. We obtain AUROC of 0.865 for the smooth set and 0.705 for the coarse set.

\(^1\)After discarding phase component

5) Homogeneous Texture Descriptor: Part of the MPEG-7 multimedia description standard, homogeneous texture descriptors are shown to perform well on image retrieval tasks, even for images with much texture [12]. Simply put, the HTD features are comprised of (logarithmically scaled) mean values and standard deviations of the pixel values in a specified region.

B. Concatenating Feature Vectors

One of the strengths of the framework is that we can concatenate multiple feature vectors and the PCA reconstruction will still function identically. This allows us to combine the strengths of multiple feature types, and even combine these with the raw pixel values if we wish to do so.

After examining every possible permutation of the features previously described, we found that excluding the HOG and Fourier transform from the feature vector gave the best result on both image sets. Figure 6 shows the resulting ROC curves when we use the other high-level features described in this section, as well as the raw pixel values, giving us the highest AUROCs so far, 0.950 for the smooth set and 0.818 for the coarse set.

![ROC Curves](image_url)

Fig. 6. ROC curves from the anomaly detection framework on the validation set, using both pixel values and the high-level features described in this section (except for HOG and Fourier transform) combined as features to be analyzed by PCA.

V. CONVOLUTIONAL AUTOENCODER

An autoencoder is a neural network that tries to learn the identity function [13], and a convolutional autoencoder combines this with image filter learning. Analogous to our framework, this means we can learn the feature representation, perform non-linear dimensionality reduction (replacing the PCA) and reconstruct the input images. As we train this network on an image set, we should be similarly able to use it to detect anomalous regions by inspecting the difference image.

We designed a convolutional autoencoder consisting of:

- Input layer: [1040 x 1040] resolution
- Convolutional layer 1: 10 [20 x 20] filters, stride [10 x 10]
- Pooling layer 1: [2 x 2] max pooling, stride [2 x 2]
- Convolutional layer 2: 10 [20 x 20] filters, stride [10 x 10]
- Pooling layer 2: [2 x 2] max pooling, stride [2 x 2]
- Autoencoder: 1090 \( \rightarrow \) 845 \( \rightarrow \) 422 \( \rightarrow \) 845 \( \rightarrow \) 1690 units
- Unpooling layer 1: uniform, [2 x 2]
- Deconvolutional layer 1: Weights shared Conv. layer 2
• Unpooling layer 2: uniform, [2 × 2]
• Deconvolutional layer 2: Weights shared Conv. layer 1
• Output layer: [1040 × 1040] resolution

Using this network, trained on the same image sets, we obtained the following results: an AUROC of 0.946 on the smooth set and 0.714 on the coarse set, figure 7 shows the ROC curves. The results on the smooth set are rather similar to those obtained by the PCA framework, the AUROC results on the coarse set are noticeably worse.

![ROC curves from convolutional autoencoder.](image)

We expect that the reason for this reduced performance is the reconstruction of the full images. In the PCA framework, we are extracting features, decomposing and reconstructing these features, and comparing the reconstruction to the extracted features. In the convolutional autoencoder, we try to reconstruct the image itself out of necessity, as we do not know what the features should be. But this means that the reconstructed images are compared to the original images, instead of the reconstructed features to the original features.

The fact that the convolutional autoencoder has to reconstruct the original image, means it can’t learn features we might describe as ‘texture descriptors,’ as these are inherently rotation and translation independent, so reconstructing the original pixel values from such features would be impossible. But these are the types of features we expect (and confirmed for the PCA-based approach) to perform well, so the comparison is not entirely fair.

It should also be noted that the metaparameters of the network are far more difficult to optimize than the parameter θ our framework relies on, and a network better designed for this specific task may perform better.

VI. SUMMARY

We have proposed a framework for unsupervised anomaly detection in aligned image sets. Table II summarizes the results obtained by the different variants. We see that while raw pixel values perform quite well on the ‘smooth’ dataset, improvement can be made by combining different feature descriptors. For the ‘coarse’ dataset, the difference is larger, drastic improvements are made by combining features.

![Table II: Results for the methods and datasets described in this work.](image)

We conclude that our PCA-based approach, which could be considered a more ‘traditional’ statistical approach to computer vision using combinations of hand-crafted features, outperforms the more ‘modern’ convolutional autoencoder, but we must also admit that the comparison is not entirely fair as we are in one case reconstructing high-level features and in the other case pixel values.

REFERENCES


Survey of Issues with Text to Speech Synthesis of Multilingual Indian Texts

Suban Krishnamoorthy
Computer Science, Framingham State University, Framingham, MA 01701, USA, skrishnamoorthy@framingham.edu

Ching Y. Suen,
Director CENPARMI, Computer Science and Software Engineering, 1455 de Maisonneuve Blvd West, Suite EV3 403
Concordia University, Montreal, Quebec H3G 1M8, Canada, suen@cenparmi.concordia.ca

Abstract - Text to speech synthesis system should know the language to interpret the text and to generate speech. It becomes a challenge if the language has to be identified automatically by the system without any external input. India has multiple well developed languages in practice. Some of the languages use the same or similar scripts. Some of the languages have multiple scripts due to evolution over time. The objective of this paper is to describe some of the script related issues in Indian languages.

Keywords: Text to Speech Synthesis, Multilingual Text, Indian Languages, Indian Scripts, TTS Issues

I. INTRODUCTION

After decades of research and development of character readers, text to speech (TTS) systems and voice input systems are getting incorporated into various products available to the public. They are not perfect. When it comes to voice input systems one of the authors always had difficulty in one form or another with the systems used by the credit card companies, reservation systems and others. With the passage of time, they are getting better. A practical wearable integrated system with Image processing, character reading and recognition, TTS, voice recognition and synthesis is still not available at an affordable price and reliability.

With the usage of computing systems proliferating in non-English speaking multilingual countries, research work has been carried out in developing systems for their populations. India is a country of many well developed languages used by millions of people. This paper briefly reviews some of the research carried out in developing TTS systems and issues involved for Indian languages.

II. LANGUAGES OF INDIA

India is rich and diverse in languages and cultures. It has 122 major languages and 2371 dialects of which 22 languages are constitutionally recognized according to 2001 census [1]. Like many countries in the world, India has central (federal), state and local governments. States are mostly divided on language boundary. Each state has a local language spoken by majority of the people in that state. English is the common unifying language across states. Hindi is recognized as the national language by the central government. Table 1 gives a list of major languages and states where they are spoken. A good map of geographic territorial distribution of the languages can be found in Wikipedia [2]. For most part, each language is used by millions of people in India. The script for each local language is well-developed with rich literature created over hundreds and in some cases thousands of years. Each region is culturally different even though the same script is used for different languages.

III. MULTILINGUAL TEXTUAL COMMUNICATION

Most written communication with government, like government forms, will have one to three languages: (1) state and local government languages, (2) national language Hindi, and (3) English as common language across states and throughout the nation. Written communication between state governments and central government may have two languages: Hindi, and English. Within the state, it will be one or two languages: local state language and English.

Federal and state governments have created websites with multilingual texts. Figure 1a shows a sample multilingual text from Tamil Nadu tourism in Tamil and English in the state of Tamil Nadu in India. Figure 1b shows text in Hindi and English from Ministry of Tourism Government of India website. Figure 1c is part of the inauguration ceremony information released from the Radha Krishna Temple of Dallas located at Allen, Texas, USA, July 4-11, 2017. It was extracted from the email the author received from the temple. Figure 1d shows the press release of the opening of a Gurdwara Sahib (Sikh religious temple) in the town of Westborough in Massachusetts State, USA.
Table 1 List of major Languages and States of India [3]

<table>
<thead>
<tr>
<th>Language</th>
<th>States / Union Territory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assamese</td>
<td>Assam; Arunachal Pradesh;</td>
</tr>
<tr>
<td>Bengali</td>
<td>West Bengal; Assam; Tripura; Andaman and Nicobar Islands;</td>
</tr>
<tr>
<td>Dogri</td>
<td>Jammu and Kashmir</td>
</tr>
<tr>
<td>Gujarati</td>
<td>Gujarat; Dadra and Nagar Haveli; Daman and Diu</td>
</tr>
<tr>
<td>Hindi</td>
<td>Chatisgarh; Himachal Pradesh; Rajasthan; Chandigarh; Gujarat; Haryana; Jharkhand;</td>
</tr>
<tr>
<td></td>
<td>Uttar Pradesh; Dadra and Nagar Haveli; Delhi; Uttarakhand;</td>
</tr>
<tr>
<td></td>
<td>Madhya Pradesh; Bihar; Delhi; Andaman and Nicobar Islands;</td>
</tr>
<tr>
<td>Kannada</td>
<td>Karnataka</td>
</tr>
<tr>
<td>Khasi and Garo</td>
<td>Megalaya</td>
</tr>
<tr>
<td>Kashmiri</td>
<td>Jammu and Kashmir</td>
</tr>
<tr>
<td>Kokborok</td>
<td>Tripura</td>
</tr>
<tr>
<td>Konkani</td>
<td>Goa</td>
</tr>
<tr>
<td>Maithili</td>
<td>Bihar</td>
</tr>
<tr>
<td>Malayalam</td>
<td>Kerala; Lakshadweep; Puducherry</td>
</tr>
<tr>
<td>Manipuri</td>
<td>Manipur</td>
</tr>
<tr>
<td>Marathi</td>
<td>Maharashatra; Dadra and Nagar Haveli; Daman and Diu; Goa</td>
</tr>
<tr>
<td>Mizo</td>
<td>Mizoram</td>
</tr>
<tr>
<td>Nepali</td>
<td>Sikkim; West Bengal</td>
</tr>
<tr>
<td>Oriya</td>
<td>Orisa</td>
</tr>
<tr>
<td>Punjabi</td>
<td>Punjab; Haryana; Delhi; Chandigarh; Delhi;</td>
</tr>
<tr>
<td>Santhali</td>
<td>Jharkhand</td>
</tr>
<tr>
<td>Tamil</td>
<td>Tamil Nadu; Puducherry; Andaman and Nicobar Islands</td>
</tr>
<tr>
<td>Telugu</td>
<td>Andhra Pradesh; Telangana; Puducherry; Andaman and Nicobar Islands</td>
</tr>
<tr>
<td>Urdu</td>
<td>Jammu and Kashmir; Andhra Pradesh; Uttar Pradesh; Delhi;</td>
</tr>
</tbody>
</table>

Figure 1a. Tamil Nadu Tourism website. [4]

Figure 1b. Ministry of Tourism Government of India website. [5]

Figure 1c. Multilingual text. [6]

"The ceremony will fetch peace and happiness to all the people who dwell in that area, and who participate in the celebrations."
Excellent samples of multilingual webpages as well as some of the challenges of handling multilingual webpages in India can be found in [8]. In particular, it is worth to look at the character, “？” in the last webpage constructed using scripts of different Indian languages.

IV. LITERATURE REVIEW OF TTS SYSTEM

TTS systems and scanners were developed for several Indian languages. It is an active research area in India. A brief review of some of the literature is given in this section. Prabhsimran Singh and Amritpal Singh [9] mention a five phase based TTS for English to Punjabi conversion: (i) Text Analysis and Detection, (ii) Text Normalization and Linearization, (iii) Phonetic Analysis, (iv) Prosodic Modeling and Intonation, and (v) Acoustic Processing [10]. Raj et al discuss issues with font to akshra mapping, pronunciation rules for aksharas and text normalization [11].

TTS systems were developed for several Indian languages such as Hindi, Telugu, Kannada and others based on concatenative synthesis approach [30-34]. Sahu and Dhole system accepts input from keyboard and file and not printed text or grapheme displayed on computer monitor [12]. Kumar et al system [13] and Swathi et al system [14] accept Unicode as input for Telugu language. Padma et al have proposed a system to identify the language of a text in a multilingual document containing Kannada, Hindi and English by analyzing the top and bottom profiles of texts [15]. Some of the texts may have non-standard words and characters such as acronyms, abbreviations, phone numbers, names from different languages, currency, etc. They may have to be preprocessed. Kallimani, Srinivasa and Reddy propose a preprocessing scheme to convert non-standard Kannada words to Kannada words before feeding it to the TTS system for Kannada language using Jflex lexical analyzer [16]. An approach to build a speech database for Kannada TTS system using Praat software for Windows operating system was developed by Ravi and Patilkulkarni [17]. Architecture of a TTS for Hindi and software suitable for limited vocabulary, for website usage, and for hand-held device is supplied by Center for Development of Advance Computing System (C-DAC) [18]; it also conducts research in developing TTS for other Indian languages [19]. A hybrid TTS system for 12 Indian languages using concatenative and statistical parametric speech synthesis frameworks is proposed by Pradhan et al [20]; it is capable of performing semi-automatic and automatic segmentation of text. Segment duration of syllables plays an important role in natural sounding speech in synthesis. Shreekanth et al proposed a TTS system to predict segment duration considering the position of a syllable in a word for Hindi using neural networks [21].

TTS systems were developed for 13 Indian languages using both open source FETIVOX framework and HTS based engine under the Consortium Mode project with leadership from several Indian Institute of Technology (IIT) and other organizations [22]. TTS systems integrated with screen readers are available for Hindi, Bengali, Marathi, Tamil, Telugu and Malayalam running on Windows and Linux. TTS systems as browser plugin are also available for 8 Indian languages Hindi, Bengali, Marathi, Tamil, Telugu, Malayalam, Odia and Gujarati for Mozilla [23] and Chrome [24].

Voice quality of TTS systems can also be checked for several Indian languages [25].
V. ISSUES IN TTS OF INDIAN LANGUAGE TEXTS

There are several issues in the design and development of TTS systems for Indian languages. Some of the issues are listed below:

A. Multiple Fonts, Scripts and Languages

Multiple Fonts: English text has numerous fonts, some of which can be seen and used in word processors. Indian languages also have multiple fonts. Some of the texts may have mixed fonts in one document. Sample texts of multiple fonts for multiple Indian languages can be seen in [26]. These pose challenges in identifying characters and developing a TTS system.

Multiple Languages with Same Script: The same script is used for more than one Indian Languages. For example, Marathi and Hindi have the same script. Similarly, Telugu and Kanada have the same script as well.

Multiple Scripts for Same Language: For certain languages several scripts exit due to changes that happened over time. The writing media and political environment also played a role in the change. Table 2 shows the script change occurred in recent years in Tamil script for a few of its alphabets. First column contains the graphemes previously used and second column contains the current graphemes. Previous graphemes were used for hundreds of years and hence there is a rich printed literature using this script. Hence, TTS system should be able to recognize these two scripts.

Table 2. Previous and current scripts of Tamil alphabets.

<table>
<thead>
<tr>
<th>Letter</th>
<th>Previous</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>NNA</td>
<td>நேந்த</td>
<td>நந்த</td>
</tr>
<tr>
<td>RRA</td>
<td>ரேர்ர</td>
<td>ரார்</td>
</tr>
<tr>
<td>NA</td>
<td>நா</td>
<td>நா</td>
</tr>
<tr>
<td>NNAI</td>
<td>நேந்தை</td>
<td>நந்தை</td>
</tr>
<tr>
<td>LAI</td>
<td>லேஞ்</td>
<td>லாய்</td>
</tr>
<tr>
<td>LLAI</td>
<td>லலேஞ்</td>
<td>லலாய்</td>
</tr>
<tr>
<td>NAI</td>
<td>நேந்தை</td>
<td>நந்தை</td>
</tr>
</tbody>
</table>

B. Variation in Pronunciation and Language

In some languages, pronunciation of a word differs based on the region. For example, Tamil spoken in certain southern parts of Tamil Nadu differs from some of the northern parts. Same is true for English spoken by Texans and others in the USA. We have observed that English spoken in some parts of United Kingdom differs from other parts. These may have to be incorporated into the TTS system to be more user-friendly. In addition, in certain cases, there is wide Regional Variation in a Language, at different parts of the country.

C. Position of Speech and Morphology

Position of speech (POS) is the process of marking up a word in a text based on its definition and context. It is also called part of speech tagging (POST), grammatical tagging or word category disambiguation. In other words, it is the process of marking a word’s relationship with adjacent or related words in a phrase, sentence or paragraph [27]. Morphology is the study of inner structure of words and their forms in various uses and constructions. A significant number of words in a natural language are ambiguous. For example, a word can be a noun or verb depending on the context. Hence, POST is a challenging task in TTS. POS taggers and tag sets were developed for several languages such as Hindi, Bengali, Panjabi, Telugu and other Dravidian languages.

Based on POST and morphological information, same orthography can produce different pronunciation in certain languages like Bengali and Hindi. In Tamil language, the same letter can be pronounced differently depending on the context. For example the letter KA (க) is pronounced in different ways depending on the context like in KANN (கண்) and GANAM(கனம்). The letter க is represented by four different letters in Hindi as shown in Table 3.

Table 3. Letter KA in Tamil and Hindi

<table>
<thead>
<tr>
<th>Tamil</th>
<th>Hindi Ka</th>
<th>Hindi Kha</th>
<th>Hindi Ga</th>
<th>Hindi Oha</th>
</tr>
</thead>
<tbody>
<tr>
<td>க</td>
<td>க</td>
<td>஖</td>
<td>஗</td>
<td>஘</td>
</tr>
</tbody>
</table>

Antony and Soman presented a survey of literature of POST for Indian languages, which are based on different tag sets for each language [28].

D. Underlining

Underlining of text causes matra’s unreadable or difficult in certain languages by the OCR system [29]. In Figure 3, the first word is not underline. The same word underlined is shown below. The interference of underline is clearly visible in the first letter.
**E. Language Identification**

For speech synthesis, the language of the text has to be determined so that the Optical Character Recognition (OCR) system can recognize the text correctly. This is challenging and sometimes very difficult task.

*Use of non-Indian Language Words:* Non-syllabic English words are used in the syllabic Indian language texts. Language identification is required to properly pronounce these words, which is not present in the text alone.

**F. Voice and Emotion**

Whether the speech should be a female or a male voice cannot be identified from the text without an external input. Similarly, the emotion in the speech cannot be identified from the text.

**VI. TTS SYSTEMS TEST RESULTS**

Researchers achieved varying degrees of results with their TTS systems. Sahu and Dhole [12] and Singh [9] have achieved high language conversion and pronunciation efficiency. Reasonably natural sounding speech has been obtained for Telugu language [12-14]. The accuracy of Raj et al system [11] has been 99% for font conversion with contextual features and 68% to 98% for pronunciation model depending on the features and language used. The rule based system by Kallaimani et al [16] for Kannada language had limited success in the normalization of non-standard words like phone number, pin code, etc. Pradhan et al [20] hybrid system using semi-automatic segmentation approach obtained mean opinion score of 3.0 out of 5.0, and preference for their automatic segmentation over semi-automatic approach. Shreekanth et al [21] tested their system using two databases, (1) syllable position considering duration variation, and (2) without duration variation. The system with duration variation resulted in mean opinion score of 4.5 and 3.5 without duration variation out of 5; both scores were better than earlier systems.

**VII. CONCLUSION**

This paper illustrated multilingual textual communication in India. Several issues to be considered in the design and development of TTS systems for Indian languages are briefly described. The challenge of automatically identifying the language of the text is pointed out. It is an active area of research and development in India. There is lot more work to be done in this area to make it reliable, easy to use and make it widely available at affordable price, in particular for the vulnerable population of the society.

**ACKNOWLEDGMENT**

Thanks to Dr. Satish Penmasta for his review input.

**REFERENCES**


[6]. [info@radhakrishnatemple.net](mailto:info@radhakrishnatemple.net)


[8]. [https://www.w3.org/International/multilingualweb/madrid/slides/lata.pdf](https://www.w3.org/International/multilingualweb/madrid/slides/lata.pdf)


[24]. https://www.google.com/search?q=calligraphy+for+indian+languages&tbm=isch&imgil=YMG-0lluFO2SKM%253A%253BLq3s-9WSSLtcnM%253Bhttps%25253A%25252F%25252Fwww.pinterest.com%25252Fpin%25252F301881981252465608%25252F&source=iu&pf=m&fir=YMG-0lluFO2SKM%253A%253B5SltcnM%253B_&usg=__3qDBzrRX1AHLLXTCyGRezZhpHU0%3D&biw=1280&bih=668&ved=0ahUKEwjZx6Dn4Vx6ahVwolOKHQ9xDvAQvcIg&ei=YNyhWdmXAlit0gKP4rmADw#imgrc=r3pd333bDExE5M


[27]. https://www.w3.org/International/multilingualweb/madrid/slides/lata.pdf


Mobile Based Assistive Technologies for Authentication of Banknotes

Tamarafinide V. Dittimi, Ali K. Hnood, and Ching Y. Suen
Centre for Pattern Recognition and Machine Intelligence
Concordia University
Montreal, Canada
{t_dittim, a_alfraj, suen}@encs.concordia.ca

Abstract—This research presents a keypoint based descriptor system for recognition and authentication of banknotes captured using a mobile smartphone device that can assist visually impaired people. The system employs a client-server architecture where the mobile app built from Unity 3D captures the paper money. The image is preprocessed before transmitting to a MATLAB system that uses a three-level algorithm to extract and combine SIFT and SURF features. The feature set is reduced by Bag of Words to classify the bills using a stacked MLP and ECOC SVM method. Finally, the result is transmitted back to the smartphone and outputted as a pre-recorded audio output in addition to a text detailing the value and authenticity of the money. The system was tested on Naira notes; further analysis was conducted using US and Canadian banknotes and promising recognition and detection rate have been obtained.

Keywords— Banknote Recognition, Counterfeit Currency, Feature Extraction, Mobile Phone

I. INTRODUCTION

Technological advancement has made life easier but has its downside as criminals have also taken advantage of this progression too. Initially, only printing houses had access to the technology needed to produce fake banknotes but today with the aid of a personal computer and a good laser printer people can duplicate paper money easily in their residence. Banknote counterfeiting is one of the core crimes on earth. Mobile phones are rapidly becoming the central computer device in people’s lives due to its mobility. Most Mobile devices are equipped with cameras and possess computational capability. They provide an easy way for the blind and visually impaired people to obtain information about paper currencies [18].

Recognition system based on features have become a common practice in computer vision systems; and it is a substitute approach to the global method which uses the entire object pattern in the identification system. The main benefits of the feature-based approach over the global techniques are that some features preserve more class-specific information, while other features are alike across distinctive classes. Secondly, some features differ much less than the global pattern of the changes of viewpoint or scaling. Thirdly, the feature-based model is indifferent to partial obstructions [10]. Although several security features have been incorporated into banknotes counterfeitters have capitalized on the innovation in printing media technologies in reproducing fake paper currencies. Hence, there is a need to design systems to reassure and protect citizens of the authenticity of banknotes in circulation.

Visually impaired people are confronted with many challenges during their daily lives, especially conducting financial transactions. Although several traditional techniques exist that assist in identifying the different denominations of the banknotes. They also have their problems and limitations to its effectiveness. However, over time notes get old and the lines could get blurry making it difficult for visually impaired always accurately identify the different note denominations. Another option employed was differentiating banknote denomination by folding them into different distinct modes, while this works with bills already in possession it is still difficult to identify paper money received during daily transactions. Also, many currencies possess tactile marks used by visually disabled people in further differentiating the denomination, but notes get worn out during circulation which could affect the Braille marks. Although several security features have been incorporated into banknotes; counterfeitters have capitalized on the innovation in printing media technologies in reproducing fake paper currencies. Thus, it has become imperative to protect citizens and reassure them of the authenticity of banknotes in circulation [19]. The rest of this paper is organized as follows: Section 2 presents the literature survey. The SIFT, SURF, and COMBINED banknote reader are discussed in Section 3. While Section 4 shows the results of the implemented recognition and counterfeit detection system. Finally, in Section 5 conclusion and future work are presented.

II. LITERATURE SURVEY

Several methods have been designed for identifying and authenticating banknotes. A Canadian banknote reader has been developed, but the system only identified Canadian notes and its denomination by outputting a pre-recorded voice output informing the user of the note value [4]. Furthermore, a system for recognition of Sri Lankan currency was also designed by [9], the method adopted a unique linear transformation function for noise reduction, edge detection for fair representation of boundaries for new and old damaged notes, and a three-layer back propagation Neural Network for Classification.
Additionally, [20] also proposed an Indian currency recognition system based on the intrinsic features of banknote image; the research was conducted on the front side of 20 note images in denominations of Rs 10, 100, & 500. Similarly, [14] also proposed a general object reader called EyeRing that was trained to recognize the value of USA paper money. It was connected to a mobile phone through the Bluetooth facility and employed in identifying the value of US dollar although it had its limitations regarding the hardware and technical know-how needed to utilize the system. Also, a compact device for recognizing US Dollar denomination has been developed; the system is called the iBill Talking Banknote Identifier. It utilizes a combination of image processing and feature extraction method to distinguish the notes and then uses a pre-recorded voice output, sequence of tones and vibrations to inform the user of the banknote value. The limitation of this approach is it's usage on only US dollar bills and cannot be employed to detect counterfeit US notes [16].

In the same view, [19] recommended an image-based Bangladesh paper money recognition technique, the system extracted the Histogram of Gradient (HOG) feature from the watermark, Latent Image, and micro-printing of the notes and classified it using an SVM method. Finally, the approach had a 100% recognition rate. Although, it was only tested on the 500 BDT and 1000 BDT. Likewise, [23] developed an Ethiopian Birr notes recognition and detection approach, the paper money was captured using a scanner consisting of Ethiopian and non-Ethiopian notes including namely US dollar, Rand, Dirham, and Shilling. The Birr notes comprised of real and fake 100 Birr notes, 50 Birr notes, 10 Birr notes, 5 Birr notes and 1 Birr notes. The technique classified the bills into one of the Ethiopian denominations using the dominant color, distribution of dominant color, hue value and SURF matching. Furthermore, for authentication of the notes, the thin and wide strip on the banknotes was utilized. Finally, the system had an average denomination recognition rate of 90.42% with an average counterfeit rejection rate of 100%. On the other hand, the technique adopted for authenticating the banknotes is flawed as its only functional with high-quality notes that retain the number of objects in thin and wide strips.

Lastly, [5] advocated for a banknote recognition system that extracts the HOG feature, employed the Principal Component Analysis (PCA) in reducing the feature size and classified using Multi-Class SVM. The system experimented on the Nigerian Naira; and further analysis was conducted using the US Dollar, Canadian Dollar, and Euro notes. Finally, the system presented promising recognition rate and processing time. The drawback of the approach is it has only been tested on grayscale images; yet to apply local features like keypoint descriptors in recognition systems, recognize one banknote at a time and has not been extended to authenticate currencies.

Thus, this research presents a keypoint based descriptor system for recognition and authentication of note images captured using a mobile smartphone device that can assist visually impaired people. The method determines the value and the authenticity of the Nigerian Naira note by employing a client-server architecture where the mobile device captures or reads the image and then sends the copy to a server that uses a three-level algorithm to process the banknote image. Furthermore, the system returns the recognition result to the smartphone that outputs a pre-recorded audio in addition to a text detailing the value of the money and authenticity. The system is based on SIFT/SURF descriptor and a stacked MLP and ECOC SVM classifier to recognize and detect Naira notes irrespective of the orientation, lighting, and scale.

**III. PROPOSED METHODOLOGY**

The primary purpose of this research is the identification of original or forged banknotes and their values. A framework for authentication of paper currency based on the analysis and experimentation using SIFT and SURF keypoint based features classified using a stacked MLP and ECOC SVM is implemented. Figure 1 depicts the SIFT/SURF based Money Reader system. The system conducts the image acquisition and preprocessing on the mobile phone then the image is transmitted to a MATLAB backend for feature extraction and classification then the results are sent back to the cell phone for text and audio output of the denomination and its authenticity evaluation.

![Fig. 1. The Proposed SIFT/SURF Money Reader](Image)

**A. Image Acquisition**

The database constitutes Nigerian notes, US dollar and Canadian dollar bills with different years of issue, the images were acquired using a digital camera and was stored in JPEG format. The photos are in a 24-bit color image format with a resolution of 300 dpi (Dots Per Inch), and each database had two faces: obverse and reverse.

**B. Image Preprocessing**

After the banknote images have been digitized, the next step is preprocessing. It involves converting to grayscale then segmentation of the banknote by separating the bill from its background. The second level requires improving the contrast of the image by applying gamma correction and histogram equalization. Furthermore, using edge detection, the boundary coordinates of the banknote is cut out of the original image [3]. Lastly, the banknotes are then rescaled to ensure uniformity in note sizes. Figure 2 shows a sample pre-processed image transmitted from the mobile phone to a MATLAB backend server. In Figure 2, a) consists of full notes without occlusions, rotations and illumination changes, b) partial notes with lighting changes, distortions and clutters, and c) distorted banknotes with wrinkles and occlusions. Furthermore, the first row consisted of Naira notes, the second row comprised of US Dollar while the third row included Canadian Dollar.
C. Feature Extraction

Feature extraction is a critical phase in the banknote classification and detection process. To efficiently classify and authenticate a note; it is crucial to define its features and characteristics adequately [7]. This research employs a combination of two keypoint based methods; Speeded-Scale Invariant Feature Transform (SIFT) and Speeded up robust features (SURF) technique.

1) SIFT Feature Extraction: This method retrieves descriptors by fast-tracking the localization of the keypoints while preserving the property of the image [2]. It is a feature extraction method for detecting and extracting local and distinct features. The technique identifies keypoints in the image, including the blobs, edges, and corners in the banknote. The system locates the keypoints in the regions of the Naira note using equation 1. Kmli is the number of keypoints which is gotten by the summation of the keypoint identified in the jth region (j = 1, 2 ... m) of the banknote using the 1st descriptor in the ith number of classes of the naira. Each SIFT detector located several keypoints from different regions of the Naira note [13].

\[ K_{mli} = \sum_{j=1}^{m} K_{jl} \]  

For each key-point in the image, a feature descriptor is computed from the magnitude of the gradient and its relative direction of the pixels in the local neighborhood. The method groups the pixels into 16 x 16 sub-regions and subregion further bundled into 4 x 4 regions in the keypoint. Each sub-region is made up of an 8-dimensional weighted directional histogram, eight bins, each turned by 45° [11]. The number of feature vectors (FVi) in the note of ith classes can be gotten using equation 2 [6].

\[ FV_{i} = \sum_{l=1}^{n} K_{mli} \]  

2) SURF Feature Extraction Description: It is based on a local invariant fast feature point detector in addition to a distinctive invariant feature point detector, which is rotation and scale invariant. It employs Hessian based blob detection method in locating interest points and is created from the sum of Haar wavelet responses [13]. SURF computes the second order Gaussian filter for all keypoints (x,y) in an image, then applies a Hessian matrix Happrox at a scale σ, where the Happrox is given by equation 3.

\[ H_{approx} = \begin{bmatrix} \frac{L_{xx}(x,y)}{\sigma} & \frac{L_{xy}(x,y)}{\sigma} \\ \frac{L_{xy}(x,y)}{\sigma} & \frac{L_{yy}(x,y)}{\sigma} \end{bmatrix} \]  

Additionally, to facilitate the filtering based on the second order Gaussian filter, a convolution approach is employed using the different scales based on a square-like filter where \( \det(H_{approx}) \) denotes the highest value in the image pyramid [1]. Furthermore, to assign the best orientation to each keypoint, a sliding orientation window is then rotated around the keypoints, and the Haar wavelet transform is used to compute the weight of each pixel in the given window using equation 4.

\[ \det(H_{approx}) = D_{xx}D_{yy} - (wD_{xy})^2 \]  

To aggregate the vectors an area made up of 20x20 pixels is created and further grouped into 4x4 area called block, thus using a 3-dimensional vector to generate the feature descriptor [24]. Figure 3 presents the Preprocessed image with detected SIFT and SURF keypoint descriptors of Naira, USD and CAD paper money employed in this research.

3) Combined SIFT/SURF Descriptor: This research also implemented a feature combination technique based on the SIFT and SURF features. The approach detected SIFT and SURF features from the banknotes separately, computed the descriptors at each identified keypoint, normalized the SIFT and SURF descriptors and then concatenated them creating a COMBINED SIFT/SURF descriptor employed as feature vectors used in the recognition and detection of the bills.

D. Feature Reduction using Bag of Words

After the feature vector had been constructed, it was realized that the dimension was significant and would increase the processing time and the inclusion of unstable descriptors could also affect the recognition rate. Thus, K-Mean was introduced to create a cluster of related features; it is an unsupervised learning process that analyzes the distribution of samples in the feature space [10]. The technique groups random samples from the SIFT, SIFT and COMBINED feature vector descriptors; that is the value of K is set to the number of related feature groups. The K-mean algorithm is then run till an average result is reached. Finally, Bag of Words model is then utilized to represent the clustered features by computing the visual words to reduce the number of the feature vectors [15]. For experimental evaluation, additional analysis was conducted using Information Gain (IG) and Principal Component Analysis (PCA) in reducing the feature set.
E. Classification

Classification involves determining the similarities and dissimilarities of a representative dataset to find quite smaller data dimensionality of the represented data [17]. For this research, the system employed stacking in Combining a multilayer perceptron (MLP) neural network with Error-Correcting Output Codes Support Vector Machine (ECOC SVM). Stacking is an approach used in training a learning algorithm to unite the predictions of different classifiers. MLP is an interconnected collection of nodes where each node denotes artificial node and arrow signifying a connection from the output of one node to the input of the next layer. The neural network comprises of the following layers: input, hidden and output layers [9]. On the other hand, the ECOC SVM technique compares all classes against each other and builds a binary classifier to distinguish each pair of class while ignoring the remaining classes. When classifying an unknown image to a particular class, the classifier conducts a vote, and the class with the majority is chosen as the best choice to assign the banknote [8, 20]. First, the dataset is split into training and testing sets, MLP and ECOC are employed in training and testing the dataset. Furthermore, a combination mechanism based on a single layer linear regression model is utilized to make a final prediction using the results of the MLP and ECOC SVM as inputs and the correct responses as the outputs.

IV. DISCUSSION

In this paper, we bypass the use of conventional programming languages like C++ and Java or platform dependent application such as Android or iOS and their relating toolbox. We instead focus on developing the mobile phone front end of the system using a Multiplatform framework called Unity 3D. It utilizes JavaScript and C# as its scripting language and can build executable applications that can run on the Web, Windows, Android, Flash, and iOS. Additionally, due to its scalability, ease of development and powerful processing capability, it is widely sorted out for the development of mobile applications [22]. While on the backend, MATLAB 9.2 R2017a was employed in the implementation of the modules. The decision to use MATLAB was because of the extensive mathematical functionality of this package as it enables inexperienced users to work with the toolbox in addition to providing automated and batch standardization of analyses and statistical tools for data representation [12]. The system was tested on an iPhone 6 and a Samsung S6 with a minimum camera resolution of 8 megapixels where each phone had a processing speed of 1.4 GHz.

A. Database

In Table 1, the details of the dataset are presented. It consisted of 3110 Naira notes, 1943 US Dollar, and 1733 Canadian Dollar further divided into genuine and counterfeit bills. The dataset of each currency is divided into original and fake paper currencies. Furthermore, the Naira notes have eight classes, which are ₦5, ₦10, ₦20, ₦50, ₦100, ₦200, ₦500, and ₦1000. While the US dollar has six categories, namely $1, $5, $10, $20, $50, and $100. Lastly, the Canadian Dollar has five groups, namely $5, $10, $20, $50, and $100.

B. Banknote Reader

This research presents a banknote recognition method based on the SIFT/SURF combination approach that classifies and authenticates banknotes using a stacked MLP and ECOC SVM. For further evaluation, another classification approach was applied to determine the robustness of the system. The SIFT and SURF keypoint descriptors have been utilized on computers, but due to its computational complexity, it cannot be fully migrated to a smartphone. Thus, we present a hybrid method that preprocesses the image on the smartphone before sending it to a MATLAB server for SIFT and SURF feature extraction and classification using a stacked MLP and ECOC SVM.

Table 2 shows the feature reduction techniques and results employed in this paper, the SIFT and SURF method presented an average feature set of 513,434 and 342,918 vectors; after feature selection, IG had 480 features, PCA gave 540 vectors, and BOW introduced 500 features respectively. While with the COMBINED approach, the concatenated descriptors had 856,352 vectors, IG produced 620 features, PCA gave 780 vectors, and lastly, BOW presented 650 features. It was evident that we could achieve a feature reduction ratio near 90% along with a decrease in classification time (about 55%) comparing with the non-optimized features.

![Accuracy Comparison between feature reduction method and the Classification technique employed on the Naira note Dataset.](image)
As can be seen from Figure 4, a comparative analysis of the feature reduction approach on four classification methods on the Naira notes. It was introduced to reduce the processing time in addition to diminishing the feature set and equalized the feature vector amongst all classes and database. The system was classified without feature reduction implemented, and for further evaluation, Information Gain (IG), Principal Component Analysis (PCA) and Bag of Words (BOW) were employed to reduce the SIFT, SURF, and COMBINED descriptors. However, there wasn’t any significant difference in accuracy between the feature vectors without feature reduction and vectors after BOW feature reduction. Moreover, it can be deduced that the proposed MLP-ECOCSVM method outperformed Naïve Bayes Multinomial (NBM), MLP, and ECOCSVM. Also, the feature reduction and classification approach that presented the best accuracy was SIFT/SURF combination (BOW) and MLP-ECOCSVM while SIFT/SURF (IG) and KNN had the lowest accuracy. Lastly, SURF technique outperformed SIFT with or without feature reduction using IG, PCA, and BOW.

Figure 5 displays the result of the performance criteria utilized in this research. The Accuracy, Precision, Recall, and F-Measure of the proposed COMBINED-BOW and MLP-ECOCSVM methods on Naira, USD, and CAD Database are shown. The result proved that the keypoint based feature (SIFT and SURF Concatenation) method employed in this research could be useful in practical banknote recognition systems. However, the Naira note database presented the highest Precision, Recall, Accuracy, and F-Measure while Canadian Dollar bills had the lowest results.

In Figure 6, the recognition time is computed for SIFT, SURF and COMBINED using IG, PCA and BOW feature reduction methods using the proposed approach on the Naira, USD, and CAD databases. From the result, the SIFT and SURF outperformed the combination across all feature reduction methods; although the difference was minimal. Additionally, PCA and IG could reduce the recognition time; but BOW had the best recognition time. However, the COMBINED had the longest recognition time due to the size of the feature vectors. Lastly, the Canadian dollar database produced the best processing time when Bag of words reduction is employed.

On the other hand, Table 3 portrays the Detection Rate (DR) and Error Rate (ER) of SIFT, SURF and the COMBINED approach using several feature reduction methods on MLP-ECOCSVM classification technique. From the result, although COMBINED (BOW) gave the best result, SURF(BOW) outperformed the other Feature reduction approaches and presented competitive detection and error rates. Finally, SIFT(IG) had the lowest detection and error rates across all datasets utilized in this research.

Moreover, the experimental evaluation shows that our proposed approach presented a competitive Recognition rate and lowest Error Rate irrespective of the quality of the fake banknote dataset. Several other feature extractor/descriptor combinations have also been experimented in this research: FAST, ORB, and FREAK. However, they produced a lower recognition/detection rate, extended the processing time or presented insignificant improvement in the system accuracy. Lastly, to reduce the processing time of the system an Apache server using a PHP 5 and a MATLAB automated service is set up, it allows an instance of the MATLAB script to be always in memory thus eliminating boot time delays. Furthermore, the method communicates with the mobile phone using a COMECSVM method.

![Performance Criteria](image)

![Recognition Time (sec)](image)

**Figure 5.** Performance Criteria the proposed COMBINED (BOW) and Multi-Class SVM method on Naira, USD, and CAD Database.

**Figure 6.** Recognition Time (sec) of Naira, USD, and CAD

**TABLE III.** Detection and Error Rates of Classifiers with the Multi-class SVM

<table>
<thead>
<tr>
<th>Classifier (%)</th>
<th>Naira Note</th>
<th>US Dollar</th>
<th>Canadian Dollar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DR</td>
<td>ER</td>
<td>DR</td>
</tr>
<tr>
<td>SIFT</td>
<td>93.00</td>
<td>7.00</td>
<td>92.56</td>
</tr>
<tr>
<td>SIFT(IG)</td>
<td>85.44</td>
<td>14.56</td>
<td>84.85</td>
</tr>
<tr>
<td>SIFT(PCA)</td>
<td>90.00</td>
<td>10.00</td>
<td>87.71</td>
</tr>
<tr>
<td>SIFT(BOW)</td>
<td>93.77</td>
<td>6.23</td>
<td>93.00</td>
</tr>
<tr>
<td>SURF</td>
<td>94.60</td>
<td>5.4</td>
<td>94.46</td>
</tr>
<tr>
<td>SURF(IG)</td>
<td>87.89</td>
<td>11.11</td>
<td>86.96</td>
</tr>
<tr>
<td>SURF(PCA)</td>
<td>92.00</td>
<td>7.00</td>
<td>90.82</td>
</tr>
<tr>
<td>SURF(BOW)</td>
<td>94.45</td>
<td>5.55</td>
<td>93.75</td>
</tr>
<tr>
<td>COMBINED</td>
<td>95.11</td>
<td>4.89</td>
<td>94.58</td>
</tr>
<tr>
<td>COMBINED(IG)</td>
<td>92.34</td>
<td>7.66</td>
<td>93.45</td>
</tr>
<tr>
<td>COMBINED(PCA)</td>
<td>94.23</td>
<td>5.77</td>
<td>94.02</td>
</tr>
<tr>
<td>COMBINED (BOW)</td>
<td><strong>95.45</strong></td>
<td><strong>4.55</strong></td>
<td><strong>94.75</strong></td>
</tr>
</tbody>
</table>
protocol through PHP also making it easy to access the server. In the same view, the smartphone, on the other hand, utilizes a POST request approach in sending banknote images to the server while the server streams back binary data to the mobile phone after recognition/detection of the banknote.

V. CONCLUSION

This research studied the extracted features to determine its usefulness in determining the authenticity of a banknote after classifying the denomination of the bills. It further evaluated the robustness of the features by taking into consideration image variation like rotation, scaling, lighting and partial occlusions. The mobile-based system included an image acquisition and processing module built using Unity 3D, in addition to a voice output stating the denomination and confirming the authenticity of the naira note. Furthermore, the Backend MATLAB Server had the SIFT, SURF and COMBINED descriptor extraction and a proposed classifier algorithm based on a stacked MLP-ECOC-SVM classification and an authentication module. The system uses a three-tier approach; the first level identifies the denomination of the banknote while the second level determines the authenticity of the currency and the last stage rejects paper currencies that are not Naira, USD, and CAD notes. The proposed method employed a 10-fold Cross Validation and was tested on a randomized database sample comprising of counterfeit and real Naira notes in addition to US and Canadian dollar. The proposed system achieved a recognition accuracy of 97.18% and a detection rate of 95.45%.

ACKNOWLEDGMENT

The authors would like to thank the Center for Pattern Recognition and Machine Intelligence (CENPARMI), the Natural Sciences and Engineering Research Council of Canada (NSERC) and Concordia University for their contributions to this research.

REFERENCES


A Robust Palmprint Matching Method

Yea-Shuan Huang  Charles Djimy Slot
Department of Computer Science & Information Engineering
Chung-Hua University
Hsinchu, Taiwan
e-mail: yeashuan@chu.edu.tw

Abstract—This paper proposes a robust palmprint matching algorithm which mainly consists of five processing steps. First, an input palmprint image is preprocessed so that noise can be decreased and both image contrast and image size can be normalized. Secondly, a CMFRAT (Contrast Modified Finite Radon Transform) algorithm is used to construct several direction-oriented masks and each mask is applied to strengthen the palmline response of a certain direction. Thirdly, an Omni-bearing palmline response image and a palmcode image are constructed in which each pixel value is set by selecting the maximal palmline response among all directional palmline responses at that pixel. Fourthly, ridge lines are extracted from the Omni-bearing palmline response image by using two processing techniques: “Non-Maxima Suppression” and “Hysteresis and Connectivity Analysis”, and only long enough ridge lines are considered to be principal ridge lines and the rest ridge lines are considered to be wrinkles or noise which will be removed. Fifthly, a block-based Chi-Square distance is used to match two palmprint images by using their palmcode as features with two kinds of pixel weights. Experiments on the public PolyU 2.0 Multispectral Palmprint Database, the proposed method has shown excellent performances for both palmprint recognition and palmprint verification.

I. INTRODUCTION

In the past, the main means to authenticate a user’s identity are based on identity card, passport, key, password, and other magnetic cards. However, these methods have many obvious shortcomings, such as they are easily copied, lost, stolen or forgotten, so inconveniences could occur when using them. Interestingly, there once was a report that 25% of people wrote their passwords on their ATM cards, and when they lost their cards, then the traditional authentication mechanism will lose any protection function. In fact, these mechanisms cannot distinguish between the genuine users and the fake ones as long as they get the necessarily physical or digital authentication objects. Thus, biometrics have been proposed to solve this problem which identifies a user’s authentication with his/her inherent features (such as eye, face, fingerprint, palmprint, voice, etc.). In recent years, because (1) a substantial decline in the price of biometric sensors, (2) recognition technology becomes more mature, (3) the public gradually know the advantages of biometrics and are willing to cooperatively use them, and (4) the privacy protection laws have been increasingly reinforced, biometric authentication technologies and related products are popularly recognized and accepted as the most promising authentication method.

In general, the common biometric authentication technologies include face recognition, facial thermal recognition, fingerprint recognition, hand shape recognition, retina recognition, iris recognition and speaker authentication. Although fingerprint recognition currently shares the largest market, but because it requires an additional fingerprint sensor chip which costs about US $10 per piece, finger print authentication indeed makes a significant cost burden to a general portable electronic devices (such as mobile phones, iPad and laptop). Therefore, if a camera which is commonly installed in a portable device nowadays as a standard component can be used to capture images, and the user’s identity can be determined through computer vision techniques, it will save considerable hardware cost and consequently a huge market opportunity can be created. So far, for the biometrics based on images taken from cameras, face recognition is most acceptable and recognized, but because of its high complexity, it is still difficult to meet the practical needs for common daily applications; both retina and iris recognition require to project low-power infrared beams into eyes, this makes a lot of people uncomfortable to use them; hand shape identification although has a convenient way to use, it has much less amount of information so that it is unable to achieve a high authentication accuracy. Given the shortcomings of the above image-based methods, identity authentication based on palmprint images becomes a feasible way to develop.

In the field of palmprint identification, Prof. David Zhang of Hong Kong Polytechnic University is the most famous researcher. In 1998, he published the first global palmprint identification system [1] by using two Gabor filters to detect the positions of the palms and a normalized Hamming distance for comparison. Prof. Zhang led a research team to establish the largest palm image database (i.e. PolyU palmprint database [2]), and continued to publish a lot of palmprint identification papers. In 1999, he presented a palm line extraction method [3]; in 2003, he presented a subspace palmprint recognition technology [4]; in 2004, he presented a palmprint classification method by extracting the three principal palm lines (life line, wisdom line and emotional line) [5]; in 2006, he presented a feature-
level fusion technique by combining different oriented Gabor filters [6]; in 2008, he presented a novel palm line extraction and recognition method by a Modified Finite Radon Transform [7]; in 2009, he presented a 3D palmprint recognition method [8]; in 2010, he presented a multispectral palmprint recognition method by taking images with four different electromagnetic waves (red, blue, green and infrared) and integrating the four individual recognition results to obtain the final recognition [9,10]; in 2011, he presented a tree structure for recognizing a large-scale palmprint images [11]; in 2012, he presented a spectral selection mechanism to select the best spectral among multispectral [12].

This paper consists of four sections. Section 2 introduces the major processing modules of the proposed method in detail. Section 3 describes experiments and performs analysis of the experimental results. Finally, Section 4 draws a conclusion of this paper.

II. THE PROPOSED METHOD FRAMEWORK

The proposed method mainly contains three processing modules, preprocessing, feature extraction and matching as shown in Fig. 1, and the three modules are described in details as below.

A. Preprocessing

The input image is first smoothed out to remove potential noise and normalized to a predefined size. Then it is operated by an image enhancement algorithm so that its intensity values can be stretched out to a desired range of values. Often, it increases the contrast of the image as shown in Fig. 2. Let \(a\) and \(b\) be respectively the lowest and highest pixel values present in the smoothed input image, \(c\) and \(d\) be respectively the desired lowest and highest pixel values of the enhanced image, and \(P_{in}\) and \(P_{out}\) be the respectively the pixel intensity value of one pixel before and after the enhancement operation. Then

\[
P_{out} = (P_{in} - a) \left( \frac{d - c}{b - a} \right) + c
\]

(1)

Fig. 1: Framework of the proposed method.

Fig. 2: (a) original palm image, (b) palm image after preprocessing.

B. Feature Extraction

There are many features in a palmprint image that can be extracted for matching. Principal lines, wrinkles, ridges, minutiae points, singular points, and textures are regarded as useful features for palmprint pattern representation. Our method makes use of the principal lines pixel as a weight vector and compute block histogram of the palm texture to perform matching tasks.

In this phase of the process, the above-mentioned features are enhanced from the preprocessed input image using a special filter called CMFRAT which stands for Contrast Modified Finite Radon Transform. Palmline strengths, direction codes and principal palm lines are extracted from the enhanced palm image for later matching computation.

1) CMFRAT

While the preprocessed image produces consistent palm line texture in various lighting environments, it enhances simultaneously both the principal palm lines and wrinkles. However, because wrinkles are unstable, it is necessary to further distinguish them so that principal palm lines can be extracted and wrinkles can be removed. Basically, compared with wrinkles, principal palm lines are longer, straighter and more evident. With these understandings, MFRAT (Modified Finite Radon Transform) [7] has been used to strengthen the main palm line signals. However, through observation and experiments, it shows that palmprint images captured by mobile devices are often affected by ambient light sources, and because a palmprint image indeed is a 3D signal, sometimes there exists reflections near the principal palm lines which results in dissatisfied processed results of MFRAT. The CMFRAT (Contrast Modified Finite Radon Transform) [1], a variation of the MFRAT, which was proposed by Huang [2], is a powerful tool which can even adopt reflection to produce a palm line response of each pixel and makes the principal palm line extraction much easier for later processing. A simplified explanation of CMFRAT is described below.

A CMRFAT mask is a square mask to enhance palmline response at a certain direction, and it contains three kind of cells, positive cells, negative cells and null cells. Basically, a CMFRAT mask sums up the intensities of all positive-cell pixels and subtracts the intensities of all negative-cell pixels of this mask. Let \(P_k\) and \(N_k\) be respectively the positive-cell set and the negative-cell set of the \(k\)-direction CMFRAT mask, and \(S_{x_0,y_0}[k]\) be the palmline of the \(k\)-direction CMFRAT mask at pixel \((x_0,y_0)\). That is

\[
S_{x_0,y_0} = \sum_{(x,y) \in P_k} P(x,y) - \sum_{(x,y) \in N_k} N(x,y)
\]
where $x'$ and $y'$ are respectively the shifts to the center of the mask in the $x$ and $y$ axes. Then the palmline response $E(x_0, y_0)$ of pixel $(x_0, y_0)$ becomes:

$$E(x_0, y_0) = \max_{k \in K} S_{x_0,y_0}[k]$$

where $K$ is the set of all the CMFRAT directions. The pixel-direction $D(x_0, y_0)$ of pixel $(x_0, y_0)$ can be determined as

$$D(x_0, y_0) = \arg \max_{k \in K} S_{x_0,y_0}[k]$$

In this paper, $E$ is referred to as an Omni-bearing palmline response image and $D$ is referred to as a palrcode image. Fig.3 shows the CMFRAT masks of nine different directions, each mask contains three kind of cells, positive cells (in gray), negative cells (in dark) and null cells (in white). Fig.4 shows an input palmprint image, its Omni-bearing palmline response image $E$ and its palrcode image $D$ generated from the CMFRAT operation.

2) Ridge Line Extraction

The enhanced palm lines in the Omni-bearing palmline response image $E$ often have different widths which make them unstable because they are prone to be affected by ambient light. Therefore, only palmline ridges are used to represent palm lines, and they are extracted from Omni-bearing response images by using “Non-Maximum Suppression” and “Hysteresis and Connectivity Analysis”.

- Non-Maximum Suppression

First, we compute $g_m(x, y)$ and $g_{dir}(x, y)$ which are respectively the gradient magnitude and the gradient direction of each pixel $(x, y)$ in the Omni-bearing palmline response image. Then, each $g_m(x, y)$ is compared with its two neighbors along the gradient direction $g_{dir}(x, y)$. If it is greater than both of its two neighbors, $g_m(x, y)$ keeps its value, otherwise $g_m(x, y)$ is set to zero.

- Hysteresis and Connectivity Analysis

We threshold the previous result by two different thresholds $t_{low}$ and $t_{high}$, with $t_{low} < t_{high}$, to obtain two binary images $E_{low}$ and $E_{high}$. Ridge pixels are formed into ridge lines respectively in $E_{low}$ and $E_{high}$ by a connected component analysis algorithm. Although most ridge lines in $E_{high}$ correspond to parts of principal palm lines, they probably should be merged in order to obtain more complete principal palm lines. Let $A$ and $B$ be two ridge lines in $E_{high}$ which belong to the same palmline, $P$ be one end point of $A$ and $Q$ be one end point of $B$. Ridge lines $A$ and $B$ are linked into one ridge line if and only if there is a ridge line $C$ in $E_{low}$ that goes though $P$ and $Q$.

Fig. 5 shows an example of ridge line extraction, Fig.5(a) is an Omni-bearing palmline response image, Fig.5(b) the $E_{high}$ image, Fig.5(c) the $E_{low}$ image and Fig.5(d) is the principal ridge lines extracted from Fig.5(a).

C. Matching Algorithm

For comparing two palmprint images $E$ and $F$, a block-based matching algorithm is designed which is summarized as follows. First, the palrcode images and the principal ridge lines of images $E$ and $F$ are computed; secondly, a 2D weight array is formed by using the principal ridge lines of
images \( E \) and \( F \); thirdly the palmcode images are divided into \( B \) blocks and the histogram of each block is computed; and finally, a Chi-Square Distance of the computed histograms of \( E \) and \( F \) is calculated to indicate the difference of the two palmprint images. In the following, 2D weight array, block histogram and palmprint distance measurement will be described in detail.

1) 2D Weight array

Because there are two kinds of pixels (principal ridge-line pixels PPR and non-principal ridge-line pixels NPPR) in one palm image, two different weights should be assigned to individual pixels to present their contributions in pattern matching. Conceptually, PPRs have more contribution on matching so that they should have a larger weight than those of NPPRs. With this understanding, for matching two images \( E \) and \( F \), a 2D weight array with the same size as \( E \) and \( F \) is constructed and each cell of the 2D array corresponds to one pixel in a palm image. The 2D array contains two kinds of cells, PPR cells and NPPR cells. A PPR cell denotes that its corresponding pixel in either \( E \) or \( F \) belongs to a principal ridge line, and a NPPR cell denotes that its corresponding pixels in both \( E \) and \( F \) do not belong to a principal ridge line. The value of each PPR cell is set to \( V \) (\( V > 1 \)) and the value of each NPPR cell is set to 1. For example, in this paper a weight of 3 is assigned to each PPR cell and a weight of 1 to each NPPR cell. Let \( w \) be a 2D array with the size of \( E \), \( V \) be the larger weight, and \( R \) and \( S \) be respectively the principal ridge-line image computed from images \( E \) and \( F \). Then the weight of cell \( j \) is assigned to be

\[
w(j) = (V - 1) \times [R(j) \cup S(j)] + 1
\]

(5)

2) Block histogram

Each palmcode image is divided into \( B \) blocks and a histogram of each block is produced as palm features. The histogram of a block is a vector \( H \) with \( N \) bins and each bin denotes the accumulation strength in a certain direction of this block, and \( H \) is computed as

\[
H_n^b = \sum_{j \in R(b)} G(D(j), k_n) \cdot w(j)
\]

(6)

where \( D \) is a palmcode image, \( k_n \) is the \( n \)th direction of the CMFRAT direction set, \( j \) is the index of pixels belonging to block \( b \), \( R(b) \) is the region of block \( b \), \( D(j) \) is the pixel value at index \( j \) of \( D \), \( w(j) \) is the weight of \( D(j) \), and

\[
G(D(j), k_n) = \begin{cases} 1, & \text{if } D(j) = k_n; \\ 0, & \text{otherwise}. \end{cases}
\]

(7)

3) Palmcode Distance Measurement

Let \( H_n^b(D) \) be the value of the \( n \)th bin of histogram of block \( b \) of palmcode image \( D \). Then a Chi-Square Distance is used to calculate the difference of palmcode images \( A \) and \( C \), that is:

\[
D(A, C) = \frac{1}{B} \sum_{b=1}^{B} \sum_{n=1}^{N} \frac{(H_n^b(A) - H_n^b(C))^2}{W_n^b}
\]

(8)

and \( W_n^b \) is computed as

\[
W_n^b = \sum_{j \in R(b)} w(j)
\]

(9)

III. EXPERIMENTAL RESULTS

A. PolyU 2.0 Palmprint Database

The images of PolyU 2.0 Multispectral Palmprint Database were captured under blue, green, red and near-infrared (NIR) illuminations. Those images were collected from 250 volunteers, including 195 males and 55 females. The age distribution is from 20 to 60 years old. Samples were collected in two separate sessions. In each session, the subject was asked to provide 6 images for each palm. Therefore, 24 images of each illumination from 2 palms were collected from each subject. In total, the database contains 6,000 images from 500 different palms for one illumination. The average time interval between the first and the second sessions was about 9 days. Each of the 500 different palms is called a palm class.

B. Compared methods

The performances of three different methods on the PolyU 2.0 palmprint database are compared. The first is a method proposed in [1], the second \( P_{\text{different weights}} \) is the proposed method with different weights on PPRs and NPPRs, and the third \( P_{\text{equal weight}} \) is a variation of the proposed method by assigning an equal weight to each pixel in the palmcode image. For the recognition task summarized in table 1, as the author of [1] didn’t provide any experiment results, only the performances of \( P_{\text{equal weight}} \) and \( P_{\text{different weights}} \) are compared. For the verification task, the performances of all the above-mentioned methods are listed in table 2.

C. Recognition

In our recognition experiment, the database is divided into 2 subsets, a training subset and a test subset. The training subset contains all 3000 images taken in the first session and the test subset contains all 3000 images taken in the second session. Each image in the test subset is matched against every image in the training subset.

Suppose \( I \) denote a test palmprint image and it’s class is \( k_{\text{real}} \), \( T_i \) denote the \( i \)th training palmprint image, \( M \) is the total number of the training palmprint images and \( M \) is 3000 here. Then image \( I \) is recognized as class \( k_{\text{recog}} \) if

\[
i' = \arg \min_{i \in \{1, \ldots, M\}} D(I, T_i)
\]

(10)
and

\[ k_{\text{recog}} = \text{class index of } T_i \]  

A recognition operation is considered correct when \( k_{\text{recog}} \) is equal to \( k_{\text{real}} \), otherwise it is considered incorrect. A recognition experiment was performed individually on red, green and blue illuminations from the above database. The results are shown in Table 1.

<table>
<thead>
<tr>
<th>Recognition (3000 images)</th>
<th>( P(\text{equal weights}) )</th>
<th>( P(\text{different weights}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Illumination</td>
<td>Correct</td>
<td>Incorrect</td>
</tr>
<tr>
<td>Red</td>
<td>2981 (99.37%)</td>
<td>19 (0.63%)</td>
</tr>
<tr>
<td>Green</td>
<td>2993 (99.77%)</td>
<td>7 (0.23%)</td>
</tr>
<tr>
<td>Blue</td>
<td>2997 (99.90%)</td>
<td>3 (0.10%)</td>
</tr>
<tr>
<td>RGB</td>
<td>2994 (99.80%)</td>
<td>6 (0.20%)</td>
</tr>
</tbody>
</table>

Table 1. Recognition accuracy of different palmprint illuminations.

D. Verification

To evaluate the verification accuracy of the proposed method, each palmprint image is matched with all the other palmprint images in the database. When two verified samples are determined to belong to the same palm, there are two kinds of matchings, one is genuine matching if they are taken from the same palm and the other is impostor matching if they are taken from different palms. Because there are a total of 6000 different palmprint images, so genuine matchings will happen \( C_2^2 \times 500 = 33,000 \) times, and impostor matchings will happen \( C_2^2 \times 500 = 17,964,000 \) times. Table 2 and Fig. 7 shows respectively the table lists and the graphs of the error rates obtained when performing palmprint verification on the red, green and blue illumination of the PolyU 2.0 database. Fig. 8 depicts the corresponding ROC curves of the latter experiment, which is a plot of the false accept rate against the false rejection rate. The intersection of the curves with the EER line is the equal error rate.

<table>
<thead>
<tr>
<th>Verification (6000 images)</th>
<th>EER %</th>
</tr>
</thead>
<tbody>
<tr>
<td>illumination methods</td>
<td>Red</td>
</tr>
<tr>
<td>[1]</td>
<td>0.024</td>
</tr>
<tr>
<td>( P(\text{equal weight}) )</td>
<td>0.042</td>
</tr>
<tr>
<td>( P(\text{different weights}) )</td>
<td>0.036</td>
</tr>
</tbody>
</table>

Table 2. Verification performances of different methods on the PolyU 2.0 palmprint database.

E. Discussion

Table 1 and Table 2 show that both \( P(\text{different weights}) \) and \( P(\text{equal weight}) \) can achieve very good performance for palmprint recognition and palmprint verification, and \( P(\text{different weights}) \) performs even better than \( P(\text{equal weight}) \). Especially \( P(\text{different weights}) \) has an almost perfect recognition accuracy (99.97%) on the blue illumination which is the highest recognition accuracy in the literature. As for the verification suggested by Table 2, except for the red illumination, our method with different weights also achieve the best performance on all the other illuminations among the three compared methods. This is because that the features that we make used of in our method, such as principal palm lines and wrinkles, are not very distinguishable in the red illumination, however they appear clear in the blue illumination as shown in Fig. 6. Consequently, our method achieves better result with the blue illumination than the red one.

Fig. 6: Palmprint images of a palm taken respectively by red, green and blue illuminations

IV. CONCLUSION

The novelty of this work lies in the fact that ridges are extracted from the palmline to be used as a weighting factor for the matching step.

In summary, this paper proposes a robust palmprint matching algorithm which consists of three major processing modules: preprocessing, feature extraction, and matching. In the preprocessing module, an input palmprint image is preprocessed so that noise can be decreased and both image contrast and image size can be normalized. In the feature extraction module, an Omni-bearing palmline response image and a palmcode image are generated for each input palmprint image by using CMFRAT masks, and principal palmprint ridge lines are constructed from Omni-bearing palmline response images. In the matching module, each palmcode image is divided into a constant number of blocks, a histogram of direction bins is constructed for each block by accumulating the palmcode of this block with two kinds of pixel weights. Basically, if a pixel belongs to a principal ridge line, it has a larger weight. Finally, a block-based Chi-Square distance is used to match two palmprint images. Experiments on the public PolyU 2.0 Multispectral Palmprint Database, the proposed method has shown an almost perfect recognition rate (99.97%) and an extremely low EER (0.017%) by using the blue-channel palmprint images. From experiments, the proposed method has
shown its superiority in matching palmprint images.

V. ACKNOWLEDGEMENT

This work is supported by Ministry of Science and Science Technology, ROC with grant no. MOST 105-2221-E-216-018.

REFERENCES


Abstract—Automatic analysis of facial attractiveness has become an emerging research topic in recent years. In this paper, the importance of angles of lines connecting facial landmarks on three prominent facial organs, namely eyes, nose and chin, for facial beauty assessment is investigated. Furthermore, two geometric facial measurements, including ratios and angles, as well as stacked regression technique are employed in order to predict female face attractiveness. The SCUT-FBP benchmark database is used as our experimental dataset. The low mean absolute errors and high Pearson correlations indicate the high accuracy of our attractiveness predictor.

Keywords— Face attractiveness, Geometric features, SCUT-FBP, Facial beauty, Beauty assessment

I. INTRODUCTION

The importance of face attractiveness has become more evident in recent years, and people pay greater attention to facial beauty. This would definitely lead to increasing demand for plastic aesthetic surgeries and spending a considerable amount of money on cosmetic products [1-5]. As a result, a large number of researchers from several fields, namely computer science, human science and medicine, are drawn to facial beauty topic. Various experiments have empirically shown the strong influence of beauty on general social life [1-4]. It is noteworthy that individuals with attractive appearance would definitely receive higher-level of social acceptance, take better jobs, receive higher incomes, and have more fulfilling lives compared with people with unattractive or ugly faces [2]. Moreover, psychological research shows that people with attractive faces are highly likely to have more positive personalities [2]. Conversely, looking unpleasant may have a deep influence on self-esteem, which would definitely lead to depression and other serious psychological disorders [1].

It is worth mentioning that the supporters of the objective and measurable nature of beauty have proposed ideal proportions or beauty canons for the human shape since ancient times [1-3]. These facial beauty rules are golden ratios, neoclassical canons, vertical thirds and horizontal fifths rules. Interestingly, faces that follow these rules would definitely be considered as attractive. Considering the development of modern technologies, there is a great interest in discovering the quantitative relationship between attractiveness and facial features using computer vision, pattern recognition, and machine learning techniques. There are three major steps for computer-based facial beauty analysis [1]. In the first step, a set of face images are collected and human raters are asked to rate these images with some integer numbers. For example, for a 10-point scale, 1 is most unattractive and 10 is extremely attractive. In the following step, data which are pertinent to attractiveness are extracted from face images. Afterwards, an automatic rater is constructed using these data. Facial feature extraction methods can be divided into two distinct categories, namely holistic and feature-based techniques. Holistic techniques extract features from the whole face. These techniques, which perform an automatic extraction of the data from the whole face, can represent the face as a point in a high-dimensional face space [1, 2]. As a result of this, PCA can be employed to provide a low-dimensional face representation such as Eigen faces. A wide range of features, including geometric, skin smoothness, color and other local structural features, are used in feature-based approach in order to represent a face. Geometric features, which are derived from positions of the facial landmarks, distances between them, and ratios of the distances between important landmarks, are the most significantly used features in facial beauty research studies. Therefore, finding the important features of the face such as nose, eyes, mouth and the related landmarks can be considered as a preliminary step in most facial beauty research studies [1].

Eisental et al. [5] assessed facial beauty using 37 geometric features (37 distances and ratios from 36 landmarks).

features such as make-up as well as image properties. Overall, they considered 37 different characteristics. Sutic et al. [12] employed 25 geometric ratios. Mu [13] attempted to test normalized geometrical coordinates of 36 landmarks to predict face attractiveness.

Despite the fact that angles and inclinations are useful geometric facial measurements, a few researchers used them in their experiments. Liu et al [4] used 574 geometric features (318 ratios, 232 angles, and 24 inclinations from 82 frontal landmarks) as well as 92 features (24 ratios, 42 angles, and 26 inclinations from 40 profile landmarks). These 574 and 92 geometric features reduced to 124 and 51 principal components by PCA, respectively. In [14], Kagian et al. extracted 84 facial landmarks from face images, and computed associated distances and slopes of lines (also known as inclination) connecting facial landmarks (3486 distances and 3486 slopes). Afterwards, these 6972 geometric features were reduced to 90 by PCA. Schmid et al. [15] employed 78 ratios and symmetry indicators from 29 landmarks (6 canons, 55 symmetry and 17 golden ratios). They used both angles and distances to compute the symmetry, but they reported a slight increase in their results. Whitehill and Movellan [16] employed geometric features based on angles. However, they did not explain which angles were employed, and also they did not report explicit results of angles.

It is worth mentioning that attractiveness estimation can be deemed to be a classification or a regression problem. Various regression techniques have been used in face attractiveness research studies, including linear [5, 8, 11, 14, 15], Lasso [13], ridge [13], Gaussian [8], pace [8], support vector regression (SVR) [4, 8, 11] and nonlinear regression [9]. Additionally, a large number of classification methods have been employed by researchers, including SVM [5, 7], KNN [5, 12], ANN [12], Ada Boost [12], C4.5 [6, 7] and Decision Tree [17].

The aim of this study is to investigate only the effect of angles of lines connecting facial landmarks on prominent facial organs, namely eyes, nose, mouth and chin. In this paper, geometric ratios between 68 landmark distances as well as angles of lines connecting facial landmarks on important facial regions are used as the representation of a face. Furthermore, stacked regression, which is a combination of different regression models, is employed to train the computational model.

The remainder of this paper is organized as follows. Section II presents the experimental data, feature extraction method as well as regression technique which are used in this study. In Section III, the experimental results are presented, analyzed and discussed. Finally, Section IV concludes the paper by summarizing the main findings and future work.

II. AUTOMATIC ANALYSIS OF FACIAL ATTRACTIVENESS

A. Preprocessing and face representation

In this study, the SCUT-FBP benchmark database [8] is used as our experimental dataset, which contains 500 images of Asian female faces. These images were rated with integer numbers ranging from 1 to 5. These facial images are of very different sizes. Therefore, it is of key importance to normalize the faces. For this reason, the faces are warped to a 600×600 image. Fig. 1 shows a face image before and after normalization. In the following step, the prominent facial regions and the related landmarks are identified. For each facial image, 68 facial landmarks are automatically extracted using “dlib” library [18]. An example of a facial image and the locations of 68 landmarks is shown in Fig. 2 (left). Furthermore, geometric features (ratios and angles) are used in this paper to represent a face. Angles of lines connecting facial landmarks on three prominent facial organs, namely eyes, nose and chin, are shown in Fig. 2 (right). Chen and Zhang [19], [20] summarized 26 putative ratio rules which have been proposed in existing literature. Only 22 of them are tested using our 68 landmarks (see Table 1). Additionally, Fig. 3 illustrates the distances between landmarks, which are used to define these putative facial ratios.
B. Construction of an attractiveness Predictor

In the following step, the selected geometric features are mapped to an attractiveness score using stacked regression technique [21] in order to evaluate the effectiveness of our computational model. Stacked regression technique combines multiple regression models via a meta-regressor. Said differently, the first-level models are trained using the original training set, and then the second-level model (meta-model) is trained on the outputs of the first-level models, which are considered as input features [22].

It should be noted that in this study stacked regression technique combines three regression models (namely, linear, linear SVR and ridge) wherein SVR with a radial basis function (RBF) kernel is used as a meta-regressor (see Fig. 4).

![Fig. 4. The stacking framework.](image)

In the above figure, “PL”, “PS” and “PR” indicate predictions of linear, support vector and ridge regression methods, respectively.

III. EXPERIMENTAL RESULTS

As mentioned in Section II, the SCUT-FBP benchmark dataset recently introduced by Xie et al. [8] is employed in this study. Moreover, two prediction metrics, namely Pearson Correlation (PC) and Mean Absolute Error (MAE), are used to assess the performance of our automatic rater.

Furthermore, several validation methods, including 10-fold cross validation, 5-fold cross validation, 6-fold cross validation, leave-one-out cross validation (LOOCV), randomly selected 70% data for training, are utilized in this study.

In the first step, the experiments are performed using 10-fold cross validation. Table II indicates the effect of angles of lines connecting facial landmarks on prominent facial organs, namely eyes, nose, mouth and chin using stacked regression technique. It can be observed from Table II that Pearson correlation values increase, and mean absolute errors decrease when adding angle features. It is noticeable that a Pearson correlation of 0.504 is obtained when using only Geometric ratios (22 features). However, the Pearson correlation rises to 0.653 by adding angle features. When the angle of lines connecting facial landmarks on mouth is used as a feature, there is a slight decrease in the Pearson correlation values. Consequently, it is not included for the rest of the experiment.

![TABLE II. EVALUATION OF FEATURES FOR FACIAL BEAUTY PREDICTION](image)

<table>
<thead>
<tr>
<th>Facial features</th>
<th>PC</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>22 ratios</td>
<td>0.504</td>
<td>0.383</td>
</tr>
<tr>
<td>22 ratios + 1 angle (chin)</td>
<td>0.627</td>
<td>0.336</td>
</tr>
<tr>
<td>22 ratios + 2 angles (chin and nose)</td>
<td>0.635</td>
<td>0.328</td>
</tr>
<tr>
<td>22 ratios + 3 angles (chin, nose and left eye)</td>
<td><strong>0.653</strong></td>
<td><strong>0.320</strong></td>
</tr>
</tbody>
</table>

The performance of four regression techniques, namely linear, support vector (SVR), ridge and stacked regression, are compared in Table III.

![TABLE III. PREDICTION PERFORMANCE USING GEOMETRIC FEATURES](image)

<table>
<thead>
<tr>
<th>Facial features</th>
<th>Technique</th>
<th>PC</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>22 ratios</td>
<td>SVR</td>
<td>0.495</td>
<td>0.406</td>
</tr>
<tr>
<td>22 ratios</td>
<td>Linear regression</td>
<td>0.497</td>
<td>0.403</td>
</tr>
<tr>
<td>22 ratios</td>
<td>Ridge regression</td>
<td>0.495</td>
<td>0.409</td>
</tr>
<tr>
<td>22 ratios</td>
<td>Stacked regression</td>
<td>0.504</td>
<td>0.383</td>
</tr>
<tr>
<td>22 ratios + 3 angles</td>
<td>SVR</td>
<td>0.625</td>
<td>0.347</td>
</tr>
<tr>
<td>22 ratios + 3 angles</td>
<td>Linear regression</td>
<td>0.618</td>
<td>0.347</td>
</tr>
<tr>
<td>22 ratios + 3 angles</td>
<td>Ridge regression</td>
<td>0.623</td>
<td>0.352</td>
</tr>
<tr>
<td>22 ratios + 3 angles</td>
<td>Stacked regression</td>
<td><strong>0.653</strong></td>
<td><strong>0.320</strong></td>
</tr>
</tbody>
</table>

It can be seen that the best Pearson correlation (0.653) and the lowest mean absolute error (0.320) are achieved by using 25 geometric features (22 ratios + 3 angles) and stacked regression method. Interestingly, stacked regression technique outperforms other regression techniques. Fig. 5 illustrates the results of different features and regression techniques, which are evaluated by 10-fold cross validation.
Moreover, in order to directly compare the proposed algorithm with existing algorithms, some results reported by Xie et al. [8] who used the SCUT-FBP benchmark dataset (the same dataset) are shown in Table IV. Moreover, the algorithm proposed by Mao et al. [7] is re-implemented. In this paper, the suggested 17 geometric features (17 distances) are used and then SVR and linear regression techniques are utilized. Similar figures are obtained by SVR and linear regression (Pearson correlation of 0.572 and 0.574, respectively). Similarly, Xie et al. [8] used 18 distances. The best result obtained by SVR (Pearson correlation of 0.608). It is noticeable that a higher correlation (0.627) as well as a lower mean absolute error (0.339) are achieved when adding angle features and using stacked regression.

Besides Geometric features, Xie et al. [8] employed Gabor features and they reported correlation of 0.643 and mean absolute error of 0.396. Interestingly, a higher prediction accuracy is achieved by our proposed algorithm (correlation of 0.653 and mean absolute error of 0.320).

In the following step, different validation methods including 5-fold cross validation, 6-fold cross validation, leave-one-out cross validation (LOOCV), randomly selected 70% data for training, are used. One of facial beauty study challenges is that there is no extensive publicly available database. Consequently, researchers usually use various databases which have different sizes. This study attempts to use the same validation method as existing works, although face databases are different.

Table V presents information on various aspects of face databases used in existing works, namely number, gender, pose and expression.

A summary of the beauty prediction methods as well as their accuracy in existing works are shown in Table VI. Eisenthal et al. [5] and Kagian et al. [14] who used LOOCV method reported Pearson correlation of 0.65 and 0.74, respectively. It should be noted that a correlation of 0.74 was obtained by using only geometric features and excluding all non-geometric features in [14]. Pearson correlation of 0.765 is achieved by our proposed algorithm using LOOCV.

Mu [13] chose 70% of data for training and achieved MAE equal to 0.52. When 70% of images (350 face images) are randomly selected for training, our algorithm obtains a MAE equal to 0.334.

Moreover, the best result reported by Dantcheva et al. [11] who used 6-fold cross validation method is Pearson correlation of 0.65. They also re-implemented the proposed algorithm by Mao et al. [7] and reported a correlation of 0.562. Pearson correlation of 0.713 is achieved by our algorithm evaluated by 6-fold cross validation. This correlation decreases to 0.693 when our algorithm is assessed by 5-fold cross validation. Whithehill and Movellan [16] reported a Pearson correlation of 0.45 when using 5-fold cross validation.

### Table IV. Comparison of Beauty Estimation Algorithms

<table>
<thead>
<tr>
<th>Facial features</th>
<th>Technique</th>
<th>PC</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>17 distances [7]</td>
<td>SVR</td>
<td>0.572</td>
<td>0.372</td>
</tr>
<tr>
<td></td>
<td>Linear regression</td>
<td>0.574</td>
<td>0.382</td>
</tr>
<tr>
<td>18 distances [8]</td>
<td>SVR</td>
<td>0.608</td>
<td>0.402</td>
</tr>
<tr>
<td></td>
<td>Linear regression</td>
<td>0.592</td>
<td>0.412</td>
</tr>
<tr>
<td></td>
<td>Pace regression</td>
<td>0.584</td>
<td>0.413</td>
</tr>
<tr>
<td></td>
<td>Gaussian regression</td>
<td>0.605</td>
<td>0.401</td>
</tr>
<tr>
<td>17 distances + 3 angles</td>
<td>Stacked regression</td>
<td>0.627</td>
<td>0.339</td>
</tr>
<tr>
<td>Holistic (Gabor + PCA) [8]</td>
<td>SVR</td>
<td>0.584</td>
<td>0.423</td>
</tr>
<tr>
<td>Hybrid(Geometric + Gabor) [8] + SVR</td>
<td>0.643</td>
<td>0.396</td>
<td></td>
</tr>
<tr>
<td>Our features (22 ratios + 3 angles)</td>
<td>Stacked regression</td>
<td>0.653</td>
<td>0.320</td>
</tr>
</tbody>
</table>

### Table V. Review of the Face Databases Used in Existing Works

<table>
<thead>
<tr>
<th>Reference (publication year)</th>
<th>Size</th>
<th>Gender</th>
<th>Pose</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Set2:92</td>
<td>female</td>
<td>frontal</td>
<td>almost neutral</td>
</tr>
<tr>
<td>Kagian et al. [14] (2008)</td>
<td>91</td>
<td>female</td>
<td>frontal</td>
<td>neutral</td>
</tr>
<tr>
<td>Whithehill and Movellan [16] (2008)</td>
<td>2000</td>
<td>both (F/M)</td>
<td>frontal</td>
<td>different</td>
</tr>
<tr>
<td>Mu [13] (2013)</td>
<td>250</td>
<td>Both (F/M)</td>
<td>frontal</td>
<td>almost neutral</td>
</tr>
<tr>
<td>Xie et al. [8] (2015)</td>
<td>500</td>
<td>female</td>
<td>frontal</td>
<td>neutral</td>
</tr>
</tbody>
</table>
TABLE VI. A SUMMARY OF PREDICTION METHODS.

<table>
<thead>
<tr>
<th>Reference (publication year)</th>
<th>Features</th>
<th>Reg. Method</th>
<th>validation</th>
<th>PC</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eisenthal et al. [5] (2006)</td>
<td>hybrid</td>
<td>Linear</td>
<td>LOOCV</td>
<td>0.65</td>
<td>N/A</td>
</tr>
<tr>
<td>Kagian et al. [14] (2008)</td>
<td>Geometric</td>
<td>Linear</td>
<td>LOOCV</td>
<td>0.74</td>
<td>N/A</td>
</tr>
<tr>
<td>Our method</td>
<td>Geometric (22 ratios + 3 angles)</td>
<td>Stacked</td>
<td>LOOCV</td>
<td>0.765</td>
<td>0.315</td>
</tr>
<tr>
<td>Whithehill and Movellan (2008)[16]</td>
<td>Holistic (texture)</td>
<td>SVR</td>
<td>5-fold</td>
<td>0.45</td>
<td>N/A</td>
</tr>
<tr>
<td>Our method</td>
<td>Geometric (22 ratios + 3 angles)</td>
<td>stacked</td>
<td>5-fold</td>
<td>0.693</td>
<td>0.334</td>
</tr>
<tr>
<td>Mu [13] (2013)</td>
<td>hybrid</td>
<td>Ridge</td>
<td>70% data for training</td>
<td>N/A</td>
<td>0.52</td>
</tr>
<tr>
<td>Our method</td>
<td>Geometric (22 ratios + 3 angles)</td>
<td>Stacked</td>
<td>70% data for training</td>
<td>0.694</td>
<td>0.334</td>
</tr>
<tr>
<td>Dantcheva et al. [15] (2015)</td>
<td>37 features (hybrid)</td>
<td>Linear</td>
<td>6-fold</td>
<td>0.65</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>Geometric (17 distances)</td>
<td>SVR</td>
<td>6-fold</td>
<td>0.562</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>Geometric (22 ratios + 3 angles)</td>
<td>stacked</td>
<td>6-fold</td>
<td>0.713</td>
<td>0.325</td>
</tr>
</tbody>
</table>

IV. CONCLUSION

To sum up, this paper develops an attractiveness computation model based on geometric ratios and angles. It can be concluded that angles of lines connecting facial landmarks on three prominent facial organs, namely eyes, nose and chin, are effective features for facial attractiveness modeling. Furthermore, stacked regression which combines three regression models (linear, linear SVR and ridge) is used to train the computation model. Several validation methods are used in this study in order to evaluate our prediction model. The small mean absolute error values and high Pearson correlations demonstrate the effectiveness of our face attractiveness predictor. It is noteworthy that makeup and hair style would certainly have a significant impact on the attractiveness of a face especially female faces. Therefore, extending computer attractiveness analysis to non-permanent facial features such as makeup, presence of glasses, and hairstyle creates a novel promising area of research. Furthermore, it is worthwhile to investigate male images, and also identify distinguishing beauty features for male and female faces.

REFERENCES

Restoring height-map images of shiny coins using spline approximation to detect counterfeit coins

Saeed Khazaee  
Department Of Computer Science and Software Engineering CENPARMI, Concordia University  
Montreal, Canada  
s_khaza@encs.concordia.ca

Maryam Sharifi Rad  
Department Of Computer Science and Software Engineering CENPARMI, Concordia University  
Montreal, Canada  
sey_shar@encs.concordia.ca

Ching Y. Suen  
Department Of Computer Science and Software Engineering CENPARMI, Concordia University  
Montreal, Canada  
suen@encs.concordia.ca

Abstract—There are presently a lot of companies, museums, and government agencies in the world which have increased the demand of automatic systems to classify precious, historical, and common coins. Thanks to the increased exigency on the intelligent approaches to recognize counterfeit coins, coin detection has been constantly evolving in recent years and image-based coin recognition has become an integral part of this research area.

In this research, we have been studying various methods related to image-based coin detection in 2-D image processing and we also used a 3-D scanner to scan hundreds of coins to apply 3-D approaches to detect counterfeit coins. Here, instead of conventional 2-D methods for counterfeit coin detection, we applied a 3-D approach to model and analyze a large set of different Danish coins. One of the most important advantages of 3-D approaches is the ability to extract features which cannot be found in 2-D images. Despite this advantage, we had a lot of unexpected degradations on shiny coin images. In order to restore the degraded images, we proposed a method based on signal separation in this paper. The experimental results showed that the proposed method outperformed previous methods in detection of counterfeit coins.

Keywords—counterfeit coin detection, spline approximation, 3D images

I. INTRODUCTION

An image-based coin recognition system takes digital images of coins as input and classifies them according to their properties. Pursuant to the previous works, coin detection is mostly based on 2-Dimensional image processing techniques. In particular, edge information had been extracted as features with Fourier transform. The difficulty level of coin detection heavily depends on the backgrounds of the coin datasets used in the experiments. Fortunately, all the datasets used for coin detection are captured with plain backgrounds, such as pure white or conveyor belt with extremely low gray level [1]. Some of the common features in coin recognition like coin diameter, thickness, weight or shape may be used to detect counterfeit coins. We should note that these basic attributes are easy to copy and the fake coins made nowadays are of high quality. Therefore, these systems cannot distinguish fake coins from genuine ones when their physical attributes are basically the same. To increase the potential of fake coin detectors and image-based coin detection, several methods based on image processing techniques and classification algorithms were proposed [2,3,4,5] and many lectures and tutorials were devoted to them [6,7]. Some of these methods are not very sophisticated and use coins’ colors and radius based features to detect counterfeit coins. Unfortunately, these approaches are incapable of distinguishing coins having the same values of the limited set of features.

In recent years, image-based counterfeit detection has expanded and many researchers have applied image processing techniques to extract effective features from the texture of the coin images [8,9]. In general, edge detection has been frequently employed in the feature extraction process. In references [9,10], an edge map was extracted and segmented into several parts. After that, the authors proposed a method using histogram analysis and Fourier transformation to handle and recognize rotated coin images. However, the extracted features were not very useful for noisy and degraded images such as rust, dust, or sulfated coin images. In [9], the authors extracted the letters on the coin surface and tried to recognize them. In spite of the mentioned novelties of the proposed method, it is clear that the method was not robust enough to distinguish the counterfeit from genuine coins whose images have weak or smooth edges. In reference [11] a new method using rotation-invariant region binary patterns based on gradient magnitudes was proposed. To increase the accuracy of coin recognition, it computes gradient magnitudes in a coin image and extracts rotation-and-flipping-robust features using local difference magnitude transform. Although the result is acceptable from feature extraction point of view, the time-performance of the model is not satisfactory. In references [12] and [13] authors proposed their methods to recognize coin images in terms of their denominations and countries of origin. However, the problem of coin recognition
significantly differs from fake coin detection investigated in this study. For coin recognition, one important concern is to reduce the sensitivity to the variations among the coins from the same class. Therefore, it is highly probable that the fake coins, especially those of high quality, are classified as belonging to the same class as their genuine counterparts under the coin recognition framework.

As we discuss later, we have been using a three-dimensional scanner to create an image data set for Danish coins. In this research, during the study on 3-D approaches to extract effective features, we have faced many challenges related to shiny coins and tried to solve this problem. There are many different technologies used in 3-D scanners; each technology has its own advantages, restrictions, and cost. For example, optical technologies face many difficulties when processing reflecting, transparent, or shiny objects. Fig. 1 illustrates how a 3-D scanner captures height and depth: it is robust regarding the quality of the coin surface, while the 2-D image of this coin is completely unreadable. In Fig. 2 we can see the degradation of a shiny coin image captured by the 3-D scanner which uses optical technology. As it can clearly be seen from this figure, in spite of the remarkable advantages of 3-D scanning for poor quality coins, there may be some abnormal and invalid results while scanning shiny and high-quality coins. [14] However, the validity of the data captured by scanning is crucial for this research as any invalid value related to height and depth will adversely affect the pattern recognition process. Therefore, our proposed restoration module should enhance not only the quality of the images as a whole, but also each small part of the image must be restored as precisely as possible. [14]

The rest of this paper is organized as follows: Section II describes the proposed method to detection counterfeit coins. In this section we will explain our proposed image restoration and feature extraction. In Section III, we talk about the experimental results and analyze them, we explain our new method to edge plane detection. Finally, we conclude in Section IV and comment on some future works.

![Image 1](image1.jpg)  
(a) Twenty Kroner 1990 coin captured by a normal camera, and (b) the same coin captured by a 3-D scanner.

![Image 2](image2.jpg)  
Fig. 2. Degradation problem for shiny coin’s image in a height-map image (wrong height and depth information)

## II. COUNTERFEIT COIN DETECTION

### A. Restoring Images Using Spline Approximation

After the straightening process, which we proposed in [14], we can see an adverse effect of improper scanning on shiny coin images. For some kinds of coins, for example Danish 20 kroner 2008, all the images are totally degraded. Our initial experiments showed that the degradation is not a random noise. We scanned three coins, four times for each coin, and we obtained the same poor results [14].

Since our proposed counterfeit coin detector is based on height and depth information, signal restoration from this digital data is an inevitable process. For a given digital data set obtained from scanning, we needed to recover the original signal, which includes more precise data, since the degraded signals related to fake and genuine coins are very similar. Here, the important information of letters and numbers corresponds to a low-frequency signal. Therefore, we must estimate the low-frequency signal and subtract it from the original signal. Thus, we have $x = A + D$; where, $x$ is the original and degraded signal, $A$ is an approximation of $x$, and $D$ contains the details of $x$. In other words, the useful information comes from the details of the signal. In our first attempt, we used wavelet transform to find $A$ or $D$. For restoring all signals and the image, several well-known methods were performed.

1) **Estimating Degradation Function using splines**  
For estimating more accurately the degradation function for each 1-D signal, we proposed an estimator, which uses control points determined by the Catmull-Rom spline. Fig. 7 shows the basis, control matrix used for estimating a Catmull-Rom spline in a specific period. Since the signals, on which the process should be performed, are not smooth enough to perform mathematic operations, we used a low-pass filter to smooth them. The critical selected point or control point is routinely an extremum point selected by the effective point selector function. This function uses a momentum to avoid local minima, which should be ignored and the signal has a lot of them. In addition, if the distance between two control points is greater than a specific threshold, the point selector will consider several points on this part of the signal instead of the critical points.
The principal advantage of this technique is that the points along the original set of points also make up the control points for the spline curve. [15] Two additional points are required on either end of the signal. Fig. 8 illustrates a small part of the original signal and the approximation of degradation signal.

Consequently, we used a low-pass filter again to obtain the final signal and we have \( A = \text{inverse}(F \ast H) \) where \( F \) and \( H \) are Fourier transforms of \( f \), and a 1-D Gaussian filter in the order. Also, the \( \text{inverse} \) function returns the inverse discrete Fourier transform of the resulted vector. In addition, there is a simple parameter optimizer which regulates the value of \( \sigma \) in Gaussian filter, with the aim of minimizing the Euclidean distance between \( A \) and \( f \). Fig. 9 demonstrates that the original image has been restored successfully. The signal related to the marked row of the restored image shows that the heights of the letters are more precise than the original rectangular image. Fig. 9 also shows that the proposed restoration produces more balanced height information for the surface of the coin. In the top of the figure, we can see a 3-D view of the straightened image of a shiny coin, and in the bottom, the restored image by the proposed method.

B. Feature extraction

Here, we consider one 1-D signal as a representative of the straightened and restored image. To achieve this aim, we calculated the median of each column of the matrix instead of all rows and made a new vector or signal which we call it, the median signal. The median signal of different coins’ images is shown in Fig. 10. Figures (a) through (c) show that the median signal of original coins which are ideally similar while in Fig. 10 (d) it is clear that the median signal of the fake coin is totally different from the genuine coins’ pattern. Therefore,
the median signal is a desirable representative for the image and we have one 1-D signal to extract features.

Extracted features in the feature extraction module were quite dependent on the time (horizontal parameter of the signal) and starting point. Here, for extracting a lot of features, we must consider a base signal and shift other signals to the same position.

1) Shifting the Median Signal
To shift the median signal to the correct position, we considered a signal as a base signal and shifted the signal to the right direction until a similarity criterion was satisfied. To compare similarity between the signals, the cross correlation has been used. From the cross-correlation function, we can obtain the correlation coefficient which will give us a single value of similarity. Right from the start, the similarity of the median signal and base signal was calculated then the median signal was shifted to the right. The best position for the median signal was marked when the similarity was the highest value. Fig. 11 (a) shows the median signal of one coin which we consider as a base of shifting. Fig. 11 (b) is the shifted signal of (c). The figure shows that the signal was shifted to the best position.

After shifting the median signal, we extracted two different kinds of features. The first kind of features were the statistical features and the next were the information about the band power, bandwidth, and distortion. Experimentally, to extract these features, we divided the signal into four parts and extracted the features for each part separately.

2) Statistical features
The first statistical feature which is extracted from each part of the signal, is the period of sequence which is computed as the minimum length of a subsequence $x_i(1:p)$ of signal $x$ that repeats itself continuously every $p$ samples in $x_i$. The length of $x$ does not have to be a multiple of $p$, so that an incomplete repetition is permitted at the end of $x_i$. If the sequence $x$ is not periodic, then $p = \text{length}(x_i)$.

To extract four next features, we estimated the mean normalized frequency of the power spectrum of a time-domain signal, $x_i$.

Features 9 through 12 are the ratio of the largest absolute value in $x$ to the root-mean-square (RMS) value of $x_i$ which are also called Peak-magnitude-to-RMS ratio. In all explanation, $x_i$ is one of the four parts of the original signal and $i = 1, 2, 3, 4$.

3) Peak Analysis features
To extract features based on peak analysis, we found the prominent peaks as far as the number of the peaks in all signals were equal. Then we estimated the average distances between peaks for each $x_i$. Therefore, there are four features based on peak analysis. Finally, we have 16 features through the feature extraction process.

III. EXPERIMENTAL RESULTS
In this paper, various well-known classifiers were employed to evaluate the performance of the proposed method. We used CENPARMI Danish dataset introduced in [14]. Table I illustrates the results of three metrics comparing the performance of the classification process for the system using 16 extracted features. In terms of Detection Rate or TP (True Positive) and FP (False Positive) which are weighted and averaged for both of genuine and fake classes; Stochastic Gradient Descent (SGD), and MLP had better results than others. SGD has correctly classified 99.3% of the coins.
tested. For the other classifiers, we can see the acceptable results in the metrics. Turning to precision, recall, and f-value, SGD was still the most remarkable classifier in Table I. In spite of the comparatively weaker results for NaïveBayes tree, and decision in detection rate, other classifiers could classify both fake, and genuine coins satisfactorily. As it can be seen clearly from Table II, precision, recall, and f-value were indispensable in evaluating the accuracy of the classifications and the results demonstrated that although NaïveBayes and decision tree had an acceptable outcome in the detection of the genuine coins and were very fast in building model, they were relatively poorer in detecting the fake coins. Apart from the capability of classifiers in the mentioned standard criteria, all the results are satisfactory for the proposed method and illustrate that the extracted features are very effective.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Time taken to build model (sec)</th>
<th>TP (%)</th>
<th>FP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGD</td>
<td>&lt; 0.01</td>
<td>99.3</td>
<td>3.8</td>
</tr>
<tr>
<td>MLP</td>
<td>0.6</td>
<td>98.4</td>
<td>3.6</td>
</tr>
<tr>
<td>NaïveBayes</td>
<td>&lt; 0.01</td>
<td>92.7</td>
<td>10.4</td>
</tr>
<tr>
<td>Decision Tree (J48)</td>
<td>&lt; 0.01</td>
<td>92.8</td>
<td>26.1</td>
</tr>
<tr>
<td>Lazy.Kstar</td>
<td>&lt; 0.01</td>
<td>96.8</td>
<td>15.5</td>
</tr>
<tr>
<td>Logistic</td>
<td>&lt; 0.01</td>
<td>96.6</td>
<td>4.5</td>
</tr>
</tbody>
</table>

TABLE II. COMPARING THE RESULTS OF CLASSIFIERS IN TERMS OF PRECISION, RECALL, AND F-VALUE FOR CLASSIFYING FAKE AND GENUINE COINS BY 16 FEATURES

<table>
<thead>
<tr>
<th>Class</th>
<th>Measure</th>
<th>SGD</th>
<th>MLP</th>
<th>NaïveBayes (J48)</th>
<th>DT (J48)</th>
<th>Lazy.Kstar</th>
<th>Logistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Genuine</td>
<td>Precision</td>
<td>0.991</td>
<td>0.993</td>
<td>0.981</td>
<td>0.933</td>
<td>0.966</td>
<td>0.993</td>
</tr>
<tr>
<td></td>
<td>Recall</td>
<td>1</td>
<td>0.976</td>
<td>0.936</td>
<td>0.965</td>
<td>0.983</td>
<td>0.963</td>
</tr>
<tr>
<td></td>
<td>F-value</td>
<td>0.996</td>
<td>0.987</td>
<td>0.956</td>
<td>0.949</td>
<td>0.974</td>
<td>0.978</td>
</tr>
<tr>
<td>Fake</td>
<td>Precision</td>
<td>1</td>
<td>0.90</td>
<td>0.712</td>
<td>0.778</td>
<td>0.90</td>
<td>0.812</td>
</tr>
<tr>
<td></td>
<td>Recall</td>
<td>0.955</td>
<td>0.964</td>
<td>0.864</td>
<td>0.636</td>
<td>0.818</td>
<td>0.964</td>
</tr>
<tr>
<td></td>
<td>F-value</td>
<td>0.997</td>
<td>0.927</td>
<td>0.776</td>
<td>0.7</td>
<td>0.857</td>
<td>0.881</td>
</tr>
</tbody>
</table>

IV. CONCLUSION AND FUTURE WORK

We have been trying to find a comprehensive way to detect counterfeit coins which can analyze the coin images as precisely as possible. In our preliminary counterfeit coin detector, the comparison of the performance of several well-known classifiers in Weka [16] illustrated that the proposed method gives excellent results in detecting counterfeit coins. Employing multi-processing will facilitate the restoration of the signals and the reduction of the processing time. Surprisingly, the proposed method had significant outcome on counterfeit coin detection when we use the four types of coins for training and testing together. Finally, it was apparent that the proposed counterfeit detection method is robust against rust, dust, and sulfation, cases. However, the biggest challenge in detecting fake coins is always the lack of fake samples. In some cases, it is nearly impossible to have fake coins as many as we can build a model or test our system. Hence, investigating feasibility of using unary (one-class) classification and PU (Positive and Unlabeled samples) learning for improving the accuracy of the counterfeit coin detector is open for the future works.

REFERENCES

Automatic High-Speed Compressive Tracking with Motion Prediction

Hongjun Li, Wei Hu
a Nantong University, School of Electronic and Information
b Nantong Research Institute for Advanced Communication Technologies
c TONGKE School of Microelectronics
Nantong, China
lihongjun@ntu.edu.cn

Abstract—Compressive tracking algorithm is a simple and efficient tracking algorithm, which has good accuracy and robustness performance. Despite much success has been demonstrated, numerous issues remain to be addressed. First, the samples selected by the algorithm do not have discriminative representation and the number of samples is relatively huge, affecting the computational complexity. Furthermore, the search window of the algorithm is fixed, when the target scale changes, noise is easily introduced into the Bayes classification model and leads the loss of tracking. In this paper, an automatic compressive tracking based on motion prediction is proposed. The number of samples is reduced and the tracking speed is improved by using motion prediction, the tracking accuracy is also improved by adjusting the tracking window size adaptively. Extensive experiments in subjective and objective comparison are performed on the OTB datasets. Experiment results show the proposed algorithm significantly better than several state-of-art algorithms on the OTB datasets. The computationally efficiency of the proposed algorithm reduced 75% compared to compressive tracking and nearly 10% of DSST tracker. Our tracker accuracy and robustness have been improved and the speed of calculation is faster.

Keywords—Compressive tracking, Motion prediction, Adapt window size, High speed

I. INTRODUCTION

As the rapid development of the electronic technology, computer vision has become a hot research topic. Video object tracking is an issue which contains image processing, pattern recognition, signal processing and so on. Object tracking remains a challenging problem due to the appearance changes caused by pose, illumination, occlusion, motion and so on. An effective appearance model is of prime importance for the success of a tracking algorithm that has attracted much attention in recent years.

Numerous effective representation schemes have been proposed for robust object tracking in recent years. Isard and Blake [1] introduced the particle filter theory into the field of machine vision last century. Collins et al. [2] considered the object tracking as a binary classification problem and use the forward learning binary classifier to classify the target and the background. Parag et al. [3] proposed the adaptive boosting linear weak classifier. Hidayatullah et al. [4] used camshift to improve multi-hue and multi-object tracking performance. Candès et al. [5] proposed the compressive sensing, and Zhang et al. [6] introduced it into the field of object tracking. However, the explosive growth of video information aggravates the burden on the computer [7]. If the time of target tracking process can be reduced, it will help us a lot. So reducing the complexity of the algorithm to improve the efficiency needs further study.

Recently, Yang et al. [8] proposed a compressive tracking method based on SIFT [9]. In order to solve the problem of non-adaptive scale, Luo et al. [10] utilized the Mean shift [11] to predict the object motion, this can improve the robustness of tracking. Zhang et al. [12] proposed an adaptive method for learning parameter of the compressive tracking. Fiza et al. [13] predicted the position by the motion history image. The compressive sensing theory [14] shows that if the dimension of the feature space is sufficiently high, these features can be projected to a randomly chosen low-dimensional space which contains enough information to reconstruct the original high-dimensional features. The dimensionality reduction method via random projection [15] is data-independent, non-adaptive and information-preserving. Although these algorithms improve the robustness of the compressive tracking, there still exists several problems. First, the calculation of the tracking process is huge. Second, the size of the searching window cannot change adaptively, and it introduces the noise into the classifier easily. Furthermore, when the object scales change, it is difficult to detect the targets.

To solve these problems, this paper analyzed the compressive tracking algorithm in details, and design a novel tracking algorithm based on motion prediction. Utilized the motion prediction to reduce the number of the candidate samples collection and improved the operating speed of the algorithm. Used the adaptive size of search window algorithm to adapt the scales of the target and avoid the noise into the module of the classifier. The proposed compressive tracking algorithm runs at real-time and performs favorably against
state-of-the-art trackers on challenging sequences in terms of efficiency, accuracy and robustness.

We propose a high speed compressive tracking based on motion prediction with adaptive size of search window (ASSWCT), which learns motion direction for search and scale estimation. Several advantages of ASSWCT is summarized as follows: (1) predict motion direction to reduce the number of the candidate regions; (2) change the size of search range adaptively by the motion vector; (3) collect the positive and negative samples according to the successful rate; (4) change the size of the tracking window adaptively according to the size of target; (5) run at real-time and perform favorably against state-of-the-art trackers on challenging sequences in terms of efficiency, accuracy and robustness.

The remainder of the paper is organized as follows. Section 2 we introduce the compressive tracking. Section 3 presents a high speed compressive tracking algorithm based on motion prediction and the details is shown in Section 4. Section 5 is devoted to experimental results and analysis, and Section 6 concludes the paper.

II. COMPRESSIVE TRACKING

The compressive tracking algorithm proposed by Zhang et al. [6] is divided into three parts, namely, feature extraction, feature compression, classifier construction and update.

A. Compressed Sensing

A random matrix \( R \in \mathbb{R}^{n \times m} \) whose rows have unit length projects data from the high-dimensional feature space \( x \in \mathbb{R}^n \) to a lower-dimensional space \( v \in \mathbb{R}^m \)

\[
  v = Rx
\]

where \( n << m \). Each projection \( v \) is essentially equivalent to a compressive measurement in the compressive sensing encoding stage. If the random matrix \( R \) in (1) satisfies the Johnson-Lindenstrauss lemma, \( x \) can be reconstructed with minimum error from \( v \) with high probability if \( x \) is K-sparse. This strong theoretical support motivates us to analyze the high-dimensional signals via their low-dimensional random projections.

B. Feature Extraction

As the coefficients in the measurement matrix can be positive or negative, the compressive features compute the relative intensity difference in a way similar to the generalized Haar-like features. Haar-like features are similar to Haar wavelet, whose characteristic value is the differences between two matrix pixels. Usually, Haar-like feature can be calculated by integral image method. The large set of Haar-like features are compressively sensed with a very sparse measurement matrix. The compressive sensing theories ensure that the extracted features of our algorithm preserve almost all the information of the original image. Therefore, we can classify the projected features in the compressed domain efficiently without curse of dimensionality.

C. Feature Compression

A very sparse random measurement matrix which can compress features, is defined as:

\[
  R_k = \sqrt{K} \begin{pmatrix}
  1 & p \frac{1}{\pi} \\
  -1 & p \frac{1}{\pi}
\end{pmatrix}
\]

Achioniopas [16] proved that this type of matrix with \( s = 1 \) or \( s = 3 \) satisfies the Johnson-Lindenstrauss lemma. This matrix is easy to compute which requires only a uniform random generator. More importantly, when \( s = 3 \), it is sparse where two thirds of the computation can be avoided. Furthermore, only the nonzero entries of \( R \) need to be stored which makes the memory requirement also very light. Then, the features can be compressed by function (1), where \( x \in \mathbb{R}^n \) is the original features, and \( R \in \mathbb{R}^{n \times m} \) is the measurement matrix. Therefore, the projected features can be classified in the compressed domain efficiently without the curse of dimensionality.

D. Classifier Construction and Update

We assume all elements in \( v \) are independently distributed and modeled with a naive Bayes classifier,

\[
  H(v) = \log \left( \prod_{k \in \mathbb{Z}} \frac{p(v_k | y=1)p(y=1)}{p(v_k | y=0)p(y=0)} \right) = \sum_{k \in \mathbb{Z}} \log \left( \frac{p(v_k | y=1)}{p(v_k | y=0)} \right)
\]

where we assume uniform prior, \( p(y=1) = p(y=0) \), and \( y \in [0, 1] \) is a binary variable which represents the sample label.

Diaconis and Freedman [17] show that random projections of high dimensional random vectors are almost Gaussian. Thus, the conditional distributions \( p(v_k | y=1) \) and \( p(v_k | y=0) \) in the classifier \( H(v) \) are assumed to be Gaussian distributed with four parameters \( (\mu^1, \sigma^1, \mu^0, \sigma^0) \) where,

\[
  p(v_k | y=1) \sim N(\mu^1, \sigma^1), \quad p(v_k | y=0) \sim N(\mu^0, \sigma^0)
\]

where \( \mu^1(\mu^0) \) and \( \sigma^1(\sigma^0) \) are mean and standard deviation of the positive(negative) class. The scale parameters in (4) are incremental update by

\[
  \mu^1 \leftarrow \lambda \mu^1 + (1-\lambda)\mu^0, \quad \sigma^1 \leftarrow \sqrt{\lambda \sigma^1 + (1-\lambda)\sigma^0} + \lambda (1-\lambda)(\mu^1 - \mu^0)
\]

where \( \lambda > 0 \) is a learning parameter, \( \sigma^1 = \sqrt{\frac{1}{n} \sum_{k \in \mathbb{Z}} (v_k - \mu^1)^2} \), \( \mu^0 = \frac{1}{n} \sum_{k \in \mathbb{Z}} v_k \). Parameters \( \mu^1 \) and \( \sigma^0 \) are updated with similar rules. The above equations can be easily derived by maximum likelihood estimation.
III. AUTOMATIC HIGH-SPEED COMPRESSIVE TRACKING BASED ON MOTION PREDICTION

A. Motion Prediction

In [6], the search strategy consider the area as the candidate regions which are around the upper left corner of previous target of the 20 Euclidean distances. The center is the position of the left vertex of the target in the previous image, and the other white pixels are the left vertexes of the candidate regions. Then, features of every candidate region which provides 50 compressive Haar-like features are extract. At last, these features of different candidate regions are calculated by a naive Bayes classifier, and the optimal region are selected as the target region in current frame.

In the processing of determining the target regions, using such a broad search strategy is not a good idea and wastes computing time. So the motion prediction is introduced into the search strategy to predict the motion direction of the target. Then search the target meticulously in the predicted direction and roughly in other directions. There are many algorithms to predict the motion trends, such as motion prediction combined with Mean Shift, and Motion History Image. When the tracking algorithm combines with the motion prediction, the robustness will be improved. However, the complexity of calculation is reduced limited, therefore, we will take an effect strategy to predict the motion, and try to reduce the computer complexity obviously.

Assume that the algorithm can track the target stably and the tracking result of the previous frames is correct. We can get the motion trend from two previous frames. The positions of the previous frames are \((x_1, y_1)\) and \((x_2, y_2)\) respectively, the motion vector \(\beta\) is defined as:

\[
\beta = (x_2, y_2) - (x_1, y_1)
\]

the motion vector \(\beta\) represents the motion direction in the previous frames. If the direction of the motion doesn’t change suddenly, the direction of motion in current frame is the same as the vector \(\beta\) . We take the search strategy that will search candidate regions meticulously in the predict direction and roughly in other directions. In the meticulous strategy, a matrix \(A\) which is the same dimension as the compressive tracking and divided into four quadrants as same as the Cartesian Coordinate System, which is shown in function (7).

\[
A = \begin{bmatrix}
1 & 0 & \cdots & 0 & 1 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
1 & 0 & \cdots & 0 & 1 \\
\end{bmatrix}
\]

(7)

The quadrant which is in the same direction as vector \(\beta\) will be searched meticulously, and it makes sure that the candidate regions are concentrated in the motion direction. If the target turns around suddenly, the regions will not be in the predict quadrant and the target will be lost. So, another matrix \(B\) is defined as:

\[
B = \begin{bmatrix}
1 & 0 & \cdots & 0 & 1 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
1 & 0 & \cdots & 0 & 1 \\
\end{bmatrix}
\]

(8)

In function (8), \(B\) is a matrix which diagonal value is 1. Take OR operation between matrix \(A\) and \(B\). The new candidate region can conclude almost all the probable regions.

The operating time of the compressive tracking is represented as \(O\):

\[
O = N_1 N_2 N_3
\]

(9)

where the number of the samples and the features are \(N_1\) and \(N_2\) respectively, \(N_3\) is the operating time of each feature.

The operating time of the compressive tracking with motion prediction is represented as \(O_1\):

\[
O_1 = \frac{23N_1 N_2 N_3}{80}
\]

(10)

as demonstrated in equation (9) and (10), the operating time \(O_1\) drop by nearly three-quarters, so the speed of tracking will be improved.

B. Search Window Optimization

The search size of candidate regions is uniform for different video in compressive tracking. If the target doesn’t move, it is not necessary to search in such a large range of candidate regions and will be a waste of resources. If the size of the search window can change adaptively according to the intensity of motion, the efficiency will be improved. In this paper, we calculate the motion distances between adjacent two frames, and change the size of search range in line with the distance. If the target is stationary, the strategy will use the uniform size window to search. In contrast, the size of window will change according to the motion distance.

C. Samples Selecting Optimization

The methods of the positive and negative samples collection are uniform in CT. The positive samples are the points offset four pixels and the negative samples are the points offset between 4 and 30 pixels,. In David video [18], the target is 75 pixels wide and 95 pixels high. The successful rate of the region offset 4 pixels is 0.9. In Biker video, the target is 16 pixels wide and 26 pixels high. When the region offsets 4 pixels, the successful rate is only 0.78. If the target gets smaller, the successful rate will be lower and these positive samples are not suitable. Therefore, our strategy try to change the offsets adaptively according to the size of target.

As shown in Figure 1, the target size is \(w \times h\). The largest offset value is \(l\) and the points of the farthest positive samples are on the circle whose radius is \(l\). We can calculate the offset value \(l\) of positive samples with the function:
\[
\left\{ \begin{array}{l}
wh - wn - hm + mn > 0.9 \\
wh + wm + hn - mn \\
m^2 + n^2 = l^2
\end{array} \right.
\]

In function (11), we can obtain \( l \approx \frac{1}{15} \min(w, h) \). This means the offset value is approximate \( \frac{1}{15} \) wide or high of the target. If the successful rate is lower than 0.5, we will get the negative samples. The strategy gets the positive and negative samples more representative, because these samples are collected according to the successful rate.

\[\text{Fig. 1. Sample selecting strategy}\]

D. Tracking Window Optimization

In CT algorithm, the size of the tracking window is fixed in the whole processing. When the size of target becomes smaller, the tracking window will contain much noise of background and the noise will influence the Bayes classifier causing the loss of target. If the size of target gets larger, it will influence the collection of negative samples and will also influence the Bayes classifier. Those can lead the loss or offset of the target. In order to solve this problem, we proposed the algorithm which can set the size of search window adaptively.

In the tracking process, the candidate regions are collected around the position of the previous frame and find the tracking location with the maximal classifier response. Then, sample the image around the location to get the positive and negative samples, and use samples to update the Bayes classifier. In this paper, when get the optimal location, the strategy will sample around the location with pixels range. Change the size of each candidate regions and find the maximal classifier response.

The center of the tracking window doesn’t change. Extract features in each region and classify by the Bayes classifier. The maximal classifier response is final tracking result. Then, extract the feature of the result and use these to update the classifier. Although the number of collection of the candidate regions increases, the number is still much lower than the CT algorithm and this strategy can improve the robustness.

IV. THE PROPOSED ALGORITHM: COMPRESSIVE TRACKING WITH ADAPTIVE SIZE OF SEARCH WINDOW (ASSWCT)

In this paper, we named the Compressive Tracking with Improved Speed ISCT and Compressive Tracking with Adaptive Size of Search Window ASSWCT. In ISCT, we add three modules, such as motion prediction module, motion distance module and sample modules. All of these modules can decrease the number of the samples and improve the speed of the operating. In ASSWCT we add re-sample module and window changing module to improve the robustness based on the ISCT.

A. Algorithm 1: ISCT

**Initialization:**
Select tracking target area in first frame manually, construct, and update the Bayes classifier.

**Input:**
Second video frame

1. Sample a set of image patches, then extract and compress these patches to get the feature vector \( V \).
2. Use the classifier \( H(v) \) in function (3) to classify each vector \( V \) and find the maximal classifier response as the tracking result.
3. Use four strategy: motion prediction, search window optimization, sample selecting optimization and tracking window optimization to sample a set of image patches to get the positive and negative samples, and then update the Bayes classifier.
4. Record the position of tracking.

**Tracking:**

The t-th video frame

1. Calculate the motion vector by the previous frames. Use motion prediction and search window optimization to sample a set of image patches, and then extract and compress these patches to get the feature vector \( V \).
2. Use the classifier \( H(v) \) in function (3) to classify each vector \( V \) and find the maximal classifier response as the tracking result.
3. Use sample selecting optimization strategy to sample a set of image patches to get the positive and negative samples and update the Bayes classifier.
4. Record the position of tracking.

**Output:**
Tracking position and classifier parameters.

B. Algorithm 2: ASSWCT

The initialization in the ASSWCT is similar to ISCT.

**Tracking:**

The t-th video frame

1. Calculate the motion vector by the previous frames. Use motion prediction and search window optimization to sample a set of image patches, and then extract and compress these patches to get the feature vector \( V \).
2. Use the classifier \( H(v) \) in function (3) to classify each vector \( V \) and find the maximal classifier response as the temporary tracking result.
3. Sample a set of image patches. Change the size of the search window, extract and compress these patches to get the feature vector \( v_1 \) by function (1). Use the classifier \( H(v) \) to classify each vector \( v_1 \) and find the maximal classifier response as the final tracking result.
4. Use the final size of the window and sample a set of image patches to get positive and negative samples by sample selecting optimization.
5. Update the Bayes classifier and record the tracking position.

**Output:**
Tracking position and classifier parameters.
V. EXPERIMENTS

A. Performance Evaluation

We use three metrics to evaluate the efficiency of the trackers. The first metric is the Successful Rate (SR). To measure the performance on a sequence of frames, we count the number of successful frames whose SR is larger than the given threshold. If the SR is larger than 0.5 in one frame, the tracking result is considered as a success. The second metric is the Center Location Error (CLE) which measures the Euclidean distance between the center of tracking box and center of ground truth box. It is widely used in evaluation metric on tracking precision. The third is computational complexity. We use Frames Per Second (FPS) to analyze the algorithm complexity. Higher FPS is benefit to the real application.

B. Experiment Setup

We evaluate tracking algorithms with challenging sequences on Online Tracking Benchmark (OTB) [18] dataset. OTB dataset contains 100 challenging sequences. It is worth noticing that the most challenging sequences are used for evaluation. The two trackers we proposed were compared with several state-of-art methods such as the CT, Fast Compressive Tracking (FCT) [19], and Convolutional Networks without Training (CNT) [20], Discriminative Scale Space Tracking (DSST) [21]. For fair comparisons, all the evaluated trackers are initialized with the same parameters. The algorithms are implemented in MATLAB, which runs on a Core i5 Quad-Core 2.5Ghz CPU with 4GB RAM. In Compressive Tracking algorithm, given a target location at the current frame, the positive samples search range is set as 4 pixels and get 45 positive samples. Set the search range between 8 and 30 pixels as the negative samples’ candidate regions and get 50 negative samples randomly. The search radius for object location detection is set as 20 pixels. The dimensionality of projected space is set as $n = 50$ and the learning parameter $\lambda$ is set as 0.85. In ISCT and ASSWCT algorithm, the size of search windows for positive and negative samples changes adaptively. The initialize search radius is set as 10 pixels. The other parameters are set as the same as the Compressive Tracking. In CNT algorithm, we will take the parameters configured in the literature. For fair comparisons, all parameters are fixed for all the experiments to demonstrate the robustness and stability. The parameters have relation with the accuracy and speed, however, we don’t consider the parameters optimization.

C. Experiment on OTB Dataset

1) Experiment on the sequence attributes

The videos in the benchmark dataset are annotated with attribute, which describe the challenges that a tracker facing in each sequence e.g., Illumination Variation, Scale Variation, Occlusion, Deformation, Motion Blur, Fast Motion, In-Plane Rotation, Out-of-Plane Rotation, Out-of-View, Background Clutters, Low Resolution. These attributes are useful for diagnosing and characterizing the behavior of trackers in such a large dataset, without analyzing each individual video. As space is limited, we report result for some attributes mentioned in OTB dataset in the following section.

Figure 2 is the results of the tracking, where the results of CT are marked in red and solid line, the results of ISCT are marked in yellow and dotted line, the result of DSST is marked in purple and dotted line, the results of ASSWCT are marked in black and dotted line and the CNT is marked in green and solid line.

![Figure 2. Track results](image)

![Track results](image)

Table 1 shows the quantitative results averaged over 10 times. Bold fonts indicate the best performance while the italic fonts indicate the second ones. The total number of evaluated frames is 7531.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>FCT</th>
<th>ISC</th>
<th>ASSWCT</th>
<th>CNT</th>
<th>DSST</th>
<th>FCT</th>
<th>ISC</th>
<th>ASSWCT</th>
<th>CNT</th>
<th>DSST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basketball</td>
<td>24.11</td>
<td>18.77</td>
<td>32.31</td>
<td>4644</td>
<td>43</td>
<td>0.78</td>
<td>0.91</td>
<td>0.98</td>
<td>0.98</td>
<td>0.07</td>
</tr>
<tr>
<td>Bird1</td>
<td>11.57</td>
<td>8.80</td>
<td>9.47</td>
<td>2593</td>
<td>16.5</td>
<td>0.21</td>
<td>0.20</td>
<td>0.39</td>
<td>0.31</td>
<td>0.02</td>
</tr>
<tr>
<td>Bird2</td>
<td>3.60</td>
<td>2.20</td>
<td>2.39</td>
<td>648</td>
<td>12.9</td>
<td>0.46</td>
<td>0.57</td>
<td>0.59</td>
<td>0.56</td>
<td>0.56</td>
</tr>
<tr>
<td>Board</td>
<td>25.98</td>
<td>16.52</td>
<td>20.70</td>
<td>4510</td>
<td>41.9</td>
<td>0.10</td>
<td>0.13</td>
<td>0.45</td>
<td>0.21</td>
<td>0.70</td>
</tr>
<tr>
<td>ClifBar</td>
<td>2.48</td>
<td>1.44</td>
<td>1.64</td>
<td>767</td>
<td>5.2</td>
<td>0.69</td>
<td>0.69</td>
<td>0.70</td>
<td>0.85</td>
<td>0.90</td>
</tr>
<tr>
<td>Crossing</td>
<td>3.0</td>
<td>1.56</td>
<td>1.59</td>
<td>795</td>
<td>3.8</td>
<td>0.95</td>
<td>0.97</td>
<td>0.83</td>
<td>1.00</td>
<td>0.98</td>
</tr>
<tr>
<td>Fish</td>
<td>9.76</td>
<td>4.76</td>
<td>5.54</td>
<td>3130</td>
<td>40.4</td>
<td>0.97</td>
<td>0.84</td>
<td>0.98</td>
<td>0.28</td>
<td>0.94</td>
</tr>
<tr>
<td>RedTeam</td>
<td>18.49</td>
<td>10.78</td>
<td>10.55</td>
<td>5316</td>
<td>18.2</td>
<td>0.21</td>
<td>0.52</td>
<td>0.59</td>
<td>1.00</td>
<td>0.56</td>
</tr>
<tr>
<td>Subway</td>
<td>5.0</td>
<td>2.68</td>
<td>2.62</td>
<td>1137</td>
<td>6</td>
<td>0.92</td>
<td>0.94</td>
<td>0.95</td>
<td>0.22</td>
<td>0.90</td>
</tr>
<tr>
<td>Suv</td>
<td>17.94</td>
<td>10.11</td>
<td>10.36</td>
<td>6067</td>
<td>56.2</td>
<td>0.14</td>
<td>0.63</td>
<td>0.59</td>
<td>0.99</td>
<td>0.98</td>
</tr>
<tr>
<td>Trellis</td>
<td>12.18</td>
<td>7.02</td>
<td>7.19</td>
<td>3650</td>
<td>89.4</td>
<td>0.37</td>
<td>0.30</td>
<td>0.57</td>
<td>0.29</td>
<td>0.99</td>
</tr>
<tr>
<td>Walking</td>
<td>20.27</td>
<td>14.85</td>
<td>13.84</td>
<td>2679</td>
<td>18.6</td>
<td>0.54</td>
<td>0.46</td>
<td>0.54</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td>Walking2</td>
<td>12.21</td>
<td>7.53</td>
<td>7.63</td>
<td>3224</td>
<td>39.4</td>
<td>0.39</td>
<td>0.39</td>
<td>0.38</td>
<td>1.00</td>
<td>0.90</td>
</tr>
<tr>
<td>Dudek</td>
<td>40.13</td>
<td>27.19</td>
<td>33.67</td>
<td>7550</td>
<td>348.3</td>
<td>0.78</td>
<td>0.92</td>
<td>0.93</td>
<td>0.67</td>
<td>0.90</td>
</tr>
<tr>
<td>Coupon</td>
<td>8.30</td>
<td>4.70</td>
<td>5.21</td>
<td>2209</td>
<td>368.9</td>
<td>0.51</td>
<td>0.74</td>
<td>0.75</td>
<td>0.34</td>
<td>0.98</td>
</tr>
<tr>
<td>Total</td>
<td>214.5</td>
<td>138.9</td>
<td>164.6</td>
<td>48918</td>
<td>1152.9</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Average</td>
<td>36</td>
<td>53</td>
<td>47</td>
<td>0.15</td>
<td>6.5</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

![Table 1. Success Rate and Tracking Speed](image)
In Table 1, ISCT processes all images using 138.9 seconds, ASSWCT costs 164.6 seconds, CT costs 216.5 seconds and FCT costs 214.5 seconds. ASSWCT reduces 51.9 seconds, almost one third of the whole tracking time. Furthermore, compared with CT, the successful rate of ASSWCT improves almost 31 percent. ASSWCT has great advantages in terms of real-time, although some sequences is lower than CNT and DSST. The value of FPS in ISCT and ASSWCT is nearly 50. In the terms of successful rate, CNT, DSST and ASSWCT all have good performance.

In Table 2, We compared Center Location Error in 15 challenge sequences in OTB dataset. ASSWCT performs the best than other trackers in average result. Considering three standard evaluation metrics, ASSWCT algorithm achieves the best or second best result in most sequences in terms of both success rate, center location error, and FPS.

## Table II. CENTER LOCATION ERROR

<table>
<thead>
<tr>
<th>Sequence</th>
<th>CT</th>
<th>FCT</th>
<th>ISCT</th>
<th>ASSWCT</th>
<th>CNT</th>
<th>DSST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basketball</td>
<td>123.41</td>
<td>119.1</td>
<td>99.29</td>
<td>7.49</td>
<td>87.64</td>
<td>256.7</td>
</tr>
<tr>
<td>Bird1</td>
<td>117.85</td>
<td>25.87</td>
<td>126.59</td>
<td>25.65</td>
<td>74.50</td>
<td>323.8</td>
</tr>
<tr>
<td>Bird2</td>
<td>43.25</td>
<td>53.02</td>
<td>17.92</td>
<td>17.75</td>
<td>44.66</td>
<td>18.3</td>
</tr>
<tr>
<td>Board</td>
<td>67.98</td>
<td>107.96</td>
<td>138.65</td>
<td>116.33</td>
<td>93.60</td>
<td>43.1</td>
</tr>
<tr>
<td>CliBar</td>
<td>3.64</td>
<td>4.28</td>
<td>2.95</td>
<td>3.98</td>
<td>9.22</td>
<td>4.8</td>
</tr>
<tr>
<td>Crossing</td>
<td>8.29</td>
<td>3.03</td>
<td>5.57</td>
<td>6.39</td>
<td>1.48</td>
<td>1.65</td>
</tr>
<tr>
<td>Fish</td>
<td>21.21</td>
<td>9.55</td>
<td>11.54</td>
<td>10.82</td>
<td>30.02</td>
<td>5.96</td>
</tr>
<tr>
<td>RedTeam</td>
<td>7.02</td>
<td>7.78</td>
<td>7.42</td>
<td>5.11</td>
<td>2.68</td>
<td>2.25</td>
</tr>
<tr>
<td>Subway</td>
<td>12.80</td>
<td>5.38</td>
<td>8.54</td>
<td>7.64</td>
<td>140.56</td>
<td>2.35</td>
</tr>
<tr>
<td>Suv</td>
<td>73.12</td>
<td>86.00</td>
<td>56.12</td>
<td>57.59</td>
<td>3.42</td>
<td>5.92</td>
</tr>
<tr>
<td>Trellis</td>
<td>56.86</td>
<td>55.59</td>
<td>49.72</td>
<td>48.80</td>
<td>39.61</td>
<td>4.45</td>
</tr>
<tr>
<td>Walking</td>
<td>5.5</td>
<td>5.51</td>
<td>7.41</td>
<td>6.01</td>
<td>1.81</td>
<td>1.67</td>
</tr>
<tr>
<td>Walking2</td>
<td>55.7</td>
<td>61.26</td>
<td>62.35</td>
<td>63.97</td>
<td>2.47</td>
<td>3.55</td>
</tr>
<tr>
<td>Dudek</td>
<td>27.54</td>
<td>30.32</td>
<td>23.99</td>
<td>23.06</td>
<td>43.70</td>
<td>14.6</td>
</tr>
<tr>
<td>Coupon</td>
<td>18.75</td>
<td>21.18</td>
<td>21.09</td>
<td>18.68</td>
<td>24.40</td>
<td>3.7</td>
</tr>
<tr>
<td>Average</td>
<td>42.42</td>
<td>46.54</td>
<td>35.61</td>
<td>28.08</td>
<td>39.98</td>
<td>46.05</td>
</tr>
</tbody>
</table>

VI. CONCLUSIONS

In this paper, we proposed an automatic compressive tracking based on motion prediction algorithm named ASSWCT. The algorithm predicts the motion direction depending on the previous frames and reduces the number of the candidate regions to improve the speed of the tracking processing. Furthermore, the algorithm can change the search range adaptively according to the motion distance calculated from the motion vector. The proposed method will reduce a lot of time in the tracking procedure. Furthermore, the algorithm combines meticulous search with rough search to improve the robustness of tracking. According to the tracking result in the first step which has located the tracking position roughly, ASSWCT resamples images and adjusts the size of the search window to improve the robustness of tracking. The simulation results show that ASSWCT significantly better than the similar tracking performance.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers and the editors for their valuable suggestions. This work was supported by the National Natural Science Foundation of China (Nos.61601248); the University Science Research Project of Jiangsu Province (No.16KJB510036); the Science and Technology Program of Nantong (No.MS12016025); Nantong University-Nantong Joint Research Center for Intelligent Information Technology (KFKT2017B04).

## REFERENCES


Off-line Handwritten Chinese Recognition and Improvement Based on Caffe Framework

1st Ning Bi  
School of Mathematics  
Sun Yat-sen University  
Guangzhou, China  
mcsbn@mail.sysu.edu.cn

2nd Haoli Sun  
School of Mathematics  
Sun Yat-sen University  
Guangzhou, China  
haolisun@outlook.com

3rd Jun Tan  
School of Mathematics  
Sun Yat-sen University  
Guangzhou, China  
mcstj@mail.sysu.edu.cn

Abstract—Off-line Chinese handwritten character recognition is an important area of pattern recognition. Because of the complexity of handwritten Chinese character, its recognition is more difficult than handwritten English recognition. This paper propose an approach of handwritten Chinese character recognition based on Caffe framework and HCL2000 data set. The article will list some general knowledge of the deep learning. Then we will use the LeNet network to train the data first, and then build several different neural networks based on the Caffe framework to test the handwritten Chinese character database. As result we compare the different networks, the performance and accuracy of those neural networks will be summarized.

Index Terms—Deep learning, Caffe, Handwritten recognition, Chinese character

I. INTRODUCTION

Handwritten recognition is an important branch in pattern recognition and has been developing for many decades. Popular machine learning methods such as support vector machine (SVM), back-propagation (BP) neural network (NN) were applied to handwriting recognition and achieve some results. But with the explosion of data especially vision data, these methods encounter bottleneck in both accuracy and efficiency until Prof. Geoffrey Hinton proposed a concept called deep learning and came up with improvement of model training methods in 2006 [1].

Obviously, handwritten numeral recognition is easier than handwritten English character recognition, but handwritten Chinese character recognition is much harder than others for several reasons: (1) there are much more characters than English. Comparing to 26 English characters, GB2312-80 code define up to 6,763 Chinese characters which is commonly used. (2) Unlike printed words, handwritten characters are freely and less normative. There are many kinds of font in Chinese such as Cao and Xing which are not easy to identify for many Chinese people. (3) Many different characters share similar characteristics like ‘己’/ji’ and ‘己’/yi’, ‘日’/ri’ and ‘白’/bai’.

Scholars have made some great achievements by using traditional classifiers [2]. Using discriminative feature learning (DFE) and discriminative learning quadratic discriminant function (DLQDF) the best recognition rate in two Chinese character database called CASIA-OLHWDB and CASIA-HWDB [3] are 95.28% (DB1.0, 4037 class), 94.85% (DB1.1, 3926 class) and 95.31% (ICDAR 2013 Competition DB, 3755 class). Currently mainstream handwriting recognition software recognition rate do not achieve high accuracy which under 90%.

With the development of deep learning, some problems in traditional classifiers can be solved and accuracy rate also is improved a lot. In a famous database called MNIST [4] which is a handwritten digit dataset, its recognition rate can be more than 99% by using LeNet model which contains 7 layers. Although digits are much easier to recognize than Chinese, its high recognition rate make LeNet model possible to be used in real market. Besides, recognition rate raised to 97.39% by Graham using spatially-sparse convolutional neural network (CNN) in ICDAR, a handwritten Chinese character recognition competition, in 2013. All of these show the potential in deep learning.

II. DEEP LEARNING AND CAFFE

Before starting our experiment, we would like to introduce some basic knowledge of deep learning. There are 3 main steps in CNN process:

- Forward propagation through the graph and get loss
- Back-propagation to calculate the gradients

Besides, a neural network is built by stacking several layers such as input/output layers, convolutional layers, pooling layers and activation layers, which will be also discussed.

A. Neural Network and Forward Propagation

A neural network contains many simple neurons, so the output of a neuron can be the input of neurons in next layer [5]. The structure of this network is shown in figure 1.

In this figure, the leftmost layer is called the input layer, and the rightmost layer is the output layer. The circles labeled "+1" are called bias units, so this network includes 3 input units (excluding bias units), 3 hidden units (excluding bias units) and 1 output unit.
In this example, \( n_l = 3 \) (denoting the number of layers), \( L_1 \) is the input layer, and layer \( L_{n_l} \) the output layer. \( W_{ij}^{(l)} \) to denote the parameter (or weight) associated with the connection between unit \( j \) in layer \( l \), and unit \( i \) in layer \( l + 1 \). \( b_i^{(l)} \) is the bias associated with unit \( i \) in layer \( l + 1 \). \( a_i^{(l)} \) denotes the output value of unit \( i \) in layer \( l \). So \( a_1^{(1)} = x_i \) in this example obviously. The neural network defines a hypothesis \( h_{W,b}(x) \) that outputs a real number. And the computation process is showed as follows:

\[
\begin{align*}
a_1^{(2)} &= f(W_{11}^{(1)} x_1 + W_{12}^{(1)} x_2 + W_{13}^{(1)} x_3 + b_1^{(1)}) \\
a_2^{(2)} &= f(W_{21}^{(1)} x_1 + W_{22}^{(1)} x_2 + W_{23}^{(1)} x_3 + b_2^{(1)}) \\
a_3^{(2)} &= f(W_{31}^{(1)} x_1 + W_{32}^{(1)} x_2 + W_{33}^{(1)} x_3 + b_3^{(1)}) \\
h_{W,b}(x) &= a_1^{(3)} = f(W_{11}^{(2)} a_1 + W_{12}^{(2)} a_2 + W_{13}^{(2)} a_3 + b_1^{(2)})
\end{align*}
\]

Denote \( z_i^{(l)} \) as the total weighted sum of inputs to unit \( i \) in layer \( l \), including the bias term (e.g. \( z_i^{(2)} = \sum_{j=1}^{n_l} W_{ij}^{(1)} x_j + b_i^{(1)} \)), so that \( a_i^{(l)} = f(z_i^{(l)}) \). We also can apply vector to computation, so the equations can be written as:

\[
\begin{align*}
z^{(2)} &= W^{(1)} x + b^{(1)} \\
a^{(2)} &= f(z^{(2)}) \\
z^{(3)} &= W^{(2)} a^{(2)} + b^{(2)} \\
h_{W,b}(x) &= a^{(3)} = f(z^{(3)})
\end{align*}
\]

This step calls forward propagation.

**B. Backpropagation and Parameter Updating**

Suppose we have a fixed training set \( \{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\} \) of \( m \) training examples. Given a training set of \( m \) examples, then the overall cost function is defined to be:

\[
J(W, b) = \frac{1}{m} \sum_{i=1}^{m} J(W, b; x^{(i)}, y^{(i)}) + \frac{\lambda}{2} \sum_{l=1}^{n_l-1} \sum_{s_l=1}^{n_l} \sum_{j=1}^{n_l+1} (W_{ji}^{(l)})^2
\]

\[
= \frac{1}{m} \sum_{i=1}^{m} \left( \frac{1}{2} \|y - h(W, b)(x^{(i)})\|^2 \right) + \frac{\lambda}{2} \sum_{l=1}^{n_l-1} \sum_{i=1}^{n_l} \sum_{j=1}^{n_l+1} (W_{ji}^{(l)})^2
\]

The goal is to minimize \( J(W, b) \) as a function of \( W \) and \( b \). We will initialize each parameter \( W_{ij}^{(l)} \) and each \( b_i^{(l)} \) to a small random value near zero, and then apply an optimization algorithm such as batch gradient descent such as Stochastic gradient descent and mini-batch gradient descent. In below our algorithm, one mini-batch which is randomly selected applies to updating parameters in every iteration.

Our rule for updating parameters is as follows.

\[
W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha \frac{\partial}{\partial W_{ij}^{(l)}} J(W, b) \\
b_i^{(l)} = b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(W, b)
\]

Where \( \alpha \) is the learning rate.

Here is the back-propagation algorithm:

- Perform a feedforward pass, computing the activations for layers \( L_2, L_3, \ldots \) up to the output layer \( L_{n_l} \).
- For each output unit \( i \) in layer \( n_l \) (the output layer), set \( \delta_i^{(n_l)} = \frac{\partial}{\partial z_i^{(n_l)}} \frac{1}{2} \|y - h_{W,b}(x)\|^2 = -(y_i - a_i^{(n_l)}) \cdot f'(z_i^{(n_l)}) \)
- For \( l = n_l - 1, n_l - 2, n_l - 3, \ldots, 2 \), for each node \( i \) in layer \( l \), set
  \[
  \delta_i^{(l)} = \left( \sum_{j=1}^{n_l+1} W_{ji}^{(l+1)} \delta_j^{(l+1)} \right) f'(z_i^{(l)})
  \]
- Compute the desired partial derivatives, which are given as:
  \[
  \frac{\partial}{\partial W_{ij}^{(l)}} J(W, b; x, y) = a_i^{(l)} \delta_j^{(l+1)} \\
  \frac{\partial}{\partial b_i^{(l)}} J(W, b; x, y) = \delta_i^{(l+1)}
  \]

Finally, after applying partial derivatives to updating rules above, parameters are updated for one time.

**C. Caffe Framework**

In this paper, all networks are built based on Caffe framework [6] which is a deep learning framework made with expression, speed, and modularity in mind and is created by Yangqing Jia and developed by Berkeley AI Research. We use Caffe under Linux environment. And some details for training and testing are presented in following.
**Local Receptive Field:** The first way to reduce parameters is local reception. We think it is not necessary for each neuron to perceive all pixels. It is possible to use local receptive field, then we get whole information by combining all local information.

![Fig. 2. Fully Connected and Locally Connected](image)

In this figure 2, the left is traditional fully connected neural net and the right is locally connected neural net. In the right graph, if one neurons in hidden layer connected with 10*10 pixels, only 1000000 * 100 parameters are needed.

**Parameter Sharing:** Parameter sharing is the second way to reduce parameter. Based on local connection, if 1000000 neurons share the same 100 parameters, the total number of parameters is 100 obviously. In order to describe how CNN works, We can pick up a sample from whole graph, for example a 5*5 pixels area, then apply features(parameters) learned from this area to other fields and get different activation value. This process can be seen intuitively in figure 3.

![Fig. 3. Parameter Sharing Process](image)

**Multi-Convolutional Kernel:** Like figure 3, there is only one kernel exist if the number of parameters is 9, which means only one feature is extracted. Obviously, it is not sufficient for feature extraction. Therefore, we apply more kernels during process of CNN so that we can learn more features. Each kernel can generate one feature map. From now on, we can understand better in following sections that introduction of main layers in CNN.

**D. Input Layer and Convolutional Layer**

In my experiment, the input are formed by graphs with pixel value of 64*64. This concept is easy to understand. Taking a 28*28 graph as an example. Shown in figure 4.

![Fig. 4. Input Neuron](image)

A convolutional layer follows the input layer. Each neuron in convolutional layers connects with 5*5 neurons in input layer. The area with 5*5 neurons calls local receptive field. Obviously, 25 connections correspond to 25 parameters \( W \) and a global bias value \( b \).

**III. Experiment**

**A. Introduction of HCL2000**

HCL2000 is one of the most influential handwriting Chinese character data set maintained by Beijing University of Posts and Telecommunications and can be used to study the law of handwritten Chinese character, shown in figure 5. It contains characters written by 1000 people, and each person writes 3755 Chinese characters. This paper divides all characters into 3755 classes, so each class consists of same Chinese character written by 1000 people.

![Fig. 5. HCL2000](image)

For example, the first class is 1000 types of Chinese character ‘啊’. So we get totally 3755000 characters. In our
Among 1000 persons, characters written by 300 persons are used as test dataset, and the rest as train dataset. Because the format of data in HCL2000 is 'hcl', so we want to convert it into 'jpg' format, where MATLAB is used to finish this job, then we save each graph as a joint photographic experts group(JPEG) format file. Since under Caffe framework the requirement for input data format is 'lmdb', we have to save name of all 'jpg' file and store their corresponding class label in one 'txt' file such as "1.jpg 1" that "1.jpg" is name of a JPEG image file and the last number 1 is its corresponding class label, which means the file "1.jpg" belongs to class 1. At last, we can use tools in Caffe framework to convert graphs and 'txt' document into 'lmdb' format.

B. Network with 1 ReLU Layer

We start from 2-convolutional-layer network with one ReLU layer. The structure of this network is as figure 6.

![Fig. 6. 2 Conv and 1 ReLU](image)

The recognition accuracy of this network is 89.6%, result may shown in table II.

In order to see whether accuracy rate can be improved by only adding convolutional layers, a network with 4 convolutional layers and 1 ReLU layer is built, and its structure is presented in figure 7.

![Fig. 7. 4 Conv and 1ReLU](image)

According to structure graphs, ReLU layer in two network is located between last pooling layer and output layer.

![Fully Connected](image)

![ReLU](image)

![Pooling](image)

![Convolutional layer](image)

![Pooling](image)

![Convolutional layer](image)

TABLE II

<table>
<thead>
<tr>
<th></th>
<th>Accuracy of Training</th>
<th>Accuracy of Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Conv+1 ReLU</td>
<td>89.6%</td>
<td>90.2%</td>
</tr>
<tr>
<td>4 Conv+1 ReLU</td>
<td>81.5%</td>
<td>82.6%</td>
</tr>
</tbody>
</table>

In this section, 3 2-ReLU-layer networks with different depth will be built. Their structure are presented in following figures. In this experiment, iteration is set to be 30000 where networks all converge.

So far, this paper get 3 networks with 4,5,6 convolutional layers seperately,see in table III shows their accuracy rate in training set and testing set. According to this table, the deeper network is, the higher accuracy we obtain. Especially, the last two networks perform well in HCL2000 that accuracy is higher than 90%. Also these networks are not as deep as GoogLeNet [7] which consists of 22 layers.

C. Other Network

Now we continue to add a ReLU layer to the network which already contains 6 convolutional layers and 2 ReLU layers.
layer. Its accuracy in testing dataset is 93.14%. Then, a dropout layer is added between last ReLU layer and fully connected layer. After training, the accuracy has been reached to 93.86% for this network.

Based on network in section 4.2, 2 ReLU layers are added to 6-convolutional-layer network, so now this network contains 6 convolutional layers and 4 activation layers. Its accuracy is 92.8% in testing dataset. Both networks get accuracy above 90% but are not as well as network in previous section.

D. Conclusion

During experiment, two of our networks perform worse than LeNet. And this paper want to increase accuracy rate by adding more layers including convolutional layers, activation layers or dropout layers. Although in first comparison the deeper network does not perform as well as the shallow one, more layers contribute to improving accuracy in following practice at least in our experiments. After adding ReLU layer up to 2, performance of the network with 4 convolutional layers is improved. Importantly, when 2 ReLU layers are fixed we add 1 convolutional layer at a time, accuracy rate is increasing obviously, see table IV. Finally, we got a network with accuracy in test set of 93.9%. This network performs best among all models built in this paper and meets initial expectation.

ACKNOWLEDGEMENT

This work was supported by the Guangdong Provincial Government of China through the “Computational Science Innovative Research Team” program and Guangdong Province Key Laboratory of Computational Science at the Sun Yat-sen University, and the National Science Foundation of China (11471012).

REFERENCES


TABLE III
COMPARISON-2

<table>
<thead>
<tr>
<th>Convolutional Layers</th>
<th>Accuracy of Training</th>
<th>Accuracy of Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 Conv+2 ReLU</td>
<td>85.5%</td>
<td>87.12%</td>
</tr>
<tr>
<td>5 Conv+2 ReLU</td>
<td>91.9%</td>
<td>91.3%</td>
</tr>
<tr>
<td>6 Conv+2 ReLU</td>
<td>93.4%</td>
<td>93.9%</td>
</tr>
</tbody>
</table>

TABLE IV
COMPARISON-3

<table>
<thead>
<tr>
<th>Convolutional Layers</th>
<th>Accuracy of Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 Conv+2 ReLU</td>
<td>87.12%</td>
</tr>
<tr>
<td>5 Conv+2 ReLU</td>
<td>91.3%</td>
</tr>
<tr>
<td>6 Conv+2 ReLU</td>
<td>93.9%</td>
</tr>
<tr>
<td>6 Conv+3 ReLU</td>
<td>93.14%</td>
</tr>
<tr>
<td>6 Conv+3 ReLU+1 Dropout</td>
<td>93.86%</td>
</tr>
<tr>
<td>6 Conv+4 ReLU</td>
<td>92.8%</td>
</tr>
</tbody>
</table>
Attention-Based Recurrent Neural Networks in Chinese Short Text Classification

1st Xin Lin  
School of Mathematics  
Sun Yat-Sen University  
Guangzhou, China  
linx56@mail2.sysu.edu.cn

2nd Jiahao Chen  
School of Mathematics  
Sun Yat-Sen University  
Guangzhou, China  
chenjh47@mail2.sysu.edu.cn

3rd Jun Tan  
School of Mathematics  
Sun Yat-Sen University  
Guangzhou, China  
mcsbj@mail2.sysu.edu.cn

4th Ning Bi  
School of Mathematics  
Sun Yat-Sen University  
Guangzhou, China  
mcsbn@mail.sysu.edu.cn

Abstract—Chinese short text classification is a crucial and challenging task. Compared to long text classification, the short one contains less information. Therefore, the high accuracy is relatively harder to achieve. Owing to the efficiency of recurrent neural network (RNN) in series problems, we try applying different variants of RNN to Chinese short text classification tasks. In addition, we firstly introduce the attention-based RNN into such field and test whether the performance has been improved or not. The experiment results reveal that such combination outperforms single RNN without attention in most situations. Especially, it possesses evident superiority in both accuracy and speed to integrate simple recurrent unit (SRU) and soft attention mechanism.

Index Terms—short text classification, recurrent neural network, SRU; soft attention

I. INTRODUCTION

Text classification refers to the process of automatic tagging of a text according to a certain pattern, which is extensively used in a many fields, such as text topic classification, spam filtering and sentiment analysis, etc. And the Chinese short text classification is a more challenging task. On one hand, the feature of short texts is much sparser. It’s very difficult to get a high accuracy based on the sparse information. On the other hand, unlike English and other similar languages, there is no delimiter between each two Chinese words. Undoubtedly it adds some extra difficulties in processing.

In recent years, the wave of deep learning is sweeping through every field, of which the natural language processing is a typical example. Especially the RNN has obtained great breakthrough achievements in various natural language processing tasks. In the same domain of deep learning, the attention mechanism has made its mark in recent years. Inspired by the attention characteristics of human, it’s made gratifying progress in both computer vision and natural language processing tasks.

However, we notice that few scholars focus on the exploration of RNN in Chinese short text classification, including LSTM [1], GRU [2], bidirectional RNN [3] and the latest SRU [4]. And less concentration are on the combination of RNN and attention mechanism. Motivated by such ideas, the paper tries to validate the efficiency of different RNN variants in Chinese short text classification problems. And it’s also our research emphasis to figure out whether the performance could be further improved with attention mechanism.

II. RELATED WORK

Due to the sparse feature of short texts, a great number of scholars focus on the feature extension. Meng et al. [5] gain search results by utilizing the web search engine, so more contexts could be added to the features. And some researchers introduce the external knowledge base to extend the features. Tang et al. [6] create the semantic dictionary based on the related domain knowledge to further improve the accuracy.

And some scholars are seeking for a better representation based on the original features. Sang et al. [7] introduce the word embedding methods to obtain the distributed representation of words, which helps to boost the information density of features. And Liu et al. [8] make a combination of word embedding and Wikipedia. The result appears to be more effective because of the semantic relevant concepts and word2vec tools.

Topic model is also a widely-used tool in short text classification. Zhang et al. [9] employ the LDA topic model to mine the latent topics in the short texts, and have the results further improved. Based on the N-gram language model and LDA model, Sun et al. [10] propose the topical N-gram methods to have the feature extended, which is also effective in the classification performance.

The deep learning methods are also used in the Chinese short text classification tasks. Wang et al. [11] propose a unified framework to expand short texts based on word embedding clustering and convolutional neural network (CNN). Zhou et al. [12] introduce the compositional RNN, which mainly considers the possible error caused by word segmentation. They propose a hybrid model of character-level and word-level features based on the long short-term memory (LSTM).

As for the attention-based methods, Bahdanau et al. [13] firstly introduce the attention mechanism to natural language processing fields. And the soft attention mechanism is also put forward at this time, which obtains better results in machine translation. Especially for long text classification problems, Yang et al. [14] propose the hierarchical attention networks. It has two levels of attention mechanism applied at the word and sentence level, enabling it to add more information to the
representation. As for the cross-lingual sentiment classification problems, Zhou et al. [15] propose attention-based LSTM network which learns the distributed semantics of the documents in both the source and the target languages.

However, there are still a limited number of researches applying deep learning methods to short text classification, especially the attention-based ones. Such findings provide us the goal and motivation of writing this paper.

III. APPROACH

A. LSTM

It’s very hard to train the simple RNN because of the gradient vanishing and gradient explosion problems, which makes it difficult to apply to the practical uses. Thus, Hochreiter et al. [1] propose the long short-term memory (LSTM), aimed at addressing the gradient vanishing problem. Memory cell is put forward to store the history information, which is controlled by the gated mechanism. Three gates are worth mentioning: input gate, forget gate and output gate. Owing to the extension above, LSTM could deal with the gradient vanishing problem efficiently, which makes it one of the most popular variants of RNN at present.

B. GRU

Due to the popularity and efficiency of LSTM, various modifications have been made on the basis of LSTM. The most famous and impactful one is gated recurrent unit (GRU) [2]. Similarly, it adopts the gated mechanism as LSTM does, but substitutes two gates for three. The two gates are called reset gate and update gate, which integrates the function of the three original ones. Furthermore, GRU simply uses the hidden state to store the history information without memory cell. Compared with LSTM, GRU owns less parameters which makes it faster and more robust. In the meanwhile, it inherits the advantages of LSTM that avoiding gradient vanishing problems. As a result, it’s widely used in many fields as well.

C. SRU

Although the gradient vanishing problem is almost resolved by LSTM and GRU, the training time is another crucial problem to cope with. Aimed at dealing with the series problems, RNN has an architecture of linear chain. Such structure makes it difficult to make full use of the advantages of GPU, which is the main cause of the low speed for training. Lei et al. [4] propose the simple recurrent unit (SRU) creatively. The most distinguishing feature of SRU is to drop the hidden state of last time step, which makes it to compute all the time steps in parallel. The adaptation could enhance the operational speed significantly because of the full use of GPU. SRU possesses two gates: forget gate and reset gate. The formulas are listed as follows:

\[
f_t = \sigma(W_f x_t + b_f) \quad (1)
\]

\[
r_t = \sigma(W_r x_t + b_r) \quad (2)
\]

Where \( W \) refers to the corresponding weights, \( b \) refers to the bias and \( \sigma \) refers to the sigmoid activation function.

Furthermore, it still keeps the memory cell compared with GRU, but it adds the highway connection based on LSTM. The formulas are as follows:

\[
\tilde{x}_t = W x_t \quad (3)
\]

\[
c_t = f_t \odot c_{t-1} + (1 - f_t) \odot \tilde{x}_t \quad (4)
\]

\[
h_t = r_t \odot g(c_t) + (1 - r_t) \odot x_t \quad (5)
\]

Where \( g(\cdot) \) refers to the activation function, and \( \tilde{x} \) refers to the transformed input at time step \( t \). The author gives clear experiments on a wide range of natural language processing tasks to verify the efficiency of the model. The results show that SRU could gain great boost in speed on the premise of no loss of accuracy. So it has a promising future to apply to the related fields.

D. Bidirectional Recurrent Neural Network

The common RNN simply considers the past information, but some series problems also depend greatly on the future information, such as the handwritten character recognition, the co-articulation in speech recognition, etc. In order to make use of the contexts, Schuster et al. [3] propose the bidirectional recurrent neural network, which is composed of two RNNs in opposite direction. One is from the first time step to the final step, while the other one is from the last time step to the first step. And thus we could fully utilize both of the past and future information. Certainly, the component of bidirectional RNN could be replaced by LSTM or GRU, such as bidirectional LSTM (Bi-LSTM) and bidirectional GRU (Bi-GRU).

E. Attention-Based Method

Since each word has different contribution to the final result, we might give each word a weight to measure such difference, which is the main idea of soft attention mechanism. In this way, our model could usually get a better performance because of focusing on the most important features.

In our paper, we compute the weight through the comparison between the word and the contexts. More concretely, we use the value of \( u_t^T u \) to measure the similarity between the output \( u_t \) and the context vector \( u \). We might as well view \( u \) as a vector representing the contexts, which could be trained with the network. Initially it’s a randomly generated parameter vector. After using the softmax function to normalize the similarity, we could gain the weight \( \alpha_t \) of each word. Taking \( \alpha_t \) as the weight and \( h_t \) as the input, we could obtain the final text vector \( s \) after the weighted sum. The architecture of attention-based model could be seen in Fig. 1. And the formulas are listed as follows:

\[
u_t = \tanh(W_u h_t + b_u) \quad (6)
\]

\[
\alpha_t = \frac{\exp(u_t^T u)}{\sum_t \exp(u_t^T u)} \quad (7)
\]

\[
s = \sum_t \alpha_t h_t \quad (8)
\]
the headlines length. As for the data set 1 and 2, we set the maximum sequence length of 40 and 55, respectively. In the meanwhile, we define the dimension of word embedding as 200.

Besides, we use the validation set to adjust the hyper-parameters of our models. And dropout didn’t appear to improve the performance, which even has a negative effect on accuracy in some cases. Therefore, we at last use only one layer of RNN. And the dimension of the neurons for two data sets are 200 and 128, respectively. We also use early stopping to prevent over-fitting problem.

C. Result and analysis

Our experiments could be roughly divided into two parts. The first part is the RNN without attention. We mainly consider five models, including LSTM, GRU, Bi-LSTM, Bi-GRU and SRU. And as for the second part, we add the attention mechanism to the models on the basis of the first part. And we could check the efficiency of attention through the comparison of the metrics. The results on two datasets are shown in Table I.

Firstly, we focus on the efficiency of the attention mechanism. As we can see in Table I, the performance has improved with the attention mechanism added. Although the accuracy has dropped a bit in some cases (e.g. GRU), such combination is still worth trying. The main reason is that accuracy and F1 score have increased in most instances while it simply consumes a little more time. Greatly to our surprise, however, the accuracy and F1 score have improved by more than 1% on both two data sets in attention-based SRU. It owns a relatively higher accuracy and faster processing speed at the same time.

Next, we make a comparison of common RNN and bidirectional RNN. According to the values of Table I, the accuracy and F1 score have dropped consistently with or without attention. The reason is probably the same as the preceding one. As we mentioned before, the advantage of bidirectional RNN is to make the full use of the contexts. But the sparse features provide less contexts to be used, which causes the drop of accuracy. And it takes much more computation time. Therefore, we believe the bidirectional RNN is not suitable for Chinese short text classification task.

V. Conclusion

In this paper, we try to validate the performance of various variants of RNN in Chinese short text classification task. And we firstly introduce the attention-based RNN into this field. Based on two public Chinese news headline data sets, we use the LSTM, GRU, Bi-LSTM, Bi-GRU and SRU to test the performance. Furthermore, we add attention mechanism on the basis of the last 5 models. Taking accuracy, F1 score and time as our measurement, we could draw the following conclusions:

- As for Chinese short text classification task, the soft attention mechanism helps to improve the results in most cases. Although it can’t guarantee the improvement in every situation, it’s still a worthwhile method in practice since it simply takes a little more time.
Attention-based SRU outperforms other variants of RNN in both accuracy and speed. It could be the most suitable attention-based RNN, in our views, in dealing with the Chinese short text classification problems.

Bidirectional RNN is not fit for coping with Chinese short text classification problem. Not only does it consume more computing time, but also it drops the accuracy.

This paper mainly verifies the efficiency of attention-based RNN in Chinese short text classification problem. However, we also believe that the accuracy still has some room for improvement. Therefore, our future work might include other combination of techniques, such as word2vec and other feature extension methods. Such work is worth further exploring.

ACKNOWLEDGMENT

This work was supported by the Guangdong Provincial Government of China through the “Computational Science Innovative Research Team” program and Guangdong Province Key Laboratory of Computational Science at the Sun Yat-sen University, and the National Science Foundation of China (11471012).

REFERENCES

Quaternion Linear Regression for Color Face Recognition

1st Yulong Wang
School of Information Science and Engineering
Chengdu University
Chengdu, China
wangyulong6251@gmail.com

2nd Yuan Yan Tang
Department of Computer and Information Science
University of Macau
Macau, China
yytang@umac.mo

3rd Cuiming Zou
School of Information Science and Engineering
Chengdu University
Chengdu, China
zoucuiming@gmail.com

4th Lina Yang
School of Computer, Electronics and Information
Guangxi University
Nanning, China
lnyang@gxu.edu.cn

5th Patrick Wang
College of Computer and Information Science
Northeastern University
Boston, USA
patwang@ieee.org

Abstract—Linear regression has shown an effective tool for face recognition in recent years. Most existing linear regression based methods are originally designed for grayscale image based face recognition. To extend linear regression for color images, we propose a novel color face recognition method by formulating the problem as a quaternion linear regression model. The proposed Quaternion Linear Regression Classification (QLRC) algorithm models each color facial image as a quaternionic signal and code multiple channels of each query color image in a holistic manner. Thus, the correlation among distinct channels of each query color image can be well preserved and leveraged by QLRC to further improve the recognition performance. The experiments on benchmark databases demonstrate the efficacy of the proposed method for color face recognition.

Index Terms—Quaternion representation, linear regression, color image, face recognition.

I. INTRODUCTION

Face recognition (FR) has been a hot research topic in pattern recognition for decades due to its wide applications in reality [1]–[3]. In the past two decades, a variety of FR methods have been proposed in the literature [2], [4], [5]. The nearest neighbor (NN) classifier may be the most popular FR method owing to its simplicity and efficacy. However, NN only uses one gallery image from each class to represent the query image, and fails to take into account the structure and distribution of data.

It has been found that facial images of a subject under varying illumination approximately lie in a low-dimensional subspace [6]. By taking advantage of such subspace structure of face images, many representative methods have been developed in recent years. For example, Inspired by the success of compressed sensing, Wright et al. [2] devised a sparse representation based classifier by coding the query image as a sparse linear combination of gallery images from all classes. Naseem et al. [4] put forward a linear regression classification (LRC) method for face recognition. To be specific, LRC belongs to the category of nearest subspace classifiers and tries to compute the minimal distance to the subspace spanned by all gallery images from each class. However, LRC is originally designed for grayscale image based face recognition and can only be applied on each color channel of color facial images individually and separately. Accordingly, the correlation information among distinct color channel may be severely discarded during coding the query image by LRC.

In this paper, we develop a quaternion linear regression classification approach (QLRC), which models each color image as a quaternion signal. Unlike LRC coding different color channels of the color image individually, the proposed QLRC approach codes multiple color channels of the query color image in a holistic manner by formulating the problem as a quaternion linear regression model. Therefore, QLRC can make use of the correlation information among multiple color channels to improve the recognition performance.
The remainder of the paper is structured as follows. In Section II, we introduce some basic facts of quaternion algebra. In Section III, we depict the proposed method for color face recognition. Section IV presents the experimental results. Eventually, Section V concludes the paper.

II. PRELIMINARIES

In this section, we introduce some basic facts of quaternion algebra. To enhance the readability, we first illustrate the notations used in this paper. To be specific, scalars, vectors and matrices are denoted by italic letters (e.g., x), boldface lowercase letters (e.g., x), and boldface upper-case letters (e.g., X), respectively. A quaternion is denoted by italic letters with a dot on the above (e.g., q̃).

The quaternion space denoted by ℍ was first proposed by Hamilton [7] as a generalization of the traditional complex space ℂ with three imagery units i, j and k. A quaternion q̃ ∈ ℍ is defined as

\[ q̃ = q_0 + q_1i + q_2j + q_3k, \tag{1} \]

where \( q_i \in \mathbb{R} \) (\( i = 0, 1, 2, 3 \)) and the three imagery units obey the quaternion rules

\[ i^2 = j^2 = k^2 = ijk = -1. \tag{2} \]

q̃ is said to be a pure quaternion if \( q_0 = 0 \). The most important characteristic of quaternion algebra is that the product of quaternions are noncommutative, i.e., \( \tilde{p}\tilde{q} \neq \tilde{q}\tilde{p} \). The conjugate \( \tilde{q} \) of \( q \) is defined as

\[ \tilde{q} = q_0 - q_1i - q_2j - q_3k. \tag{3} \]

The modulus |q̃| is defined by |q̃| = \( \sqrt{\tilde{q}\tilde{q}} \). Analogous to vector of real numbers, a quaternionic vector \( \tilde{v} \in \mathbb{H}^n \) is denoted as \( \tilde{v} = [v_1, \ldots, v_n] \) where \( v_i \) is the \( i \)-th coordinate of \( \tilde{v} \). The inner product of two quaternionic vectors \( \tilde{u} \) and \( \tilde{v} \) is defined by

\[ \langle \tilde{u}, \tilde{v} \rangle = \tilde{u}^H \tilde{v} = \sum_{i=1}^{n} \tilde{u}_i \tilde{v}_i \in \mathbb{H}, \]

where \( \tilde{u}^H = [\tilde{u}_1, \ldots, \tilde{u}_n]^T \) denotes the conjugate transpose of \( \tilde{u} \). Then, the \( \ell_2 \) norm of \( \tilde{v} \) can be written as ||\tilde{v}||_2 = \( \sqrt{\langle \tilde{v}, \tilde{v} \rangle} \). Analogously, a quaternionic matrix \( \tilde{M} \in \mathbb{H}^{m \times n} \) is a matrix of quaternion \( \tilde{M} = (M_{ij}) \) where \( M_{ij} \in \mathbb{H} \).

III. QUATERNION LINEAR REGRESSION CLASSIFICATION

Many recent studies [8], [9] have shown that quaternions can well adapt to color images. In this work, we model each RGB color image as a pure quaternionic matrix \( \tilde{I} \in \mathbb{H}^{m \times n} \) where \( m \) and \( n \) denote the number of rows and columns of \( I \), respectively. It can be written as

\[ \tilde{I} = 0 + I_ri + I_gj + I_bk, \]

where \( I_c \in \mathbb{R}^{m \times n} \) denotes the \( c \)-channel of \( I \) and \( c \in \{ r, g, b \} \) is defined for red, green and blue, respectively. For ease of formulation, it is common to stack the pixels in each channel \( I_c \) into a vector denoted as \( y_c \) for the \( c \)-channel. Accordingly, each color image \( \tilde{I} \in \mathbb{H}^{m \times n} \) is transformed into a quaternionic vector \( \tilde{v} \in \mathbb{H}^d \) (\( d = m \times n \)), which can be expressed as

\[ \tilde{v} = 0 + v_r i + v_g j + v_b k. \]

Consider a face recognition problem with \( K \) subjects corresponding to \( K \) classes. Assume that we are given \( n_k \) gallery color facial images \( \tilde{x}_k^1, \ldots, \tilde{x}_k^{n_k} \) from class \( k \) for \( 1 \leq k \leq K \). Here \( \tilde{x}_k^i \in \mathbb{H}^d \) denotes the \( i \)-th vectorized color image from class \( k \). For ease of statement, these images from class \( k \) are arranged as columns of a quaternionic matrix \( \tilde{X}_k = [\tilde{x}_k^1, \tilde{x}_k^2, \ldots, \tilde{x}_k^{n_k}] \in \mathbb{H}^{d \times n_k} \). Let \( \tilde{X} = [\tilde{X}_1, \tilde{X}_2, \ldots, \tilde{X}_K] \) be the quaternionic matrix of all gallery color images. The goal is to correctly recognize any new query color facial image \( \tilde{y} \in \mathbb{H}^d \).

Algorithm 1 Quaternion Linear Regression Classification

<table>
<thead>
<tr>
<th>Input:</th>
<th>Gallery color image matrix ( \tilde{X}_k \in \mathbb{H}^{d \times n_k} ), ( k = 1, \ldots, K ) and a query color image ( \tilde{y} \in \mathbb{H}^{d \times 1} ).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output:</td>
<td>identity(( \tilde{y} )).</td>
</tr>
<tr>
<td>1:</td>
<td>Compute the class-specific quaternionic coefficient vector ( \tilde{c}_k = \tilde{X}_k^+ \tilde{y} ), ( k = 1, \ldots, K ).</td>
</tr>
<tr>
<td>2:</td>
<td>Compute the class-specific reconstruction ( \tilde{y}_k = \tilde{X}_k \tilde{c}_k ), ( k = 1, \ldots, K ).</td>
</tr>
<tr>
<td>3:</td>
<td>Calculate the distance between the query color image and the class-specific reconstruction ( d_k(\tilde{y}) =</td>
</tr>
<tr>
<td>4:</td>
<td>Predict identity(( \tilde{y} )) = arg min ( _k ) ( d_k(\tilde{y}) ).</td>
</tr>
</tbody>
</table>

With the notations above, we depict the proposed QLRC (Quaternion Linear Regression based Classification) method for color face recognition as follows. Firstly, we attempt to represent the query color image \( \tilde{y} \) as a quaternion linear combination of the gallery images from each class. To this end, we minimize the squared class-dependent reconstruction error as below

\[ \min_{\tilde{c}_k \in \mathbb{H}^{n_k}} ||\tilde{y} - \tilde{X}_k \tilde{c}_k||^2_2, \quad k = 1, 2, \ldots, K. \tag{4} \]

The optimization problem above has a closed-form solution and can be solved efficiently [9]. Concretely, the optimal quaternion representation vector \( \tilde{c}_k \) of the problem above can be computed as

\[ \tilde{c}_k = (\tilde{X}_k^H \tilde{X}_k)^{-1} \tilde{X}_k^H \tilde{y} = \tilde{X}_k^+ \tilde{y}, \]

where \( \tilde{X}_k^H \) is the conjugate transpose of \( \tilde{X}_k \) and \( \tilde{X}_k^+ \) denotes the quaternion pseudo-inverse of \( \tilde{X}_k \) [9], [10]. Secondly, we compute the class-specific reconstruction of the query color image

\[ \tilde{y}_k = \tilde{X}_k \tilde{c}_k, \quad k = 1, \ldots, K. \]
Then the distance between the query image and the class-specific reconstruction can be calculated as

$$d_k(\hat{y}) = \|\hat{y} - \hat{y}_k\|_2, \quad k = 1, \ldots, K.$$  

According to the concept that patterns from a single-object class lie on a linear subspace [6], the query image $\hat{y}$ should be closer to the subspace of data in the same class as $\hat{y}$ than other subspaces associated with other classes. Therefore, the query color image $\hat{y}$ is assigned to the class minimizing the distance

$$\text{identity}(\hat{y}) = \arg \min_k d_k(\hat{y}).$$

Algorithm 1 summarizes the complete procedure of the proposed QLRC method for color face recognition.

### IV. Experiments

In this section, we carry out experiments on real data to verify the efficacy of the proposed approach for color face recognition. To begin with, we illustrate the used databases and the competing methods. Specifically, we consider two benchmark real-world face image databases, i.e., the AR database and the Caltech Faces Database. Fig. 1 shows some sample images from these two databases. For fair comparison, we also extend the grayscale image based NN and LRC methods to utilize color face images and the corresponding methods are referred to as ENN (Extended Nearest Neighbor) and ELRC (Extended Linear Regression Classification), respectively. Specifically, ENN first computes the minimal distance between the query image $\hat{y}$ and gallery images in each class

$$d_k^{\text{ENN}}(\hat{y}) = \min_{\hat{a} \in \mathbb{X}_k} \sum_{c \in C = \{r, g, b\}} \|y_c - a_c\|_2, \quad k = 1, \ldots, K,$$

where $y_c$ and $a_c$ denote the $c$ channel of the query image $\hat{y}$ and the gallery image $\hat{a}$, respectively. Then we assign $\hat{y}$ to the class with the minimal distance $d_k(\hat{y})$. As for ELRC, it first performs the traditional grayscale image based LRC [4] on each channel of color images separately and obtains the reconstruction $\hat{y}_{c,k}$ associated with each class in each channel. Then we calculate the sum of the class-dependent distance between $y_c$ and $\hat{y}_{c,k}$ in each channel

$$d_k^{\text{ELRC}}(\hat{y}) = \sum_{c \in C = \{r, g, b\}} \|y_c - \hat{y}_{c,k}\|_2, \quad k = 1, \ldots, K.$$  

Finally, the query image is assigned to the class yielding the minimal distance. It is worth noting that ELRC utilizes gallery images to represent the query image in each color channel separately and fail to take into consideration the correlation information among distinct channels. In contrast, the proposed QLRC method codes multiple channels of the query color image in a holistic manner and well preserve the correlation information among color channels for recognition.

#### A. AR Database

The AR database consists of over 4000 color frontal images of 126 subjects (70 men and 56 women). There are 26 images of each subject taken in two separate sessions with distinct lighting condition, expressions and occlusions. We utilize a popular subset containing 50 men and 50 women [2] to verify the efficacy of the proposed method. For each image, The facial images are cropped and normalized to $165 \times 120$ pixels. The original pixel values are used as features.

In the first experiment, we assess the recognition performance of QLRC and other competing methods using different feature dimensions, i.e., 48, 80, 192, 300. The corresponding downsampling ratio is 1/20, 1/15, 1/10, and 1/8, respectively. Specifically, we use the 13 images per subject taken in Session 1 for training while the other 13 images in Session 2 for testing. The recognition results of competing methods are compared and shown in Fig. 2. Note that the proposed QLRC method can yield higher recognition rates than other competing methods with varying feature dimensions.
B. Caltech Faces Database

The Caltech Faces database contains 450 color facial images of 27 subjects with different illumination and expression variations. As shown in Fig. 1, the images in this database are captured with distinct background. For evaluation, we first detect and extract face portions from the original images using the Viola-Jones face detector [11]. In the experiment, we randomly select 10% images per subject for training while the rest are used for testing. Analogous to the experiment for AR database, we also perform recognition with different feature dimensions, i.e., 48, 80, 192, 300. Fig. 3 shows the recognition results of different FR methods on the Caltech Faces database. As shown in Fig. 3, QLRC outperforms other comparison methods with notable performance gains in terms of recognition rates. This is attributed to the fact that QLRC treats and processes multiple color channels of the query color image holistically instead of coding them individually.

V. CONCLUSION

This paper presents a quaternion linear regression classifier with application to color face recognition. We model each color image as a quaternionic signal and code each query color image as a quaternion linear combination of gallery color images. Thus, the proposed method can well take advantage of the correlation information among multiple color channels of each query color facial image to further improve the recognition performance. This gives rise to the advantages of the proposed method compared with previous linear regression based face recognition approach.

REFERENCES

Computational Optimization for Fast and Robust Automatic Segmentation in Virtual Microscopy using Brute-force-based Feature Selection

1st Clément Bouvier
NEOXIA
Paris, France
MIRCen, CEA
Fontenay-aux-Roses, France

2nd Cédric Clouchoux
NEOXIA
Paris, France

3rd Nicolas Souedet
MIRCen, CEA
Fontenay-aux-Roses, France

4th Anne-Sophie Hérard
MIRCen, CEA
Fontenay-aux-Roses, France

5th Zhenzhen You
MIRCen, CEA
Fontenay-aux-Roses, France

6th Caroline Jan
MIRCen, CEA
Fontenay-aux-Roses, France

7th Philippe Hantraye
MIRCen, CEA
Fontenay-aux-Roses, France

8th Gilles Mergoil
NEOXIA
Paris, France

9th B. Helena Rodriguez
NEOXIA
Paris, France

10th Thierry Delzescaux
MIRCen, CEA
Fontenay-aux-Roses, France

Abstract—With the fast development of large computer infrastructures and the widening of their usage, the quantity of data produced and processed in all scientific fields is rapidly increasing. In histology, this increase is due to optical microscopes allowing generation of high-resolution images, below $1\mu m^2$ per pixel. Quantification of relevant biomarkers in structures of interest is crucial to understand biological processes and can be performed through image segmentation. Segmentation methods are increasingly relying on machine learning techniques, which use multiple information, or features, extracted from each pixel of the raw data. As a result, the amount of data to handle is becoming an obstacle to process large series of histological images at the whole brain level. Feature selection algorithms provide a framework to reduce the required information, but lack of algorithmic stability when dealing with real datasets.

This paper presents a methodology aiming at finding a small and stable set of features, to allow fast and robust segmentation of high-resolution histological whole sections. A brute-force searching is run to extract an exhaustive list of feature sets, and the quality of subsequent segmentations is evaluated using F-Score. The selection of features is performed among the extracted sets, minimizing a criterion based on the features yielding the best results. Our work led to select one optimal set of features composed of one color feature and one textural feature working in synergy, decreasing at the same time the computation load while preserving segmentation quality. The proposed methodology can be used as a potential benchmark to evaluate other feature selection algorithms in terms of computation load and stability.

Index Terms—Virtual microscopy, feature selection, brute-force, segmentation, color, texture, machine learning

I. INTRODUCTION

Histology allows visualization of physiological and pathological markers, improving our knowledge of biological processes. In the context of brain development, aging and neurodegenerative diseases, histology enables studies of physiology and potential therapeutical targets using specific markers [1]. The ability to quantify the expression of these markers is improving our fundamental understanding of underlying biological mechanisms [2]. Hence, virtual microscopy is a powerful tool for automatic and exhaustive quantification of stained tissues. Optical microscopes can reach in-plane resolution below $1\mu m^2$, allowing small biological objects detection such as neurons. Therefore, the histological images size has dramatically increased from megabytes (MB) to terabytes (TB), making exhaustive quantification in high resolution images, a task very challenging and demanding in computer resources.

From an image processing perspective, these techniques of automated tissue quantification increasingly rely on supervised machine learning (SML) algorithms. In comparison to more traditional algorithms such as thresholding, SML algorithms provide genericity due to the separation between training and classification processes. This in turn allows to specialize only the training stage according to the targeted staining, leaving the classification process unchanged [3]. In SML algorithms, feature information derived from the raw data is combined with information from the manual segmentation (annotations) during the training process. The validation dataset is then classified using the same set of features. The number of features can go up to hundreds [4], [5], drastically increasing the weight of data to be processed. The generated amount of data per biological study (up to 100 TB) makes the exhaustive quantification based on pixels extremely complex, if not impossible, to handle on traditional computing settings.

To overcome this complexity issue, a family of algorithms
called feature selection aims at increasing the classification quality and reducing the number of required features, by removing non-informative features [6]. These non-informative features can alter classification results. In real datasets, such features are predominant and lead to a severe lack of algorithmic stability [7]. Renewed interest in such stability is leading to new ways of seeking small and stable subsets for real data [8].

In this paper, we propose a method to find such a small and stable feature set, while preserving the quality of classification. We first describe each feature and present the classification algorithm. Our approach relies on the brute-force searching technique. In this approach, all the combinations of features derived from a single database are used to segment images. The different results are concatenated and then analyzed to find an optimal feature set based on feature distribution in terms of F-Scores. The goal is to select robust and informative features, based on their presence in the best combinations, and to avoid local optimum selection of feature. Computational time and memory cost before and after the selection are also presented. In addition, the presented results show interesting properties of features, especially the existence of a synergy between texture and color, as well as redundancy between features describing the same property.

II. FEATURE EXTRACTION AND MACHINE LEARNING

A. Feature description

1) Color features: A color space is a specific representation of color. Different color spaces have been described, each having its own specificities [3]. Red Green Blue (RGB) is the most used, and describes colors through their red, green and blue components. RGB is an easy-to-encode and linear color space. XYZ is a linear transformation of RGB, and is defined to fit the human perception of color. Y describes the luminance, Z the blue hue and X a combination of red and green hues. Components of both RGB and XYZ are highly correlated, making it non-optimal for complex classification processes. Hue Saturation Value (HSV) is a nonlinear transformation of RGB. An interesting aspect of HSV is hue separation from the other color properties, which makes HSV components less correlated than RGB ones. This space has singularities at low saturation levels, and a slight change in RGB values can result in a large shift in the transformed HSV values, which makes it unstable at low saturation levels. L*a*b* is a nonlinear transformation of XYZ, where L* describes luminance, a* the red to green hue component and b* the blue to yellow hue component. Similarly to HSV, L*a*b* has less correlated components than XYZ but is unstable at low saturation levels.

2) Textural features: Local textural information depends on the pixel intensity information inside a given pixel neighborhood [9]. Without prior knowledge of an optimal scale, textural features have to be computed at multiple amplitudes to avoid missing information. Multiple features were designed to describe textural information. Gabor filter is one of the most used filter [10], [11]. It lies on five different parameters, creating a large number of features to be computed and stored. For example, if four different angles and four different wavelengths are used and other parameters are static, 32 different features are generated at one amplitude and for one pixel, making Gabor filter difficult to use on massive datasets. Haralick features (Gray Level Co-Occurrence Matrix statistics) describe statistical arrangement of gray level intensities around a given pixel [9], [11]. Since computational complexity of co-occurrence matrix generation is high, Haralick features would take months to be calculated on a whole 150-gigabytes (GB) image (see section III. 1. for detailed biological image properties). In addition, the number of features ranges from 16 to 64 per pixel and for each amplitude, leading to the same computational issues presented for Gabor filters. Lastly, Local Binary Pattern (LBP) is a rotation and Gray-Scale invariant textural feature widely used, described by Ojala et al. [12]. Gray level intensities lying in a circle are thresholded by the gray level of the central pixel. The generated binary vector represents the concatenated thresholded values of the circle. This binary vector is minimized and converted into a decimal value. LBP is fast and straightforward to compute, with only one component stored, making this feature interesting in our context.

B. Supervised machine learning

SML methods are generally split in two steps: 1) training, to fit a model with the training dataset and 2) classification, to segment the entire dataset using the fitted model. This type of methods is used to tackle down many pattern recognition problems in virtual microscopy, such as tissue detection and characterization [4], [5]. When trained with annotations, SML methods can perform close to manual expert segmentations, which is one of the most used segmentation methods. Random Forest (RF) is one of the most popular SML methods. RF is an ensemble of fully grown decision trees. Each classification tree is trained based on bootstrap samples from the training dataset, only using a subset of randomly selected features [13]. Each tree provides a classification decision, and the majority of the decisions prevails. RF can fit nonlinear decision boundaries, making it particularly useful to separate tissue from background [14]. In the proposed approach, we chose to use a standard RF implementation (scikit-learn [15]) with a hundred trees of a maximal depth of four [16].

III. MATERIALS AND METHODS

A. Biological material

We used a brain of a 9 y.o. healthy macaque which was cut into 40-µm-thick sections [17]. Neuronal marker Neuronal Nuclei (Neu-N) was used to stain 134 sections. Two sections were digitized using a whole-slide imaging bright field scanner (Axio Scan.Z1, Zeiss), with a x20 magnification factor. The image in-plane resolution was 0.22x0.22µm. Each digitized section weighed approximately 150 GB. A neurobiologist extracted 100 sample images (512x512 pixels each) in different anatomical regions of the brain (Fig. 1). Sample images locations were chosen to get a comprehensive coverage of the different anatomical regions, as well as diversity in shape,
color and density of the neurons. Sample images were manually segmented in three classes: stained tissue, unstained tissue and background. Training and validation datasets were defined following three rules: 1) each dataset must have the closest number of images extracted from the right and the left brain hemispheres, 2) each dataset must have at least one image from each anatomical region and 3) each dataset must have the closest number of images from the two digitized sections. Resulting training and validation datasets were composed of 56 and 44 sample images, respectively.

B. Methodology

1) Design of the initial feature set: The initial vector of features included all of the color spaces described earlier in this paper (RGB, HSV, XYZ, L*a*b*) and LBP at four different scales. The first three LBP structuring element amplitudes were chosen to be equal to the estimated minimal size (11 pixels radius), the estimated mean size (40 pixels radius) and the estimated maximal size of a neuron (68 pixels radius), which correspond to neuron sizes from 2.5 to 15 \( \mu m \). These three LBP were called LBP11, LBP40 and LBP68 respectively. A fourth and last structuring element amplitude was set to twice the estimated maximal diameter of a neuron, to have a description of neuronal distribution (LBP134). We also computed the mean and the variance gray level intensity images, respectively called Mean and Var, with a square convolution kernel size equal to the estimated minimal size of a neuron to avoid relevant tissue loss. The resulting feature vector had a size of 18.

2) Brute-force searching: Brute-force searching is a powerful approach, which enumerates all the possible candidates for the solution to a given problem [18]. Despite its high computational cost, brute-force searching has the great advantage of always providing a solution to a given problem. In this study, we were searching for a stable and small set of features in order to reduce computational load as well as RAM usage. Multiple segmentations were performed on each possible combination from the initial feature vector. Each resulting segmentation, processed on the validation dataset was compared to the corresponding manual segmentation using F-Score criterion [19]. F-Scores computed on each combination of features were then concatenated in a single table for further analysis. To test the stability of selected subsets, a brute-force searching was performed in a cross validation setting, by switching training and validation datasets. For the 18-size feature vector, the number of combinations generated by brute-force searching was 262,143. To reduce processing time, the brute-force searching step was run for all combinations with a feature vector size ranging from 1 to 4 features, resulting in 4,047 different segmentations to process.

3) F-scores table analysis: The different combinations were sorted using the harmonic mean between F-Scores of classes 1) stained tissue (neurons) and 2) merged unstained tissue and background. For each feature, a position vector derived from the sorted table was then computed. Position vector represented the list of positions in the sorted table where the feature appeared. Each position vector was scored with a Median Position Value (MPV), which was defined as the median of position vector. The lower the MPV of a feature, the denser this feature representation in the highest F-Scores. To facilitate representation of MPV, we computed relative MPV percentages. Relative MPV percentages was related to a specific mean MPV (general mean, mean through color components). The feature with the lowest MPV (equivalent to lowest relative MPV percentage) contributed the most to the best results, and therefore could be selected.

4) Optimal feature definition: We defined the optimal feature vector iteratively by selecting at each round the optimal feature minimizing the MPV, and by subsequently considering in the next iteration only the combinations including the previous selected features. This strategy rapidly led to a decrease in the number of combinations to be investigated. The selection stopped whenever the addition of a new feature decreased the resulting segmentation quality. A flowchart of this method is presented in figure 2.

C. Computational environment

The presented work was developed and integrated in the BrainVISA collaborative software platform (http://brainvisa.info/web/index.html) [20]. Software libraries enabling input and output partial access and distributed CPU computation (somaWorkflow [21]) were used. Computations were performed on Ubuntu 14.04 LTS 64-bits with Intel
**IV. RESULTS**

**A. Brute-force searching**

Among the 4,047 combinations, three effects could be observed when the combination size increased (Fig. 3). First, the mean of F-Scores got closer to the F-Score obtained with the initial set including 18 features (0.87). Second, the spread of F-Scores decreased. Third, both minimal and maximal F-Score values increased. We also computed the percentage of combinations presenting a F-Score above 0.8, which was considered as a correct quality for automatic segmentation. For each feature combination size, results were as follows: 53% for size of 1, 71% for size of 2, 85% for size of 3 and 96% for size of 4. Most of 4-size combinations had F-Scores similar to the initial feature set F-Score.

We were able to compare color spaces with relative MPV percentages of color spaces components and their spreading (Fig. 4). The relative MPV percentages were then ranked from lowest (the best) to highest (the worst) value (Fig. 4 top). For each color space, we computed the arithmetic mean relative MPV percentages and the spread between the lowest and the highest relative MPV percentage ranks between its components (Fig. 4 bottom). XYZ had the lowest mean MPV (-21.88%), followed by RGB (-1.29%) and L*a*b* (3.35%). RGB had the highest rank spread and its Red channel had the highest relative MPV percentage. Red channel was mostly present in the worst combination. Saturation channel had the lowest relative MPV percentage of HSV color space. XYZ was the best color space in our configuration.

**B. Feature selection**

The results of selection obtained for direct (DV) and cross validation (CV) are presented in table I. In both validation settings, the first selected features were LBP68 and Z. In addition, the third feature slightly altered the resulting segmentation quality. Based on these results, we chose [LBP68 Z] as the preferred subset. Moreover, this subset was also the selected one in cross validation setting, confirming its stability. F-Scores were similar for the selected subset and the initial set of 18 features (Tab. II).

**C. Computational load and time**

Brute-force searching computation was performed in a parallel computing environment [16]. For this study, we used 16 cores in parallel. Using this setting, brute-force computation was achieved within 5 days. At the segmentation stage, using the stable subset [LBP68 Z] led to a decrease in RAM usage of a factor of 9, in comparison with the initial set of 18 features. Overall, with an initial data weight of 150 GB, the amount of...
computed data to be processed was decreased from 650 GB (18 features) to 75 GB (2 features). From a computational time perspective, a single 150 GB image was segmented in 50 minutes with the optimal feature set, compared to 7 hours and 45 minutes using 18 features. As a consequence, if computations were performed on a single core sequentially, computational times would have been multiplied by a factor of 16, leading to 80 days for brute-force searching, 5 days for 18-feature-based segmentation and 15 hours for 2-feature-based segmentation.

V. DISCUSSION

As expected, the F-scores increased with the number of features, confirming similar observations that increasing the number of features improved the quality of segmentation results [4], [5]. Moreover, the spread of min-max F-Score distribution became narrower with the increase of the feature vector size, reducing the number of outliers and increasing the probability of finding a robust and suitable set of features. In addition, with four features, 96% of combinations had a F-Score above 0.8, suggesting that the segmentation results might not significantly benefit from a vector size larger than 4 features. This maximum size is dependent of used dataset [22].

We also compared all the color spaces used in the initial set of features. As observed in figure 4, XYZ was preferred over HSV. In the presented data, saturation levels appeared to be low, and results showed that the saturation channel S had the lowest MPV among the three components of HSV. The previous demonstration of HSV instability at low saturation levels [3] strengthened our choice. RGB and L*a*b* had similar mean MPV. However, the Red component of RGB was present in the worst combinations, significantly disturbing segmentation as shown in figure 4. Consequently, L*a*b* was preferred to RGB as an optimal color space. Brute-force searching showed that color spaces close to human perception (such as XYZ and L*a*b*) were the color spaces of choice for the segmentation of histological images. This result has been highlighted in previous studies [3], [4], and was also consistent with the fact that histology staining was originally designed to be discerned by human eyes. Similarly, upcoming work is aiming at comparing supplementary texture features of interest, including Gabor filters, Haralick features and LBP.

The proposed methodology succeeded in finding a compromise between robustness, computational and memory load. Of note, the two main properties of images, color and texture, were represented in the selected subset and were working in synergy. In the proposed optimization process, the segmentation quality decreased after two rounds before increasing again. This suggests that non-informative features, which alters classification results [6], are added to the data model in the third round. Disturbances due to non-informative feature addition can be easily detected and ultimately removed. Finally, our method showed stability of the brute-force searching in cross validation setting. However, stability is not guaranteed. Cross validation has to be performed each time to assess stability of selected feature subset.

On a more general note, feature selection algorithms are classified in three groups: filter methods, wrapper methods and embedded methods [22]. Our method is included in the latter group. Although embedded methods are costly in computational load compared to filter and wrapper methods, they also have been shown to be strongly dependent on the used classification methods [22]. However, contrary to other methods, embedded ones take into account the final classification result as a selection condition, providing more reliable results. A future work will consist in using brute-force-based feature selection as a standard to benchmark other feature selection algorithms in terms of memory cost, computation time and stability. Such a benchmarking aims at providing a guideline to choose the most adapted feature selection algorithm and can be applied to different staining or species tissue samples to be segmented.

Although brute-force searching computation can be very costly in computational time and resources, two factors are advocating for its use in our context [16]. First, brute-force searching can be highly parallelized, taking advantage of High Performance Computing (HPC) resources now available.

<table>
<thead>
<tr>
<th>Table I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature selection results in direct validation (DV) and cross validation (CV) settings. In column C, features in bold represent common features between direct and cross validation selections. In column F, feature vector position in bold highlights the decrease in position between second and third rounds.</td>
</tr>
<tr>
<td>a b c d e f</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>DV</td>
</tr>
<tr>
<td>CV</td>
</tr>
<tr>
<td>DV</td>
</tr>
<tr>
<td>CV</td>
</tr>
<tr>
<td>DV</td>
</tr>
<tr>
<td>CV</td>
</tr>
<tr>
<td>DV</td>
</tr>
<tr>
<td>CV</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison of F-Scores with the initial set of 18 features and the selected set of features [LBP68 Z], Class Neuron represents stained tissue. Class Other represents merged unstained tissue and background. Initial set is the feature set with 18 features.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Classes</th>
<th>DV</th>
<th>CV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>0.87</td>
<td>0.97</td>
</tr>
<tr>
<td>[LBP68 Z]</td>
<td>0.86</td>
<td>0.97</td>
</tr>
</tbody>
</table>
Consequently, computational time can be drastically decreased by adding more computing resources. Second, even with a five-day-long brute-force searching, the processing time of one 150 GB image segmentation was reduced from 7 hours and 45 minutes to 50 minutes by using the selected stable subset, showing that brute-force searching approach is worth of using from 2.5 TB of raw data. Since the images of histological sections of a macaque whole brain correspond approximately to 15 TB of raw data, exhaustive exploratory analysis of one or more brains could greatly benefit from brute-force-based feature selection [2].

VI. CONCLUSION

In summary, we propose a new methodology to find a small and stable feature set preserving the quality of microscopy image segmentation. The selected subset couples one color feature and one textural feature, working in synergy. This subset is robust to permutation of training and validation datasets. In addition, its reduced size favors the processing of massive datasets produced today following histology (whole-slide imaging, 3D microscopy). From a preclinical imaging perspective, the presented methodology can be potentially applied to other histological stainings or animal models, providing a robust framework to optimize feature selection and produce reliable tissue segmentation. Ongoing work is aiming at implementing a fully industrialized platform based on HPC infrastructure, which will be of great interest for large volumetry preclinical studies, in both academic and industrial environments.

ACKNOWLEDGMENT

This work is supported by the French national funds PIA2 program under contract No. P112331-3422142 (3DNeuroSecure project: http://www.3dneurosecure.com/) and NEOXIA.

REFERENCES

Novel Graph-based Image Segmentation: Application to Medical Imaging

1st Sarada Prasad Dakua  
Department of Surgery  
Hamad Medical Corporation  
Doha, Qatar  
sdakua@hamad.qa

2nd Gajendra Kumar Mourya  
Department of Biomedical Engineering  
North Eastern Hills University  
Shillong, India  
gkmourya@nehu.ac.in

3rd D. Bhatia  
Department of Biomedical Engineering  
North Eastern Hills University  
Shillong, India  
dbhatia@nehu.ac.in

4th Julien Abinahed  
Department of Surgery  
Hamad Medical Corporation  
Doha, Qatar  
jabinahed@hamad.qa

5th Abdulla Al-Ansari  
Department of Surgery  
Hamad Medical Corporation  
Doha, Qatar  
aalansari1@hamad.qa

Abstract—Low contrast medical images are the result of variety of reasons, starting from low power imaging machine to the nature of diseases. This paper describes a novel method to segment low contrast clinical images. The method is split in two parts: 1) contrast enhancement- wavelet based stochastic resonance is used to enhance the contrast, and 2) a graph-based method is used where a new weighting function is proposed in order to be used in the graph-based method. The method is tested on publicly available datasets and promising results with regards to accuracy are obtained when compared with the ground-truth images.

Index Terms—MRI, CT, image segmentation, graph-based method

I. INTRODUCTION

Liver cancer is the fifth-most common malignancy in men and the eighth-most common in women worldwide. Segmentation is the key in any treatment planning whether it is surgical or non-surgical. This is because the clinicians first wish to check the shape and size of the effected human organ. Therefore, one can find a number of segmentation methods, however, it continues to be a challenging task mainly due to noise [1]. As a result, the clinicians find it even difficult to segmentation manually. Furthermore, since manual segmentation is tedious and time consuming, it is therefore always desired to have a method that requires minimum user interventions. Unlike traditional methods that usually follow filtering procedures degrading the imaging quality [9], we utilize the noise present in the image itself in a constructive manner to enhance the contrast using Stochastic Resonance (SR) theory [7], [8]. Despite numerous efforts to develop automated segmentation methods ([3]-[6]), without doubt, graph-based methods [14] have attracted much attention since they have the potential to segment the object without heavy reliance on learned/encoded priors. However, weighting function in these methods heavily influence their performance. Therefore, we have explored and introduced a new function in a graph-based approach.

II. MATERIALS AND METHODS

We have used two sets of data, one for left ventricle (CMR) and the other for liver. Short-axis MR image sequences have been obtained from subjects having various pathologies by the Hospital for sick children in Toronto [16]. All subjects are...
under the age 18 with each image sequence consisting of 20 frames and 8-15 slices. The spacing between slices are 6\text{mm} - 13\text{mm} and resolution of each image slice is 256 x 256 with pixel spacing 0.93\text{mm} - 1.64\text{mm}. The corresponding ground truth images are collected from the same hospital. The liver data set has been collected from MICCAI 2007 Grand Challenge workshop [17]. The pixel spacing, inter slice distance, and resolution of the image sequence are 0.55\text{mm} - 0.8\text{mm}, 1\text{mm} - 3\text{mm}, and 512 x 512, respectively. The ground truth data are also provided by the workshop organizer.

Our proposed method has 2 main blocks: (1) input image contrast enhancement and (2) graph-based image segmentation.

A. Contrast Enhancement

SR concept itself is a nonlinear phenomenon in a double well system (Fig. I) where the presence of noise is vital for optimal system accomplishment [11]. Importantly, the process of transforming the low contrast images into high contrast takes place without introducing any blocking artifacts. There are two main components that are supposed to be maximum in order SR to occur at a certain noise level, they are signal to noise ratio (SNR) and input-output correlation [7]. The noise must allow the signal to cross the threshold and therefore, low noise (images) does not have any role providing maximum SNR. There have been some attempts to use SR in spatial and Fourier domains [7], [10], however, in this work, we aim to use wavelet transform since discrete wavelet transform (DWT) has the potential to provide spatio-frequency localization enabling to enhance features at each sub-band and at different scales.

**DWT based Contrast Enhancement** In this methodology, the input data is subjected to 2D DWT that transforms an \( A \times B \) sized image \( I(c,d) \), \( c,d \) representing pixel coordinates, as:

\[
W(l_0,u,v) = \frac{1}{\sqrt{AB}} \sum_{c=1}^{A} \sum_{d=1}^{B} I(c,d) \phi_{l_0,u,v}(c,d) \quad (1)
\]

\[
W^s(l,u,v) = \frac{1}{\sqrt{AB}} \sum_{c=1}^{A} \sum_{d=1}^{B} I(c,d) \psi^s_{l,u,v}(c,d) \quad (2)
\]

where \( s \in \{H,V,D\} \) and \( l_0 \) is a starting scale chosen arbitrarily. The approximation of \( I \), at scale \( l_0 \), is defined by the coefficients \( W(l_0,u,v) \), whereas \( W^s(l,u,v) \) add horizontal, vertical and diagonal details for \( l \geq l_0 \): \( A = B = 2^L \), \( l = 0,1,2,...,J - 1 \), and \( u,v = 0,1,2,...,2^L - 1 \).

The SR is applied to the detail and approximation coefficients to obtain the stochastically enhanced coefficients that are set in DWT domain as \( W^s(l,u,v)_{DSR} \) and \( W(l_0,u,v)_{DSR} \). Then the enhanced image \( I_{\text{enhanced}} \) can be obtained by inverse discrete wavelet transform (IDWT):

\[
I_{\text{enhanced}} = \frac{1}{\sqrt{AB}} \sum_{u,q} \sum_{v} X_{\phi}(l_0,u,v) \phi_{l_0,u,v}(c,d) + \frac{1}{\sqrt{AB}} \sum_{s \in \{H,V,D\}} \sum_{l \geq l_0} \sum_{p} \sum_{v} X_{\psi}^s(l,u,v) \psi^s_{l,u,v}(c,d) \quad (3)
\]

Fig. 1. (a) Stochastic resonance in the biphasic form: The dashed arrow represents the transition under noise, crossing over the shaded threshold C [7]; (b) Gaussian, DoG and DroLoG weighting functions, (c) Difference of CFs due to Gaussian and DroLoG.
Selection of Double Well Parameters

This parameter is obtained by maximizing the SNR and it is maximized when $a = 2\sigma_0^2$, where $\sigma_0$ is the noise level in input image. If the restoring force is the gradient of bistable potential function $U(x)$, a maximum additive periodic signal $R = F\sin\omega t$ is required to stabilize the potential function. Therefore, $R = -\frac{dU}{dx} = -ax + bx^3; \frac{dR}{dx} = -a + 3bx^2 = 0$ resulting $x = \sqrt{a/3b}$. A maximum force $\sqrt{\frac{4a^3}{27b}}$ can be obtained at this value of $x$ resulting $F\sin\omega t < \sqrt{\frac{4a^3}{27b}}$. Keeping the left term of $F\sin\omega t < \sqrt{\frac{4a^3}{27b}}$ to maximum with $F$ as 1, $b < \frac{4a^3}{27}$ for weak signals.

B. Graph-based Image Segmentation

In any graph-based method, the user needs to mark foreground and background by placing two sets of seeds and each seeded region returns a probability map distinguishing foreground from background.

Influence of Weighting Function

In a graph-based approach, the 2D image is considered as an electric network, where the pixels are nodes. The weighting function maps the change in image intensities to edge weights [14]. It is noteworthy to mention that the performance highly depends on the choice of this weighting function.
function that reflects the correlation of two nodes. Grady reports that Gaussian weighting function is perfect in clean images, but the performance degrades significantly when applied on medical images with low contrast and high noise level [15]. Laplacian operator is a suitable tool for edge detection, however, it alone produces double edges at the boundary resulting pseudo-segmentation. Therefore, we aim to utilize the salient features of Laplacian operator at the fundamental level combining with Gaussian operator as given below.

**Derivative of Laplacian of Gaussian Weighting Function**

A blurred edge $v(x)$ is a combination of an exponential and a step function: $v(x) = \begin{cases} e^{-\frac{(x-G)^2}{2\sigma^2}}, & x \leq t \\ 1, & x > t \end{cases}$ where $t$ represents the window width and $\sigma$ the extent of the blurring, $(0 \leq \sigma \leq S, S \in \mathbb{R})$. $\sigma = \begin{cases} S, & \text{blurred edge} \\ 0, & \text{sharp edge} \end{cases}$. The quantity of blurring increases with increase in $S$. $E_{c}$ is a characteristic function (CF) that characterizes if a certain weighting function is suitable for proper segmentation of certain medical image data. The CF is defined by a WF and an edge, then $E_{c}^{w}$ (characteristic function with respect to edge, $v$, and a WF, $w$) determines the potential of the WF in accommodating the edge pixels in final segmentation. In other words, a CF determines whether a particular WF is suitable in accurately detecting an edge from little given edge information (due to high value of $\sigma$). A decreased magnitude of $E_{c}^{w}$ indicates that the edge of the input image is more likely to be segmented.

**Characteristic function** : Conceptually, the CF $E_{c}^{w} = \int_{0}^{2t} v(x) w(x) dx$, therefore, lower magnitude of $E_{c}^{w}$ is always desirable. The derivative of Laplacian of Gaussian (DroLoG) weighting function is defined as $w_{DroLoG} = \frac{x^2 - 2\sigma^2}{\sigma^4} e^{\frac{-x^2}{2\sigma^2}} - \frac{x^2 - 2\sigma^2}{\sigma^2} e^{\frac{-x^2}{2\sigma^2}}$, $\sigma$s are usual standard deviations. The magnitude of the difference between two CFs (due to Gaussian (G) and DroLoG) is expressed as:

$$h = E_{c}^{DroLoG} - E_{c}^{G}$$

where:

$$h = \sqrt[2]{\frac{1}{2\pi \sigma^4}} \left[ \frac{3(1-t^2) - \frac{1}{\sigma^2} + \frac{1}{2\sigma^2} t^2}{\sigma^2} \right] - \frac{1}{2} \left( \frac{1}{2\sigma^2} \right)^2$$

$$\left( \Gamma\left(2, \frac{t^2}{\sigma^2}\right) - \Gamma\left(2, \frac{1}{2\sigma^2}\right) \right) - \sqrt{\frac{1}{2\pi \sigma}} \sqrt{\frac{\pi}{8}} \left( \left( \frac{2t}{\sigma^2} \right)^2 - \left( \frac{1}{\sigma^2} + \frac{1}{2\sigma^2} \right) \left( \frac{t^2}{\sigma^2} \right) \right)$$

$$erf\left( \sqrt{\frac{t^2}{2\sigma^2}} + \frac{2t}{2\sigma^2} \right) - \frac{2t}{\sigma^2} \sqrt{\frac{\pi}{8}}$$

$$\left\{ erf\left( \sqrt{\frac{2}{\sigma^2}} \right) \right\} + 2\sigma \sqrt{\frac{\pi}{8}} \left\{ erf\left( \sqrt{\frac{2}{\sigma^2}} \right) \right\} -$$

where $erf$ is the error function and $\Gamma(a,x) = \int_{x}^{\infty} e^{-t^2}a^{-1} dt$. The influence of the CF on blurring can be explored by studying (4) and varying the values of $t, \sigma$ individually. $\sigma$ is varied as shown in Fig. 1 and for $t > 3$, $|E_{c}^{DroLoG} - E_{c}^{G}|$ is negative indicating DroLoG as the better alternative than Gaussian weighting function for better segmentation.

**Summary**: 1) First the input image is contrast enhanced, 2) the seeds are placed on the contrast enhanced liver image for marking the liver from the background, 3) segmentation method using DroLoG is applied to segment the liver.
Fig. 3. Segmented slices of a subject with the proposed method using (a-d) Gaussian weighting function, and (e-h) DroLoG weighting function. (i-l) Liver segmentations using DroLoG weighting function.

### TABLE I
SEGMENTATION METHODS VALIDATION ON CARDIAC DATA WHEN THE WEIGHTING FUNCTION IS GAUSSIAN AND DROLOG.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Method</th>
<th>$I_h$</th>
<th>FOM</th>
<th>$H_d$</th>
<th>Subject</th>
<th>Method</th>
<th>$I_h$</th>
<th>FOM</th>
<th>$H_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>Gaussian</td>
<td>3.81</td>
<td>0.68</td>
<td>5.98</td>
<td>Fourth</td>
<td>Gaussian</td>
<td>4.34</td>
<td>0.64</td>
<td>6.25</td>
</tr>
<tr>
<td></td>
<td>DroLoG</td>
<td>3.48</td>
<td>0.83</td>
<td>5.16</td>
<td></td>
<td>DroLoG</td>
<td>4.04</td>
<td>0.91</td>
<td>5.06</td>
</tr>
<tr>
<td>Second</td>
<td>Gaussian</td>
<td>5.23</td>
<td>0.69</td>
<td>6.28</td>
<td>Fifth</td>
<td>Gaussian</td>
<td>4.60</td>
<td>0.77</td>
<td>6.32</td>
</tr>
<tr>
<td></td>
<td>DroLoG</td>
<td>5.14</td>
<td>0.84</td>
<td>5.60</td>
<td></td>
<td>DroLoG</td>
<td>4.32</td>
<td>0.84</td>
<td>5.61</td>
</tr>
<tr>
<td>Third</td>
<td>Gaussian</td>
<td>4.03</td>
<td>0.87</td>
<td>6.07</td>
<td>Sixth</td>
<td>Gaussian</td>
<td>4.60</td>
<td>0.77</td>
<td>6.32</td>
</tr>
<tr>
<td></td>
<td>DroLoG</td>
<td>3.46</td>
<td>0.82</td>
<td>5.23</td>
<td></td>
<td>DroLoG</td>
<td>4.13</td>
<td>0.83</td>
<td>5.40</td>
</tr>
</tbody>
</table>

III. RESULTS AND DISCUSSION
The results after contrast enhancement of the liver and left ventricle images are provided in Fig. 2. The segmentation results due to Gaussian and DroLoG weighting function in graph based method are given in Fig. 3. From Fig. 1, it is evident that DroLoG has moderate decreasing rate if compared with Gaussian indicating that distant pixels with similar image
intensities can be grouped together to build the desired probability map and final segmentation (as shown in Fig. 3(e)-3(h)), while Gaussian weighting function results pseudo segmentations. Papillary muscles [2] are well segmented using DroLoG which is not the case in case of Gaussian weighting function. The epicardial contour is derived by an active contour model [15] that takes blood pool boundary of left ventricle resulted by the above method as its initial contour. DroLoG function when evaluated on liver data, it gives promising output as shown in Fig. 3(i)-3(l). The results are objectively evaluated by three parameters (in Table. I) viz., Intra-region uniformity ($I_h$) [18], Pratt’s Figure of merit (FOM) and Hausdorff’s distance ($H_d$). Higher value of $I_h$ and FOM and lower value of $H_d$ are desired for good segmentation. The Fig. I also shows that DroLoG performs better than Gaussian weighing function.

IV. CONCLUSION

Liver disease is a leading cause of death worldwide. The surgical treatments of liver cancer rely on the high-resolution imaging such as CT for the liver surgery planning process. In this paper, a wavelet based SR method is presented for contrast enhancement of input CT image. Subsequently, the influence of weighting function in graph based methods is discussed and a new weighting function, DroLoG, is proposed. This weighting function has moderate decreasing rate as compared to Gaussian weighting function indicating its potential to segment the low contrast, high noise, and pathological liver images. In future, we intend to extensively validate this method on other human organs and imaging modalities such as MRI, PET, etc.

ACKNOWLEDGEMENT

This work was partly supported by NPRP Grant #NPRP 5-792-2-328 from the Qatar National Research Fund (a member of the Qatar Foundation).

REFERENCES


Fractal-windowed based Empirical Mode Decomposition Scheme for Protein Sequence Analysis

Lina Yang1,2, Pu Wei1,2,*, Lin Bai1,2,*, Cheng Zhong1,2, Jianjia Pan3, Patrick Wang4, Yuan Yan Tang5,
1School of Computer, Electronics and Information, Guangxi University, Nanning 530004, China
2Guangxi Colleges and Universities Key Laboratory of Parallel and Distributed Computing, Nanning, China
3Hong Kong Pan American Image Information Technology Limited, Hong Kong SAR, China
4Computer and Information Science, Northeastern University, Boston, USA
5Beijing Advanced Innovation Center for Big Data and Brain Computing BDBC, Beihang University, Beijing, China
∗Corresponding Author
Email: lnyang@gxu.edu.cn, baliin@gxu.edu.cn, chzhong@gxu.edu.cn, jianjiapan@gmail.com, yytang@umac.mo

Abstract—The protein sequence similarities can be used to detect the biological function and the interaction of the proteins. The authors are concerned in this paper with the fractal-windowed based algorithm of protein sequence analysis, which combines the fractal dimension with the empirical mode decomposition. We consider each protein sequence as a signal sequence, apply empirical mode decomposition and suitable fractal dimension to generate a new encoding feature. Each protein sequence can be decomposed into certain intrinsic mode functions. A fixed windows fractal dimension (FWFD) applied to each IMF(Intrinsic Mode Functions) and original signal, the features of the protein sequences can be obtained. The experimental results state clearly that the feature extracted by the proposed method is better than that of the existing methods including the pure empirical mode decomposition.

I. INTRODUCTION

The protein sequence comparison methods have attracted considerable attention in the biologist community. Researchers find that the proteins with similar amino acid sequences normally have closer functions. Hence, to get the proper and reliable methods to analysis the similarity from the large amount of protein sequence database has become one of the most important areas of protein study. There have been extensive study and application of it. The key mechanism is to discover the similar segments, structure-function domains or sequences with similar function.

Huang [1] proposed a nonlinear signal decomposition technique—Empirical Mode Decomposition (EMD), has been widely used in biomedical research and protein sequence analysis. Zhu S.M et al [2] presented a layered comparison method which is based on empirical mode decomposition, and protein sequences are characterized by the intrinsic mode functions with a new cross-correlation formula. Fractal geometry is a non-integer and useful concept in describing dynamical structure [3]. It is also a useful method to indicate varies in both amplitude and frequency of a signal [4]. The subject of fractal dimension is proposed in [3], the fractal dimension has been applied in the area of image processing, e.g. image compression [4], texture analysis [5], image encoding [6], etc. More recently, it refers to be a useful method for studying the proteins. Some applications of fractal dimension to genome sequences are presented and references cited there in [7], [8]. Hu [9] applied the fractal method for interpolating the 2D discrete points for amino acids, to obtain the protein sequences curve, and the authors computed the local fractal based similarity to form a global feature vector. Our research group [10] proposed a hybrid method which is based on fractal dimension and with wavelet transform, to analysis the protein sequence similarity.

This paper aims at exploring the fractal dimension calculation in protein sequence analysis for similarity comparison. We emphasize that although the empirical mode decomposition has such wide applications, it has not been used with fractal algorithm for protein sequence comparison in bioinformatics. An outline of this paper is as follows. In Section 2, we outline the basic concepts of empirical mode decomposition and the approximated theories of fractal dimension that we shall consider. Our proposed algorithm will be discussed in Section 3. Numerical experiments results are summarized in Section 4, showing that our proposed approach are more efficient than the existing methods. The conclusion is drawn in the final section.

II. EMPIRICAL MODE DECOMPOSITION AND FRACTAL DIMENSION CALCULATION

A. Empirical mode decomposition

N. E. Huang etc. [1] designed a non-linear and nonstationary data analysis method - Empirical mode decomposition (EMD). It presented that the EMD decomposes any signal into components, which is called intrinsic mode function (IMF). IMF forms the Multi-scale signal from higher frequency to lower frequency. It satisfies two conditions to define: the signal is the number of zeros and poles up to a difference of 1; any point of signal, its local maximum/minimum point are determined on the zero mean under the envelope. Once the extrema are identified, all of the local maxima are connected by a cubic spline line as the upper envelope.

The sifting process serves two purposes: to eliminate riding waves; and to make the wave-profiles more symmetric. Toward this end, the sifting process has to be repeated more times.
In the second sifting process, \( h_1 \) is treated as the data, then 
\( h_1 - m_{11} = h_{11} \).

The sifting process has two effects: to eliminate riding waves; and to smooth uneven amplitudes. To guarantee that the IMF components retain enough physical sense of both amplitude and frequency modulations, a criterion must be defined for the sifting process to stop. This can be accomplished by limiting the size of the standard deviation, \( SD \), computed from the two consecutive sifting results as

\[
SD = \sum_{i=0}^{T} \frac{|h_{1(k-1)}(t) - h_{1k}(t)|^2}{h_{1(k-1)}^2(t)}
\]

The following predetermined criteria can stop the sifting process: the component \( c_n \) or the residue \( r_n \) becomes so small, it is less than the predetermined value of substantial consequence, or the residue \( r_n \), becomes a monotonic function, no more IMF can be extracted. We finally obtain \( X(t) = \sum_{i=1}^{n} c_i + r_n \). Thus, a decomposition of the data decompose into \( n \)-empirical modes, residue \( r_n \), which can be either the mean trend or a constant.

EMD is the first step of the Hilbert Huang transform, which is a powerful tool in the time series analysis of nonlinear and non-stationary processes. Signal noise discrimination was based on EMD\[14\]. IMF obtained through EMD is from high and non-stationary processes. Signal noise discrimination was proposed for ECG signals and respiratory signals using the EMD method in articles \[14\], \[15\], \[16\], \[17\], \[18\], \[19\], \[20\].

### B. Petrosian’s Algorithm

Petrosian applied a fast fractal dimension estimation based on Katz’s FD \[11\]. A signal is recorded by subtracting continuous values from the original signal. Due to the sequence of subtractions, a binary sequence is created assigning +1 if the subtraction result is positive, or -1 if the result is negative, respectively. The FD is then computed as:

\[
D = \frac{\log(n)}{\log(n) + \log\left(\frac{n}{\n_{\Delta}}\right)}
\]  
(1)

where \( n \) is the number of steps of a signal (total values minus 1) and \( \n_{\Delta} \) is the number of sign changes in the binary sequence.

The computational cost of Petrosian’s FD is low since it only calculates the changes of successive values. It’s obvious that in Equation (1), a larger \( \n_{\Delta} \) means a larger fractal dimension, because larger \( \n_{\Delta} \) means more complex curve.

### C. Higuchi’s Algorithm

Assume an one-dimension signal \( \{x_i\}, i = 1...N \). By defining a measurement scale \( k \), we separate it to \( k \) new signal denoted as:

\[
x^k_m = \left\{ x(m), x(m+k), x(m+2k), ..., x(m + \left\lfloor \frac{N - m}{k} \right\rfloor k) \right\}, m = 1...k
\]  
(2)

where \( \left\lfloor \cdot \right\rfloor \) is the floor operation, \( m \) is the starting position and \( \left\lfloor \frac{N - m}{k} \right\rfloor \) indicates how many terms in \( x^k_m \). Given an example of \( k = 3 \) and \( N = 11 \):

\[
X = \{ 2, 5, 4, 7, 7, 8, 3, 7, 7, 1, 1 \}
\]

\[
x_1^3 = \{ x(1), x(4), x(7), x(10) \} = \{ 2, 7, 3, 1 \}
\]

\[
x_2^3 = \{ x(2), x(5), x(8), x(11) \} = \{ 5, 7, 7, 1 \}
\]

\[
x_3^3 = \{ x(3), x(6), x(9) \} = \{ 4, 8, 7 \}
\]

Then we can calculate the approximated length by measurement scale \( k \) and started from \( m \):

\[
L_m(k) = \left( \sum_{i=1}^{\left\lfloor \frac{N - m}{k} \right\rfloor} \left| x(m + ik) - x(m + (i - 1)k) \right| (N - 1) \right) / \left( \left( \frac{N - m}{k} \right)^2 \right), m = 1...k
\]

(3)

where \( (N - 1)/\left( \frac{N - m}{k} \right)^2 \) is the normalization term. Dividing by \( \left( \frac{N - m}{k} \right)^2 \) is to calculate the average difference between each sample in the signal. After multiplying \( N - 1 \) in the numerator, it is the approximated length of the signal by given measurement scale \( k \). Continue the previous example:

\[
L_1(k) = 4.07 \quad L_2(k) = 2.96 \quad L_3(k) = 2.78
\]

At last, the average length of the signal is given by:

\[
L(k) = \frac{1}{k} \sum_{m=1}^{k} L_m(k)
\]

(4)

which is under the measurement scale \( k \). In the previous example, we can get:

\[
L(3) = 3.27
\]

While given \( k = 1...K \), where \( K \) is a pre-defined maximal value of \( k \), the fractal dimension \( d^* \) of the signal which is also the slope of \( \log(L(k)) \) and \( \log(\frac{1}{k}) \) can be evaluated by a simple least square method:

\[
d^* = \arg \min_d \sum_{k=1}^{K} \left( d \log \left( \frac{1}{k} \right) - \log \left( L_k \right) + c \right)^2
\]

(5)

where \( c \) is the bias. In our example, we can calculate for \( k = 1..3 \):

\[
L(1) = 23.00 \quad L(2) = 7.88 \quad L(3) = 3.46
\]

\[
d^* = 1.7505
\]

To sum up, Katz’s algorithm is a fast edition of Petrosian’s. Both of them can approximate the fractal dimension without using the scaling value. Higuchi’s algorithm can get the more accurate result than the other two algorithms. In this paper, our proposed protein comparison algorithm will adopt Higuchi’s algorithm as the calculation of fractal dimension.
### Table I: Feature values of each amino acid

<table>
<thead>
<tr>
<th>Name</th>
<th>Abbreviation</th>
<th>code</th>
<th>Residue Weight</th>
<th>H-value</th>
<th>pKa</th>
<th>pKb</th>
<th>pI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alanine</td>
<td>Ala</td>
<td>A</td>
<td>71.08</td>
<td>1.8</td>
<td>2.34</td>
<td>9.69</td>
<td>6.00</td>
</tr>
<tr>
<td>Cysteine</td>
<td>Cys</td>
<td>C</td>
<td>103.15</td>
<td>2.5</td>
<td>1.96</td>
<td>10.28</td>
<td>5.07</td>
</tr>
<tr>
<td>Aspartic acid</td>
<td>Asp</td>
<td>D</td>
<td>115.09</td>
<td>-3.5</td>
<td>1.88</td>
<td>9.60</td>
<td>2.77</td>
</tr>
<tr>
<td>Glutamic acid</td>
<td>Glu</td>
<td>E</td>
<td>129.12</td>
<td>-3.5</td>
<td>2.19</td>
<td>9.67</td>
<td>3.22</td>
</tr>
<tr>
<td>Phenylalanine</td>
<td>Phe</td>
<td>F</td>
<td>147.18</td>
<td>2.8</td>
<td>1.83</td>
<td>9.13</td>
<td>5.48</td>
</tr>
<tr>
<td>Glycine</td>
<td>Gly</td>
<td>G</td>
<td>157.05</td>
<td>-0.4</td>
<td>2.34</td>
<td>9.60</td>
<td>5.97</td>
</tr>
<tr>
<td>Histidine</td>
<td>His</td>
<td>H</td>
<td>137.14</td>
<td>-3.2</td>
<td>1.82</td>
<td>9.17</td>
<td>7.59</td>
</tr>
<tr>
<td>Isoleucine</td>
<td>Ile</td>
<td>I</td>
<td>113.16</td>
<td>4.5</td>
<td>2.36</td>
<td>9.60</td>
<td>6.02</td>
</tr>
<tr>
<td>Lysine</td>
<td>Lys</td>
<td>K</td>
<td>128.18</td>
<td>-3.9</td>
<td>2.18</td>
<td>8.95</td>
<td>9.74</td>
</tr>
<tr>
<td>Leucine</td>
<td>Leu</td>
<td>L</td>
<td>113.16</td>
<td>3.8</td>
<td>2.36</td>
<td>9.60</td>
<td>5.98</td>
</tr>
<tr>
<td>Methionine</td>
<td>Met</td>
<td>M</td>
<td>131.20</td>
<td>1.9</td>
<td>2.28</td>
<td>9.21</td>
<td>5.74</td>
</tr>
<tr>
<td>Asparagine</td>
<td>Asp</td>
<td>N</td>
<td>114.11</td>
<td>-3.5</td>
<td>2.02</td>
<td>8.80</td>
<td>5.41</td>
</tr>
<tr>
<td>Proline</td>
<td>Pro</td>
<td>P</td>
<td>97.12</td>
<td>-1.6</td>
<td>1.99</td>
<td>10.60</td>
<td>6.30</td>
</tr>
<tr>
<td>Glutamine</td>
<td>Glu</td>
<td>Q</td>
<td>128.13</td>
<td>-3.5</td>
<td>2.17</td>
<td>9.13</td>
<td>5.65</td>
</tr>
<tr>
<td>Arginine</td>
<td>Arg</td>
<td>R</td>
<td>156.19</td>
<td>-4.5</td>
<td>2.17</td>
<td>9.04</td>
<td>10.76</td>
</tr>
<tr>
<td>Serine</td>
<td>Ser</td>
<td>S</td>
<td>87.08</td>
<td>-0.8</td>
<td>2.21</td>
<td>9.15</td>
<td>5.68</td>
</tr>
<tr>
<td>Threonine</td>
<td>Thr</td>
<td>T</td>
<td>101.11</td>
<td>-0.7</td>
<td>2.09</td>
<td>9.10</td>
<td>5.60</td>
</tr>
<tr>
<td>Valine</td>
<td>Val</td>
<td>V</td>
<td>99.13</td>
<td>4.2</td>
<td>2.32</td>
<td>9.62</td>
<td>5.96</td>
</tr>
<tr>
<td>Tryptophan</td>
<td>Trp</td>
<td>W</td>
<td>186.22</td>
<td>-0.9</td>
<td>2.83</td>
<td>9.39</td>
<td>5.89</td>
</tr>
<tr>
<td>Tyrosine</td>
<td>Tyr</td>
<td>Y</td>
<td>163.18</td>
<td>-1.3</td>
<td>2.20</td>
<td>9.11</td>
<td>5.66</td>
</tr>
</tbody>
</table>

Fig. 1: Process of proposed method.

### III. Proposed Approach

In this paper, we consider a protein sequence as a signal sequence, and apply a new approach which is based on empirical mode decomposition and Higuchi’s fractal dimension to analyze and construct the feature vector of a protein sequence. The hydropathy feature value was selected among the various features, a vectorial representation of the protein sequence is achieved by using the hydropathy feature. By using the Empirical mode decomposition to process the protein sequences, each protein sequence can be decomposed into certain intrinsic mode functions. Using sliding window is to deal with the values within the fixed window length of a signal. For values in each fixed window, we calculate the Higuchi’s fractal dimension. An overall description of the proposed method is shown in Figure 1.

We believe that the application of empirical mode decomposition in protein sequence analysis can achieve a more prominent role. In order to divide the sequence more quickly and accurately, in this paper, we propose innovatively that the empirical mode decomposition and the fractal dimension are linked and applied to protein sequence alignment. Through the sample training to extract features, the sequence has been more accurately divided.

Table 1 shows that scientists have observed lot of features of each different amino acid so far [12]. It lists Residue weight, Hydropathy value (H-value), pKa(protein kinase A), pKb(protein kinase B), pl(isoelectric point). The hydropathy value as a popular feature has been widely used [13]. That is a very important property which helps find the similarity between protein structures. The one-to-one mapping of amino acids and their hydropathy values are shown in Table I.
IV. EXPERIMENTS

The graphical description of the protein sequences is presented in Figure 2. In this figure, eight sequences are provided, including four in the first group, and the rest in the second group. The number of training samples determines the accuracy of the results. The larger the difference is, the more the minimum number of training samples is required, and the large difference sequences require fewer training samples to achieve the goal.

It is the features vectors of EMD with fractal dimension algorithm for protein sequences. The graphical description of the features for the original signal and the layers of IMF are illustrated in Figure 3 and Figure 4. In Figure 4, we select IMF1, IMF4, IMF7, and IMF10 to describe the differences. According to the experiments, number of ND5 (NADH dehydrogenase subunit 5) protein sequences in the NCBI (National Center for Biotechnology Information) database are used in the experiment.

In Figure 5, we can see that protein sequences have been correctly identified by our method. The serial number behind the sequence name [1], [2] indicates the category to which it belongs. The combination of empirical mode decomposition and fractal dimension has been successfully applied to the analysis of protein sequences.

Here we use discrete wavelet transform to analyze the sequence. The initial part digitizes the protein sequence and then dwt transforms the sequence. After this we calculating the similarity of clustering tree map. In the end we generated a result. Obviously, EMD processing in this place is better than wavelet.

As shown in the Figure 6, the wavelet can obtain some features of the sequence, but there is still some error. The trained empirical mode decomposition method can obtain more accurate results to a large extent. As the number of training samples increases and the number of decomposition layers increases, the obtained results will become more accurate.

V. CONCLUSION

To begin with the analyze of large amount of protein sequence data precisely and effectively, it expressed in the terms of fractal dimension based empirical mode decomposition, measure the similarity of protein sequence using hydropathy characteristic.
Fig. 5: The phylogenetic tree of ND5 protein sequences constructed by our approach.

Fig. 6: The phylogenetic tree of ND5 protein sequences constructed by DWT method.

The main contribution is that we proposed a novel model, which is based on empirical mode decomposition and fractal dimension to get the feature vectors of protein sequence, and measure the sequence similarity. As what we have anticipated, the experimental results illustrate that the performance of proposed model is better than that of pure EMD model. With the increase of the number of training samples and the increase of the number of layers in EMD, the classification result achieves more stability.

ACKNOWLEDGMENT

This work was financially supported by the Guangxi Natural Science Foundation No. 2017GXNSFBA198226 and 2016XNSFBA380163, the National Natural Science Foundation of China under Grant No.61462005, Scientific Research Foundation of Guangxi University No. XGZ160483, and by the Higher Education Undergraduate Teaching Reform Project of Guangxi with No. 2017JGB108 and 2017JGA124, it also supported by the project entitled-Beta Protein Sequence Analysis using Sliding window and Time-Frequency method.

REFERENCES


A new ConvNet architecture for heartbeat classification

V. Dentamaro  
College of Computing  
Georgia Institute of Technology  
Atlanta, GA, USA  
vincenzo@gatech.edu

D. Impedovo  
dep.t of Computer Science  
University of Bari  
Bari, Italy  
donato.impedovo@uniba.it

G. Pirlo  
dep.t of Computer Science  
University of Bari  
Bari, Italy line 4:  
giuseppe.pirlo@uniba.it

G. Vessio  
dep.t of Computer Science  
University of Bari  
Bari, Italy  
gennaro.vessio@uniba.it

Abstract—This paper presents a new convolutional neural network architecture for heartbeat classification. The architecture, that uses a reduced number of layers, with respect to other CNN used for heartbeat classification, is able to achieve high accuracy in heartbeat classification following the AAMI recommendations. In particular, using the well-known and researched electrocardiogram (ECG) MIT-BIH Arrhythmia database, the proposed convolutional neural network architecture shows similar performance when compared to the state of art classification algorithms using classical machine learning approaches.

Keywords—MIT-BIH, Deep Learning, Convolutional Neural Networks, ConvNet, ECG, heartbeat, classification, tensorflow, theano

I. INTRODUCTION

Heartbeat classification is a widely studied problem in pattern recognition. Classifying heartbeats from electrocardiogram (ECG) data into categories means finding irregularities and discrepancies in beats that are classified as normal and beats classified as non-normal. The AAMI (Association for the Advancement of Medical Instrumentation) [1], categorizes the non-life-threatening arrhythmias into 5 main classes namely: non-ectopic (N), supraventricular ectopic (S), ventricular ectopic (V), fusion (F), and unknown (Q). We used only 4 of these 5 classes: N, S, V and F. When classifying heartbeats, physicians look at specific features of the ECG signal, such as R-R interval, QRS complex, P-wave and T-wave[14].

![Fig.1 Features of ECG signal](image)

In this paper a simple convolutional neural network is proposed allowing the further development and inspection of a heartbeat classification tool for supporting cardiologists.

The paper is organized as follows: Section II presents a brief review of AAMI recommendations and classical ECG classification using machine learning with feature extraction techniques. Section III shows the dataset preparation processing. More specifically, the MIT-BIH database preprocessing is discussed and the creation of image files from electrocardiograms is shown. Section IV briefly shows how convolutional neural networks work. Section V shows the proposed architecture used for the presented simple ConvNet. The experimental results and comparison with state of art methodologies are addressed in Section VI. Section VII presents conclusions of work.

II. LITERATURE REVIEW

According to the AAMI standard [1], the Massachusetts Institute of Technology - Beth Israel Hospital (MIT-BIH) Arrhythmia Database is the one recommended for evaluating automatic arrhythmia classification systems. Many works make use of MIT-BIH Arrhythmia Database, however, few of them follow the AAMI class division scheme and a more realistic evaluation protocol (inter-patient paradigm). This paper takes into consideration only works that use the inter-patient paradigm as opposed to intra-patient paradigm. It has been shown that intra-patient paradigm has great accuracies with values over 98% (100% few works) [3] but it cannot be used in real world scenarios, where the goal is to classify beats of a specific person starting from a model trained on beats of different people.

In da S. Luz et al. [4] a full automatic system for arrhythmia classification from signals acquired by a ECG device is divided in four steps: ECG signal preprocessing; heartbeat segmentation; feature extraction; learning/classification. In each of the four steps, an action is taken and the final objective is the discrimination/identification of the type of heartbeat. As it is possible to note, the feature extraction step plays an important role in the classification accuracy of the heartbeat. De Chazal et al. [2] use ECG-Interval and morphological features and weighted linear discriminants as classification algorithm. Accuracy is 83%. Llamedo and Martinez [5] used wavelet, sequential forward floating search and VCG as features. They also used weighted linear discriminants with accuracy of 93%. De Lannoy et al. [6] used ECG-Intervals, weighted SVM, Hermite Basis Function Coefficients, morphological features and High Order Statistics Cummulants features. The classification algorithms used is weighted SVM. The accuracy is 83%. Escalona-Moran et al. [7] used the raw wave as the main feature and Reservoir Computing as classification algorithm. The accuracy is 98%. Park et al. [12]
used the Hermite Basis Function Coefficients and High Order Statistics Cummulants features. Accuracy is 85%.

III. ECG-PRE PROCESSING

In this paper, following the protocol suggested in the work of De Chazal et al. [2], the heartbeats from the MIT-BIH database are split into two sets so that the database becomes more coherent with reality. This split was conceived in order to adapt to practical situations. De Chazal et al. [2] proposed the inter-patient paradigm where the training and testing subsets were constructed from different ECG recordings so that inter-individual variation was taken into account and the classifier would exhibit better generalization ability. The first set is composed of all heart-beats of records: 101, 106, 108, 109, 112, 114, 115, 116, 118,119, 122, 124, 201, 203, 205, 207, 208, 209, 215, 220, 223 and 230, called Dataset 1 (DS1). While the second is composed of all heartbeats of records: 100, 103, 105, 11, 113, 117, 121, 123,200, 202, 210, 212, 213, 214, 219, 221, 222, 228, 231, 232, 233 and 234, called Dataset 2 (DS2). Literature reviewed make use of the same dataset split.

The tools provided by PhysioNet [8] for ECG sampling and to retrieve annotation of each heartbeat, are also used. In this work, only the 0-th channel (the first channel of the ECG data frame) of each record was extracted and the signal was filtered with wavelet decomposition using Daubechies 6 (db6) level 0 decomposition (Fig.2).

The R peaks of the ECG signal are identified by the annotation position extracted with PhysioNet tools. Using R peaks position is possible to segment the ECG in heartbeats of same duration. In this work 1000ms are chosen as the duration of a heartbeat centered at the R peak. For each beat a picture (a PNG) image is created by plotting the beat in black and white format creating 256x256 b/w png images (Fig.3).

It is worth noting that the normal class dominates the distribution of heartbeats among classes with 89.5% of total beats. To balance the dataset, a random padding technique is applied on the heartbeats of all other classes, different from the normal class, allowing to have the same number of heartbeats in all classes. The padding works by randomizing the amount (in percentage) of pixels on the x axis to be shifted left or right. Also the orientation of shifting is random. By default, we have selected a maximum randomization of 10%. All heartbeats images, along with their respective class, are lately treated as 2D array, divided by 255 for normalization purpose. All pre-processed heartbeats are saved inside the DS1 dataset. Same for the DS2 dataset which is used later for testing purposes, ensuring so to strictly following the AAMI inter-patient paradigm as discussed earlier.

IV. CONVOLUTIONAL NEURAL NETWORKS

Convolutional neural networks are a specialized kind of neural network for processing data that has a known, grid-like topology [9]. It is typically a combination of the following three layers: convolutional layer, polling layer, and fully connected layer. Those layers are typically stacked one to the other, with one or more convolution layers, followed by a max-pooling layer, followed by one or more fully-connected layers (multi-layer perceptron) as shown in Fig. 4. Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers [9]. In Convolutional networks, there are receptive fields that connect the input layer to a feature map. These receptive fields can be understood as overlapping windows that slide over the pixels of an input image to create a feature map. The stride lengths of the window sliding as well as the window size are additional hyperparameters of the model that are defined a priori. The process of creating the feature map is also called convolution [10]. The convolutional layer is, usually, followed by a pooling layer (or sub-sampling layer). In pooling, neighbors feature detectors are summarized to reduce the number of features for the next layer. Pooling can be understood as a simple method of feature extraction where the average or maximum value of a patch of neighboring features is taken and passed on to the next layer.
V. PROPOSED ARCHITECTURE

The proposed architecture, described in Figure 5, has been developed with Keras 2.0.9 for Python using both Tensorflow and Theano as backends. The architecture consists of 14 layers. It is the result of various trials and errors. The general idea was to start simple and add layers that reduces the complexity while compressing the size of the image pushed onto the next layers while extracting relevant features from image by using kernel moving windows to summarize features at different levels. The results of our tests is the following network:

The input layer works on 256x256 pixels images that are pushed into the next layer. The second layer is a convolutional 2D layer with 8 output channels, the kernel size is a 3x3 moving window followed by the strides (also known as subsamples) in both the x and y directions of 1 pixel. The activation function is the rectified linear unit (ReLU Fig. 6).

The third layer is a 2D max pooling layer, having a pool size of 4 pixels in both x and y directions and strides of 2 pixels in both x and y directions. The fourth layer is a convolutional 2D layer with 8 output channels, the kernel size is a 3x3 moving window followed by the strides in both the x and y directions on 1 pixel. The activation function is the rectified linear unit (ReLU in Fig. 6). Identical to the fourth layer is the fifth layer. The sixth layer is a 2D max pooling layer, having a pool size of 2 pixels in both x and y directions and strides of 2 pixels in both x and y directions.

The seventh, the eighth and the ninth layers are convolutional 2D layers that have always 8 output channels, the kernel size is a 3x3 moving window followed by the strides (also known as subsamples) in both the x and y directions of 1 pixel. The activation function is the rectified linear unit (ReLU in Fig. 5).

The tenth layer, the eleventh layer and the twelfth layers are equal to the ninth layer but the output channels are respectively 16, 24 and 24.

The thirteenth layer is a 2D max pooling layer with pool size of 2 pixels in both x and y directions and strides of 2 pixels in both x and y directions. Follows a flatten layer used to flatten the output from the previous layers. This flatten layer is composed by a dense layer (Multi-Layer Perceptron) with 4 neurons (the number of classes) and activation function softmax useful for our classification task (Fig. 7). The softmax function squashes the outputs of each unit between 0 and 1 and divides each output such that the total sum of the outputs is equal to 1. The output of the softmax function (Fig. 7) is used to determine the class with highest confidence value.

VI. EXPERIMENTAL RESULTS

The model was trained with a categorical cross entropy loss function whereas adadelta [11] was used as optimizer with Keras default settings. The training set DS1 is split in two
separate sets, namely the train and validation set, in each independent run, the validation set is 20% of the whole training set and not considered when training. The train process used a batch size of 32 images and, after plotting the accuracy on both training set and validation set, it was possible to determine that 25 epochs are enough for providing good results and good generalization power even if it still shows a slight overfitting gap as it’s possible to note from the learning curve in Fig.8a and Fig.8b where the trend of accuracy after 25 epochs is stable but the validation loss (Fig.8b) keeps increasing thus overfitting.

For the test, carried out on the DS2 dataset, four basic measures were considered: true positive (TP), false negative (FN), false positive (FP), and true negative (TN). From these measures, according to AAMI recommends [1], other statistical parameters have been calculated:

1. Sensitivity \( (Se) = \frac{TP}{TP+FN} \), that indicates the ability of a test to identify positive cases;
2. False Negative Rate = 1 - Sensitivity
3. Specificity (Sp) = \( \frac{TN}{FP+TN} \), that indicates the ability of a test to identify negative cases;
4. False Positive Rate = 1 - Specificity

The experimental results are the following:
Sensitivity=93.74%, False Negative Rate=6.25%, Specificity=75.88%, False Positive Rate=24.11%, Accuracy=90.73%.

The weighted normalized (weighted because of the unbalanced nature of the test set and normalized with respect to the number of elements for each class) confusion matrix in Fig.9 shows that Normal class has 97% of accuracy, the Super Ventricular Ectopic class has 76% of accuracy, the Ventricular Ectopic class has 81% of accuracy and Fused has only 46% of accuracy. A deeper analysis on Fused class description, showed that Fusion class is (as its name suggests) the fusion of Normal Beats and Ventricular Ectopic beats. Also in this case the confusion matrix in Fig.9 reveals that the majority of misclassified beats in Fused class were classified as Ventricular Ectopic showing the closeness of these two classes.

Despite compare results with literature reviewed is often difficult, a careful selection of research papers that used the same dataset and the same dataset split was performed, thus results are comparable with the state-of-the-art literature. In particular, accuracy is superior compared to De Chazal et al. [2], Park et al. [12] and De Lannoy et al. [6], whereas it is almost equivalent to Llamedo et al. [5]. Conversely, accuracy is not as good as Escalona-Moran et al. [7].

The training time took 18 minutes only on an Intel Xeon E3 quad core processor at 2.5Ghz with hyper threading, 32GB of DDR4 RAM and GPU accelerated with Nvidia Quadro M1000M with 2GB of Video RAM and 512 cuda cores. The DS1 dataset is about 48.3GB (balanced) on disk and the DS2 dataset is 33 GB (unbalanced) on disk. The amount of RAM needed is not only the sum of those two big dataset but also the amount of RAM to store data structures and the ConvNet.

VII. CONCLUSIONS

This work presented a new ConvNet architecture for effective heartbeats classification. The solution, that works on 2D black and white representation of the electrocardiogram’s heartbeat, is simpler with respect to other techniques for heartbeats classification in the literature and shows very good performances. It is worth noting that, conversely to previous approaches in the literature, the proposed architecture does not need the feature engineering phase.

In the next future a more complex ConvNet could be used to classify not only arrhythmias but also other critical conditions.
like sudden cardiac death (SCD), stroke prediction and atrial fibrillation classification.

REFERENCES


Fractal Modeling of Big Data Networks

Mahdi Barat Zadeh Joveini
Young Researchers and Elites club, Science and Research Branch, Islamic Azad University, Tehran, Iran.
m.joveini@outlook.com

Javad Sadri
Dept. of Computer Science & Software Engineering, Faculty of Engineering and Computer Science, Concordia University, Montreal, Quebec, Canada, H3G 1M8.
j_sadri@encs.concordia.ca

Hoda Alavi Khoushhal
Educational School of Joghatai city. Joghatai, Iran.
hoda.alavi.kh@gmail.com

Abstract—In recent years, the key role of complex and big data networks in different areas of sciences have conducted many studies to analyze their structures and functions and to understand these systems deeply. Since the classical network models used to indicate complex networks do not capture their main features such as centrality, clustering, degree distribution, etc., several attempts have been made to introduce new network models with desired features. In this paper, a novel method is presented based on fractal theory for modeling of big data networks. It is capable of modeling different complex networks. It is based on a mapping of the adjacency matrix into an nD space. In fact, it is a self-similarity network evolving the model based on the similarity on an adjacency matrix. We then show that the model has the expected properties and it can actually be seen as a general model for complex networks. Therefore, this model can be used to classify and to cluster data and to predict events in complex networks.

Keywords—Complex networks, fractal theory, data modeling, big data analysis

I. INTRODUCTION

Network theory, as a part of graph theory, is the study of complex interacting systems to represent either symmetric or asymmetric relations between distinct objects. Network theory has applications in social networks, computer science, and biology. To model, visualize and analyze complex systems, the network theory provides a set of techniques. [1], [2], [3]. A complex system [4] is composed of many constituents which may interact with each other. Because of these involved interactions between their parts or between a given system and its environment including dependencies and relationships, cause these systems to be “complex”, it is difficult to model the behavior of these systems. To overcome this complexity, it is useful to represent the system as a network. The structure of a complex network shows the nodes corresponding to the units and the edges to the connections between the units of the original system. By a network representation can seize the intricate structure behind such complex systems [5], [6]. In order to analyze the representation, however it is faced with spatial and temporal complexity, can obtain the properties of the complex network. First step in analyzing is data modeling and visualizing which analyze the data and its relationships to other data [1], [6]. For this reason, to understand interactions within complex networks, real-world networks can be simulated [7]. Accordingly, advanced visualization techniques appeared, and some parameters have been introduced recently.

In the recent years, introducing graphs with suitable properties has attracted great interest. The different approaches have been presented to model graphs or networks such as Bipartite Graph Generator [8], Sequential Graph Generator [9], Markov Graph Generator [10], The Watts and Strogatz model [11] and Internet specific models [12]. In mathematics, the Kronecker graph approach introduced by Leskovec et al. can reveal several properties of real networks simultaneously. This approach is focused around hierarchic adjacency matrices; so, in this case the self-similar structure is achieved by Kronecker multiplication as follows

\[ C = A \otimes B = \begin{pmatrix} a_{1,1}B & a_{1,2}B & \cdots & a_{1,m}B \\ a_{2,1}B & a_{2,2}B & \cdots & a_{2,m}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}B & a_{n,2}B & \cdots & a_{n,m}B \end{pmatrix} \] (1)

Here, \( \otimes \) denotes the Kronecker product, an operation on two matrices of arbitrary size resulting in a block matrix, \( C \) is dimensions \((n \times n') \times (m \times m')\), \( A = [a_{ij}] \) and \( B \) is sizes \( n \times m \)

Probabilistic graphical models such as Bayesian networks [14], by combining principles from graph theory, probability theory, computer science, and statistics, are used to represent knowledge about an uncertain domain [15], [16]. A Bayesian model has been developed to group similar customer choices by Durante et al. to model customer behavior via a cluster-dependent mixture on undirected networks [17]. Shaohua Tao et al. propose the information transfer model to analyze the self-similarity of complex networks [18]. Another model by Shaohua Tao et al. based on similarity between the nodes is the self-similarity network evolving model. Their paper consists of two kinds of node removing method; first, the random fault and second, deliberate attacks. By this method the node is removed completely random and then the node having the major link with the other nodes is removed [15]. Shifting data into a graph/network pattern by visualization and analysis of data as networks allows exploration of relationships between entities in different fields in the graph and the network as a whole [6], [17].

Self-similarity which is called fractal structure, means to split an object into tiny parts so that each one is a copy of the whole (as Benoit Mandelbrot set, each part is a reduced-size piece of the whole) [19]. The fractal feature of complex data networks which have been rarely studied yet, is important concept to describe complex systems in nature and be relevant in network theory [20]. Developing capabilities for analyzing, designing and controlling complex networks require to address some major elements: to understand network structure, to understand network dynamics, to model predictively and to simulate dynamic networks. It is the final aim of a research program on mathematical modeling to simulate and analyze complex networks [21]. In this paper, we propose a new model of big
data networks based on fractal theory to analyze their structures with similar properties. It presents a complex network in an nDimensional \((nD)\) space, where vertices place in different zones of the fractal model. One of our goals is to improve this modeling for motif discovery and clustering algorithms and any like these. Here, we also introduce new properties and applications to evaluate and analyze complex networks. Experimental results show this model performs more efficiently than similar ones in terms of spatial and temporal complexity.

The rest of this paper is organized as follows: Section 2, the details of proposed method, Section 3, experimental results and discussion and Section 4, conclusions and future works will be expressed.

II. PROPOSED MODEL

Gergely Palla et al. [6], Árpád Horváth et al. [22] and Mahdi Barat Zadeh Joveini et al. [23] have described the generation process of multifractal network in their papers in details. In this work, we slightly modified their methods to convert a complex network into a multifractal network.

In this research, the multifractal network at level \(L = 1\) is divided into four equal zone squares. And their values are calculated based on the number of \(M_{uv}\)s in the adjacency matrix. Here, \(M_{uv}\) represents an edge from vertex \(u\) to vertex \(v\) in matrix \(M\). At the levels, the probability of each zone is calculated as follows

\[
p_{ij}^{L=1} = \frac{N_{zone}}{N_{total}}; \quad \text{(in each level } L: i, j \in [1, 2^L]) [23] (2)
\]

\(N_{zone}\) denotes the number of \(M_{uv}\)s in a zone and \(N_{total}\) is the total number of \(M_{uv}\)s in the adjacency matrix. At each level, the probability of each zone is different from others. The multifractal network probabilities may be calculated at several levels. The number of levels depends on the complex network \((L: \{1, 2, ..., \log(\text{Node})\})\). The multifractal network size and the adjacency matrix size are the same. at the next levels, this process is repeated for each zone; and each zone divided into four equal zones (two rows and two columns, thus \(m\) is equal 2). Next, the probability scale \((p_{ij}^{\log(\text{Node})})\) is obtained by recursively multiplying each zone with the \(L\) time generated scale (which is equivalent to take the \(L\)th product of the generated scale). In other word, the probability at the last level is achieved by recursively multiplying each zone value with the probabilities of corresponding zones of the same zone at prior levels. The above procedure is the standard process of generating a multifractal, resulting in all zones, each zone associated with a probability \(p_{ij}\) equivalent to a product of \(L\) factors from the original generated \(p_{ij}\) given as

\[
p_{ij}^L = \prod_{q=1}^{L} p_{ij/q}
\]

where

\[
i_q = \left\lfloor \left( (i-1) \mod m^{(L-1)} \right) \mod m^{(L-2)} \right. \mod m^{(L-1)} \left. \mod m^{(L-q+1)} \right\rfloor + 1 [23] (4)
\]

The indices \(i_q\) can be calculated by this formula too. The details of the multifractal network are shown in Table 1. As this table shows, the first level consists of four zones, the second level consist of sixteen zones, the third level consists of sixty-four zones and so on. Each level, zones with the most number of \(M_{uv}\) have the highest probability value and may participate in many connections with other zones. In fact, each \(M_{uv}\) same at all levels is part of the probability of multifractal network at the last level. Furthermore, for undirected networks, the multifractal network is symmetric and for directed networks, it is non-symmetric (see Fig. 2). For more details, see [6], [22], [23], Table 2 and Fig. 1.

---

Fig. 1. The construction of different levels in a multifractal network
\[ p = -\prod_{m} \log m = \text{he probability values} \]

The proposed model is created four hypothetical cluster zones not only by using the promising vertices in the zones with the highest probability values. In fact, the proposed model ignores some zones by detecting substructures of each network. These zones are not required to be explored, as a result, this improves the runtime of modeling. In other words, this method clusters zones not only by using the promising vertices in the zones with the highest probability values in the multifractal network but also by ignoring the rest of structures on it. As a result, this method can be used to cluster data and to predict events in the multifractal network based on its probabilities. Based on literature review, there is no background about it and similar algorithms are not capable to do so. This method can be used to discover network motifs [17] in complex networks, especially in biological science.

In this current paper, we only focus on representation of complex network in fractal theory. Because of the fractal feature of the complex networks, we used fractal theory to model them. Fractal theory has many advantages such as modeling, simulation, structure analysis and evaluation of structural complexity. As mentioned in this paper, a complex network with a lot of nodes and edges converts a multifractal network with probability value. It is known that the process of probability matrix (the multifractal network) is faster and easier than the process of adjacency matrix with many nodes and edges. Based on fractal theory [27], we modeled the structure of big data networks by our proposed method to decrease the runtime and memory usage. In our method, low memory consumption and low runtime for complex networks, especially very large complex networks, are the most important features. The proposed method presents a new fractal model of big data networks with various statistical properties. In the proposed method, a graph or network is based on a multifractal model (a self-similarity model) so that there are no edges and nodes of graphs or networks because they are converted to values in multifractal network. Then, the probability values of multifractal network are

### III. Experimental Results and Discussion

In this Subsection, various big data networks [24, 25] have been tested by the proposed method and some of the experimental results have been described (e.g. see Fig. 2). In the multifractal network, the zones with the highest probability values at levels, contain the most promising connections of the network nodes. It means that much data is focused on that zone. In contrast, the zones with the lowest probability values are ignored (see Fig. 2). In these zones, there are \( M_{uv} \)S of the complex network that do not contribute to create a subgraph or a motif. Also, the zones with close probability can create a cluster. In Fig. 2, the proposed method is created four hypothetical clusters that each of them consists of the zones with different probabilities. The weight of each cluster shows the probability of existing information therein.

![Fig. 2. The probability matrix of USpowergrid [24] Multifractal Network at Level \( L = 3 \)](image)

After ignoring the unpromising \( M_{uv} \)S of the complex network, the remaining zones have produced a subnet of the original network where the structure of obtained subnet and the original network is almost the same. If QuateXelero [26] is applied on this subnet, the obtained results of this subnet and original network will almost be the same. In fact, the proposed model ignores some zones by detecting substructures of each network. These zones are not required to be explored, as a result, this improves the runtime of modeling. In other words, this method clusters zones not only by using the promising vertices in the zones with the highest probability values in the multifractal network but also by ignoring the rest of structures on it. As a result, this method can be used to cluster data and to predict events in the multifractal network based on its probabilities. Based on literature review, there is no background about it and similar algorithms are not capable to do so. This method can be used to discover network motifs [17] in complex networks, especially in biological science.

In this current paper, we only focus on representation of complex network in fractal theory. Because of the fractal feature of the complex networks, we used fractal theory to model them. Fractal theory has many advantages such as modeling, simulation, structure analysis and evaluation of structural complexity. As mentioned in this paper, a complex network with a lot of nodes and edges converts a multifractal network with probability value. It is known that the process of probability matrix (the multifractal network) is faster and easier than the process of adjacency matrix with many nodes and edges. Based on fractal theory [27], we modeled the structure of big data networks by our proposed method to decrease the runtime and memory usage. In our method, low memory consumption and low runtime for complex networks, especially very large complex networks, are the most important features. The proposed method presents a new fractal model of big data networks with various statistical properties. In the proposed method, a graph or network is based on a multifractal model (a self-similarity model) so that there are no edges and nodes of graphs or networks because they are converted to values in multifractal network. Then, the probability values of multifractal network are

### Table 1. Multifractal Network Details

<table>
<thead>
<tr>
<th>( L )</th>
<th>The number of rows or columns</th>
<th>The number of zones</th>
<th>Probability of each zone</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>( p_{ij} )</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>16</td>
<td>( p_{ij}^2 )</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>64</td>
<td>( p_{ij}^3 )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>( n = \log \text{[Node]} )</td>
<td>( m^n )</td>
<td>( m^n + m^n )</td>
<td>( p_{ij}^0 ) - ( p_{ij}^1 )</td>
</tr>
</tbody>
</table>

### Table 2. Pseudo code of Multifractal Network Generation Process

```plaintext
procedure MFNG (\( G_{n,m} \): complex network)
begin
    1. initialize the multifractal network (level L = 1);
    2. calculate the multifractal network values (levels \( L = 2 \) to \( L = \log(\text{Node}) \));
        a. for all \( i \) & \( j \) to \( 2^L \)
            \[ p_{ij}^L = \prod_{s=1}^{L} p_{i,s} \cdot p_{j,s} \];
        b. end for
    3. return \( p \); \(/p = \text{Multifractal network}\)
end
```
used to analyze, model and visualize big data networks. The proposed method is useful to model complex systems with special features. It is known spatial and temporal complexity are indicated when a scientist analyzes and simulates graph or network structures. One of the important features of this method is that the network structure becomes more accurate by increasing the number of levels of the multifractal network. All these capabilities have been obtained based on fractal theory. This might aid to interpret the characteristics of new networks and to understand well their organization.

IV. CONCLUSION AND FUTURE WORKS

To model graph and network structure, many methods are presented. In this paper, we have proposed a new method to model and to make big data networks suitable for wide different usages on undirected and directed networks. A significant step towards the realistic modeling of complex data networks is to extract the properties of the main network using its \( nD \) structure which can be shown in the obtained network very well. So it is important to understand the details of its structure which can be obtained by specifying statistical characteristics in big data networks. Our proposed method is not only able to model different types of networks, but also to predict the vertex position of a complex network to provide a useful tool for their clustering and classifying. There are still a lot of problems to extend this model. In the future, we would like to improve and extend our model in order to discover and locate network motifs in complex networks using fractal theory.

REFERENCES

Application of Pattern Recognition in Mineral Segmentation and Identification

Izadi Hossein*
Department of Petroleum Exploration Engineering
University of Tehran
Tehran, Iran
hossein.izadi@ut.ac.ir

Sadri Javad*
Department of Computer Science and Software Engineering
Concordia University
Montreal, Canada
j_sadri@encs.concordia.ca

Abstract—Pattern Recognition and Machine Learning techniques provide great tools for solving many problems in geosciences including mineralogy, petroleum engineering, and rock mechanics. In this paper, we aim to focus on mineralogy in which we use thin sections in plane and cross polarized light mounted from rock samples for such purposes as mineral segmentation and identification. Minerals inside thin sections are generally divided into altered and not-altered, where altered minerals are frequent in industrial applications and have color changing and do not have close and evident boundary/edges. Intelligent mineral segmentation in thin sections containing altered minerals is the main contribution of this paper in which did not widely considered yet. We use an incremental learning method to segment minerals and then the segmented minerals are identified by artificial neural network in a cascade approach. Features used in this study include color and texture parameters in RGB and HSI color spaces for both plane and cross polarized light images. The output of the proposed method can be used in several important applications in mineralogy.

Keywords—Intelligent mineral segmentation and identification, geological thin section, image processing and analysis, incremental learning, cascade approach.

I. INTRODUCTION

Machine learning is widely used in geosciences for feature/parameters prediction [1], recognition [2], classification [3], and analysis [4]. Mineralogy, which is a branch of geosciences, studies different features of minerals inside rock samples such as grain size [5], mineral percentage [3], and name of mineral [2]. Transparent optical mineralogy is based on microscopes with plane and cross polarized light and thin sections mounted from rock samples [6] where aims to segment and identify minerals inside thin sections (Fig. 1). Minerals inside thin sections are altered or non-altered, where altered ones are frequent in industrial applications which have color changing and do not have close and evident boundary/edge (Fig. 1). Working with optical microscopes is burdened with errors due to the light reflected from light source to human expert’s eyes and makes it tired. Besides, in such cases that many thin sections need to be studied, it is time consuming [2]. Consequently, developing an algorithm for intelligent mineral segmentation and identification based on images captured from thin sections makes great contribution in modern computational mineralogy.

Several algorithms were developed in the literature for intelligent mineral segmentation and identification; however, most of them studied synthetic thin sections or those contained non-altered minerals. We initially present mineral segmentation literature, and then present for identification. An algorithm was developed based on image processing techniques [7]. Gradient filtering and Lazy Grain Boundary (LGB) method was used for grain boundary detection in thin sections [8]. In another study, an alternative algorithm for boundary detection of grains using maximum likelihood classifier was suggested [9]. Another method based on watershed algorithm and manual correction was proposed [10]. In another study, a manual computer program (TSecSoft) which segmented minerals by using minimum birefringence color intensity was proposed [11]. A region competition framework, which also included region merging method was proposed [12]. In another study conducted by us, a clustering algorithm was used to segment minerals [3]. In identification literature, we have developed an algorithm based on Artificial Neural Network (ANN) and clustering mineral identification based on color features with the accuracy of 95.4% [13]. In another study conducted by us, texture features were successfully used for mineral identification [14]. A classification algorithm was developed for identifying macroscopic scale minerals in thin sections in which 90.00% overall accuracy using a maximum likelihood classifier [15]. In another study, 10 minerals were identified using an ANN with the overall accuracy of 93.53% [16]. In the experiment carried...
In this paper, which is the fifth paper of our series of five papers in this field, we initially present our previous results and methods, and then present our contribution chained to previous results. Our contribution is developing an algorithm for intelligent mineral segmentation in both thin sections containing altered and non-altered minerals, with special attention to altered minerals. In the proposed method, thin section images are captured in plane and cross polarized light, and a 1×12 optical feature vector for each pixel is extracted. Afterwards, an incremental algorithm is used for segmenting minerals inside thin section. Note that anticipating the number of minerals inside thin section is impossible, and so using such clustering algorithms as k-means or KNN is not practically applicable in mineral segmentation. The main contribution of this study is dealing with non-altered minerals segmentation.

The rest of this paper is organized as follows. In section II, database and feature extraction for this study is presented. Section III includes our previous results which produced a research-based software titled as Intelligent Mineral Segmentation and Identification System (IMSIS). In Section IV, our new method for mineral segmentation is described. The experimental results are presented in Section V, and conclusion remarks are stated in Section VI.

II. DATABASE AND FEATURE EXTRACTION

Petrographic microscopes or polarizer microscopes are commonly used for manual minerals identification in thin sections [17]. They also work with two kinds of lights including plane polarized and cross polarized lights. In order to study rock samples, they are turned into thin sections. A thin section is a 30µm thick slice of rocks mounted on a glass microscope slide. To establish a database for this study, the images were captured in both plane and cross polarized lights. In order to give the same values to the features, RGB and HSI color components are linearly mapped for assigning to the clustering algorithm, as it is shown in Fig. 3. In order to give the same values to the features, RGB and HSI color components are linearly mapped from 0 and 1.

III. IMSIS, INTELLIGENT MINERAL SEGMENTATION AND IDENTIFICATION SYSTEM

In this Section, our previous works for intelligent mineral segmentation and identification are presented [2, 3, 13, 14], and then our new contribution is presented. An intelligent system for mineral segmentation and identification has been developed in our previous works [2, 3, 13, 14] (Fig. 4). As shown in Fig. 4, two images of a thin section in plane and cross polarized light were fed to the system. The phase#1 was intelligent mineral segmentation which was fed by thin section images in plane and cross polarized light, and after normalizing and image resizing, color parameters were extracted, and then an incremental clustering algorithm was used for segmenting minerals and producing several mineral clusters [3, 13]. Afterwards, those mineral clusters containing index minerals were chosen and then, those mineral clusters with a similarity of greater than 85% were combined together. Finally, several mineral clusters were produced as the segmented minerals [13, 3]. The phase#2 was the intelligent mineral identification [2, 14]. To deal with intelligent mineral identification, a cascade classification approach was used. The proposed algorithm was based on color parameters and textural features, and artificial neural networks were also used for identifying minerals. As shown in Fig. 4, the produced mineral clusters, which were the output of the phase#1, were passed to the proposed method for mineral identification. Initially, in feature extraction step, a 1×12 color parameter vector for each pixel in the mineral clusters was extracted including RGB and HSI components in both plane and cross polarized light. The first level of the cascade was color-based identification. In the second level of the cascade, two 2×120 texture feature vectors in both plane and cross polarized light images are extracted from each...
mineral cluster. The first $2 \times 120$ feature vector refers to the plane polarized light, and the second one refers to the cross polarized light. Moreover, the first row of each $2 \times 120$ texture feature vector includes five texture features in RGB color space containing correlation ($C$), energy ($E$), homogeneity ($H$), maximum probability ($P$) and entropy ($A$) in eight directions including $0^\circ$, $45^\circ$, $90^\circ$, $135^\circ$, $180^\circ$, $225^\circ$, $270^\circ$ and $315^\circ$ based on three gray scales of 64, 128 and 256 from the original images (Fig. 4). The second row is similar to the first one except that it refers to the HSI color space. The final output of the IMSIS was the number of minerals, name of minerals, percentage for each mineral, and name of probable target rock that the thin section mounted from (Fig. 5). However, some pixels were missed in phase#1 and so were not recognized (Fig. 5). In this paper, we solve this limitation. In the next Section, we present our proposed method.
IV. THE PROPOSED METHOD

Since anticipating the exact number of minerals in thin sections is very difficult, conventional clustering algorithms like k-means [22] or ANNs [23] are not appropriate for minerals clustering or segmentation of thin sections. Therefore, it is necessary to use an algorithm which does not need the number of clusters (minerals) as a priori parameter. Moreover, the algorithm must follow an incremental nature in order to take new minerals and assign them into formerly
created clusters or to create new mineral clusters. Due to the ART (Adaptive Resonance Theory) algorithms having online incremental learning capabilities, the basic idea to design proposed clustering algorithm is adopted from the ART. This algorithm can run faster than k-means algorithm because it scans all the data points only one time during the clustering procedure. Accordingly, this approach is suitable for clustering of online incoming data [24].

The steps of the proposed clustering algorithm are presented as follow [24].

1. Adjust the minimum similarity threshold ($\delta$).
2. Cluster centers list $\phi = \emptyset$.
3. Read the next input pixel.
4. Find all similar clusters’ center to the inputted pixel in the cluster center list with a similarity greater than $\delta$ in the range of 0 and 1.
   - If found: assign the pattern to those clusters, and adjust the cluster center, and combine those clusters which have interconnected with each other (have at least one common pixel).
   - If not found: create a new cluster and insert the inputted pattern to cluster center list as a new cluster center.
5. Repeat steps in 3-4 for all the input patterns.

In order to produce the segmentation map, all pixels are assigned to the proposed method one by one, and based on a minimum similarity threshold ($\delta$), several clusters are produced. According to step#4, our algorithm tries to minimize the number of clusters, while trying to keep the minimum similarity threshold among the pixels in each cluster. Combining the produced cluster is the key point of our algorithm to deal with altered mineral segmentation like shown in Fig. 2. In the next Section, our experimental result for segmenting minerals exhibit in Fig. 2 are presented and compared with our previous method for intelligent mineral segmentation [3].

**V. Experimental Results**

The proposed algorithm classifies all pixels of the images into several clusters with respect to $\delta$. We do not suggest any value for $\delta$, and it could be determined by the human expert who uses our method. Here, for segmenting thin section shown in Fig. 2, we set $\delta=0.93$. The segmentation map and diagram for changing the number of clusters (mineral clusters) during clustering are shown in Fig. 6.

![Segmentation map](image)

**Fig. 6.** A thin section including Biotite and Orthoz as altered mineral and Quartz as non-altered mineral; and its corresponding segmentation map and diagram of changing the number of mineral clusters during clustering

Based on Fig. 5 which presents results by the algorithm proposed by our previous study [3], and Fig. 6 which exhibits the segmentation results by the proposed method, there is no missed pixels in Fig. 6, and also color changing existed in altered minerals are figured out dramatically by the proposed method.

**VI. Conclusion**

Intelligent mineral segmentation and identification in thin section still remains a challenging task in modern computational mineralogy. Intelligent mineral segmentation in thin sections contain altered minerals is more complex than those contain non-altered minerals. Intelligent method for mineral segmentation tries to make the hardware smarter for
studying thin sections images as close as possible to human expert. Besides, developing an intelligent method for mineral segmentation should be based on such natural complexity of minerals in thin section as color changing, open and non-evident boundary/edges. In two papers of our previous works, as the first attempt, we have developed an intelligent system for mineral segmentation in thin section with especial attention to thin sections contains altered minerals. In our previous method, some pixels were labeled as background pixels and missed from final results which this challenge has been solved in this paper. The main novelty of the proposed method is combining interconnected clusters during the clustering procedure. The experimental results indicated that the proposed method outperforms other segmentation methods in the literature for intelligent altered mineral segmentation in thin sections. The proposed method could be applied in applications which require a real time segmentation or efficient identification map such as mineralogy, and petrography.

REFERENCES


Sequential Transformation with Geometric Constraints for Matching Oblique Aerial Images

Woo-Hyuck Song, Hong-Gyu Jung and Seong-Whan Lee
Dept. of Brain and Cognitive Engineering
Korea University
Seoul, Korea
{whsong, hkjung00, sw.lee}@korea.ac.kr

Abstract—This paper presents a fast and accurate method for matching oblique aerial image pairs. In order to achieve accurate matching results, we must consider viewpoint differences. Existing methods that match aerial image pairs with viewpoint differences undergo heavy computation and have difficulty finding corresponding feature points when applied to aerial images. In this paper, we propose a homography matrix evaluation step based on geometric constraints to increase the accuracy of image matching results. In addition, we achieve faster matching through a sequential transform simulation that reduces computational complexity. Experimental results show that the proposed method improves aerial image matching in terms of computational efficiency while achieving successful results.

Keywords—SIFT descriptor, feature matching, aerial image, viewpoint difference, homography matrix

I. INTRODUCTION

Due to the recent advances in aerial imagery, high-resolution aerial images are easily accessed and acquired. Among the many applications that involve these high-resolution aerial images is aerial image matching, which finds correspondences between an image pair that show the same area and transforms the pair with a same coordinate system. Manual annotation of Ground Control Points (GCPs) in aerial images for matching is a classical approach which requires human inspections. Instead, feature-based matching automatically detects salient regions around feature points and descriptor vectors are computed for the points. Matching is performed with these vectors by choosing the closest pair as a correspondence. Among the various handcrafted descriptors, the Scale Invariant Feature Transform (SIFT) [1] is the most popular due to its superior performance [2].

The SIFT descriptor is based on the Difference of Gaussians (DoG) blob detector and orientation assignment from the histogram of gradients. This makes SIFT invariant to the scaling and rotation of images. However, SIFT descriptors are not fully affine invariant, and its matching performance degrades when significant viewpoint differences exist [2]. To overcome this problem, previous studies [3-6] have utilized SIFT descriptors and matched oblique image pairs. In particular, Affine-SIFT (ASIFT) [3] acts as a baseline approach in this paper.

ASIFT is a variant of SIFT designed to match images with different viewpoints. ASIFT simulates the input images to cover every transformation within the affine space, then uses the SIFT descriptor to compare and match the simulated images. This process is called transform simulation, as it imitates various viewpoints to find correspondences between images with viewpoint differences. However, ASIFT suffers from high computational complexity, as it does not take into account the relationship between the given images. In other words, ASIFT considers every possible cases of viewpoint differences.

There are three difficulties in the aerial image matching task. First, aerial image pairs are usually multi-temporal, meaning that there are non-linear differences between corresponding pixels in the two images. For example, shadows from clouds or local changes such as new buildings or roads may give one area a different visual appearance at different times. Changing imaging conditions over time can change the pixel information in an unexpected way. This can lead to wrong correspondences in the image matching process, which is not usually mentioned in previous studies [3-6] that match the same scenes with viewpoint differences only. Second, as aircraft with an on-board camera does not always fly parallel to the ground, oblique viewpoint can occur. Aircraft in flight can rotate around three principal axes, and such motions are represented as roll, pitch, and yaw. This rotation becomes a major challenge, especially in urban areas where buildings exhibit different appearances depending on the point of view an image is taken from. Third, as aerial images include many objects, feature-based matching incurs a heavy computational cost. A very low number of feature points have correspondences in the final matching result, and their distribution may be uneven [7]. This is due to the aforementioned multi-temporal and oblique characteristics of the aerial imagery, and many feature points are regarded as outliers after Random Sampling Consensus (RANSAC) [8]. Even though a very low number of feature points have correspondences in the final matching result, a large number of feature points are compared to find correspondences based on the extracted feature descriptors.

The goal of this paper is to overcome the challenges listed above, and to present a method that performs faster yet accurate oblique aerial image matching. The main idea was inspired by ASIFT, which proposed a novel approach of transform simulation. The contribution of this paper is three-fold. First, we propose a sequential transform simulation. Specifically, we stop the transform simulation if the maximal correspondence is found to reduce unnecessary computations. Second, we evaluate the
homography matrix with geometric constraints after RANSAC to prevent incorrect matching results. Lastly, we present the experiment results to prove the effectiveness of the proposed methods. A multi-temporal oblique aerial image dataset is used to show the superiority of the proposed method in terms of computation time compared to ASIFT.

The remainder of this paper is organized as follows. Section II describes the sequential transform simulation and the homography matrix evaluation, while section III presents the experimental results and its analysis. Finally, section IV draws conclusions.

II. METHODS

A. Concept of image matching

In this paper, two input images are termed sensed image \( I_s \) and reference image \( I_R \). We assume that a camera is placed far away from the ground, which is equivalent to high-altitude aerial images, and depth variation within a scene is negligible compared to camera-ground distance. Furthermore, we consider two images that show the same region, but have different points of view due to varying camera angles.

Projective transformation is a general model that defines the geometric relationship between two input images. Mathematically, projective transform is represented as a 3x3 homography matrix \( H \), which correlates the pixel coordinates of the sensed image \( x_s \) and the reference image \( x_R \).

\[
x_s = Hx_R
\]

The homography matrix \( H \) can be expressed with three parts: \( A \), \( p \), and \( \Delta \). \( A \) is the affine matrix, \( p \) is the projective vector, and \( \Delta \) is the translation vector.

\[
H = \begin{bmatrix} a & b & g \\ c & d & h \\ e & f & 1 \end{bmatrix} = \begin{bmatrix} A & \Delta \\ p^T & 1 \end{bmatrix}
\]  

The matrix \( H \) can be determined only if there are at least 4 matching ground point pairs \( \{x_g, x_s\} \). As we assume that the camera-ground distance is large enough, projective transformation can be approximated to affine transformation matrix \( A \). Translation vector \( \Delta \) can be ignored by assuming that the camera axis meets the image plane at a fixed point.

\[
\text{Fig 1. Parameters from the affine matrix } A \text{ decomposition.}
\]

B. Sequential transform simulation

The affine matrix \( A \) with positive determinant \( \lambda t \) can be decomposed into several transform parameters: where \( \lambda > 0 \) is the scale factor, angle \( \psi \in [0,2\pi] \) is the camera spin, \( t = 1/\cos \theta \) is the tilt, and angle \( \phi \in [0, \pi) \) is the camera longitude. Angle \( \theta \in [0, \pi/2) \) is the camera latitude. The geometric interpretation of the affine matrix decomposition is shown in Fig. 1.

\[
A = \lambda \begin{bmatrix} \cos \psi & -\sin \psi & t \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos \phi & -\sin \phi & t \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]

According to [3], the \( \phi \) range of \( [0, \pi) \) is enough since the \( \psi \) range covers cases where \( \phi \) exceeds \( \pi \). Therefore, we follow the \( \phi \) range defined in ASIFT. The SIFT descriptor is invariant to scaling and rotation within plane, so we focus on the two angles \((\theta, \phi)\) only. Transform simulation samples various image transformations with angles \( \theta \) and \( \phi \). Sampling frequency has a trade-off issue, as more computation is required for dense angle positions but ensures a more precise match. In [3], \( t \) and \( \phi \) are determined and validated experimentally as follows.

\[
t = 1/\cos \theta = [1, \sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}]
\]

\[
\phi = \left[0, \frac{b}{r}, \frac{2b}{r}, ..., \frac{mb}{r}\right] \text{ with } b = \frac{2\pi}{5} \text{ and } m = \frac{\pi t}{b}
\]

Sampling locations are illustrated as Fig. 2. As \( t \) is increased, the sampling of \( \phi \) becomes denser to cover more extreme viewpoint differences. To summarize, \( t \) is multiplied with fixed number of \( \sqrt{2} \) for each step, while \( \Delta \phi = 72^\circ/t \) which depends on current \( t \) value. There are total of 43 transform simulations according to range of (4) and (5).

The sequential transform simulation proposed in this paper improves ASIFT in terms of computational cost. ASIFT is speeded up by two-resolution procedure to compensate for increased computations. Initial matching is performed using downsampled images, then the final matching result with the original images is acquired from simulating only the top-5 transforms that yield the largest number of correspondences. However, this two-resolution approach is not feasible for aerial image matching. Downsampling an image loses its pixel information. Due to very low Number of Correct Matches (NCMs), or correspondences between aerial image pairs, it is even harder to find matches with low-resolution images. This will result in the failure of the low-resolution matching stage.

Instead, we focus on the fact that ASIFT does not care about the distribution of NCM with respect to the angles \((\theta, \phi)\). Even

\[
\text{Fig 2. Sampling locations of the angles } (\theta, \phi).
\]
if the viewpoint difference is small, ASIFT compares every simulated images. As the two angles change its value during transform simulation, there exist a peak where NCM becomes the maximum. The goal of the sequential transform simulation is to find this peak, which will represent the angles \( (\theta, \phi) \) that produces the most probable viewpoint difference between the sensed image \( I_s \) and the reference image \( I_R \). After this peak, transform simulations later on cause excessive viewpoint differences, and will not produce more NCM. Fig. 3 is a graph showing the relationship between the distribution of the correspondences and angles \( (\theta, \phi) \) as an example. NCM will increase until the peak, where the most correspondences are found. Then NCM will decrease as less correspondences are found with larger tilt angle \( \theta \). Therefore, there is no need to continue transform simulation over the full range of camera position angles \( (\theta, \phi) \), but only need to search until the NCM peak occurs.

Fig. 3 represents the angle range of the proposed method as a shaded area on \( \theta - \phi \) plane. In order to search for the global maximum of NCM, camera position angles are increased within the range of (4) and (5). This process is termed as sequential transform simulation, and described as Fig. 4. The process starts at \( \{ t = 1, \phi = 0^\circ \} \), or position (a) in Fig. 4 having no viewpoint difference. We simulate the reference image \( I_R \) only and compare with the sensed image \( I_s \) using SIFT descriptors to find correspondences. Then the tilt \( t \) is increased to \( \sqrt{2} \), and for each \( \phi \) value \( I_R \) is simulated to find a peak, starting from \( \{ t = \sqrt{2}, \phi = 0^\circ \} \) or position (b) in Fig. 4. Then the maximum NCM between the current \( t \) value and the previous \( t \) value are compared to find the global maximum. If the peak is observed for \( t = \sqrt{2} \), or (d) in Fig. 4 for example, then the transform simulation continues with \( t = 2 \) to verify. The NCM for \( t = \sqrt{2} \) must be larger than that of both \( t = 1 \) and \( t = 2 \) if the right homography is found. For the case of Fig. 4, NCM in (d) should be larger than that of (a) and (i). If this condition is met, which represents the global maximum, transform simulation is stopped assuming that the right homography is achieved without the need for searching any further. If not, the process continues with larger \( t \) values to find simulation that produces more correspondences. For example, NCM of position (i) will be compared with position (d) and position (p). Fig. 5 shows examples of simulated images. As the tilt angle is increased, the image becomes narrower, and less feature points are extracted as the SIFT descriptor is applied to each simulated image. Notice that rotating an image with longitude \( \phi \) is different from \( \psi \) rotation that preserves the image shape.

C. Homography matrix evaluation

As mentioned before, NCM is very low when matching aerial images with temporal differences and distortions from viewpoint differences. The transform simulation only relies on NCM to solve the homography matrix \( H \), and this could lead to wrong matching results. Matching failures during sequential transform simulation should be filtered out in order to have desirable results. Fig. 6 represents the appropriate and inappropriate cases of transform results, showing the shape of transformed images. We note that an appropriate homography matrix does not guarantee the successful final matching result. This procedure is designed to remove failure cases from inappropriate homography matrices, and only the remaining success cases are compared using NCM to produce the final matching result.

We aim to filter out failure cases by evaluating the homography matrix \( H \). Fig. 6(a) is a test rectangle \( ABCD \) having the same size as the reference image. By examining the shape of transformed quadrangle \( AB'C'D' \), we can filter out the
 inappropriate homography matrices. Such cases represent situations that can never happen with viewpoint differences. Fig. 6(d) shows a twisted quadrangle, and Fig. 6(e) represents a concave quadrangle. Fig. 6(f) is a “collinear” case, which we will explain later in detail. To summarize, failure cases represent situations where four vertexes of quadrangle $A'B'C'D'$ are not in a same plane.

The criteria for the evaluation of the homography matrix is identifying whether two diagonal lines $A'C'$ and $B'D'$ intersect. This is equivalent to checking if two vertexes, either $B', D'$ or $A', C'$ lie in same or different side of a diagonal line, $A'C'$ or $B'D'$ respectively. The relative location of two vertexes from a diagonal line will decide if the two diagonal lines intersect. We have proposed a novel algorithm based on these observations to evaluate the homography matrix with coordinates of three vertexes. Determining the relative location of point $p3$ from line segment $p1p2$ is equal to comparing the area of two triangles that are formed by three points of $\{p1, p2, p3\}$.

We define a function having coordinates of three points $\{p1, p2, p3\}$ as input. Term $p2.x$ represents the x-coordinate of point $p2$, and $p1.y$ represents the y-coordinate of point $p1$.

$$f(p1, p2, p3) = (p2.x - p1.x)(p3.y - p1.y) - (p3.x - p1.x)(p2.y - p1.y)$$  \(6\)

We only check the sign of function $f$, not its absolute value. If the signs of $f(A', C', B')$ and $f(A', C', D')$ are different, this means that the points $B'$ and $D'$ are on opposite side with respect to $A'C'$. In Fig. 6(d), $A'C'$ and $B'D'$ do not intersect, and we must also check if the points $A'$ and $C'$ are on the opposite side with respect to $B'D'$ by calculating $f(B', D', A')$ and $f(B', D', C')$. Further details are shown in Fig. 7, which checks if $A'C'$ and $B'D'$ intersect and its intersection is not one of four vertexes.

The first half of $6$ represents the area of gray triangle in Fig. 7 while the second half represents the area of black triangle. Only the difference between these two areas matters, so multiplications of $1/2$ in front of both first and second term are ignored. As in Fig. 7(a), if $p3$ is on the left of $p1p2$, the area of gray triangle is larger than the area of black triangle, and thus $6$ is positive. Similarly, Fig. 7(b) represents the case where $p3$ is on the right of $p1p2$. Since the area of gray triangle is smaller than the area of black triangle, $6$ is negative. In Fig. 7(c), $p3$ is on $p1p2$ meaning that the area of gray triangle is same as the area of black triangle. Therefore $6$ is zero and points $\{p1, p2, p3\}$ are said to be collinear, located on a same line segment. Fig. 7(d), (e), (f) shows that this algorithm can be extended to any distribution cases of points $\{p1, p2, p3\}$.

There are numerous ways to check if two line segments intersect, however this method has the advantage of recognizing collinear case without additional calculations. Collinear case means that an endpoint of one diagonal line barely touches the other diagonal line. An example of collinear case is shown in Fig. 6(f), where vertexes $A', D', C'$ are collinear. Transformation that changes a right angle of rectangle $ABCD$ into a flat angle cannot happen from viewpoint difference of camera. Therefore, a collinear case is classified as failure. Simpler algorithms that find an intersection between two line segments do not consider the location of intersection.
III. Experiment

A. Dataset and performance metrics

We conduct an experiment to evaluate the performance of the proposed image matching framework using the aerial image dataset. The aerial image dataset contains four pairs of images with viewpoint differences. We combined oblique images from [9] and nadir images from Google Earth [10] to create pairs that show the same area of Dortmund, Germany, but have viewpoint and temporal changes. Original images of [9] were scaled down, because the size was too large compared to [10]. Fig. 8 displays the aerial image dataset, and the sizes of images in the dataset are presented in Table I. Oblique images are sensed images, and nadir images are reference images in this experiment.

For evaluation, performance metrics are required for quantitative analysis. We use two metrics in this paper: NCM and Matching Precision (MP). NCM is the total number of correctly matched feature point pairs, or correspondences after RANSAC. MP is the ratio between NCM and Number of Matches (NM). NM includes both inlier points and outlier points, prior to RANSAC. MP measures the ratio of correspondences that contributes in the final matching result, thus denoting the quality of the matching result. We did not use Root Mean Squared Error (RMSE) for evaluation because RMSE needs manually selected GCP coordinates on images, which is a subjective measure. RMSE is defined as the square root of mean square error from the homography matrix.

<table>
<thead>
<tr>
<th>Image pair</th>
<th>Simulate</th>
<th>Proposed</th>
<th>ASIFT</th>
<th>Proposed</th>
<th>ASIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Both reference and sensed images</td>
<td>-</td>
<td>1120.3(x)</td>
<td>-</td>
<td>1769.1(x)</td>
<td></td>
</tr>
<tr>
<td>Reference image only</td>
<td>48.67</td>
<td>65.45(x)</td>
<td>49.26</td>
<td>83.50(x)</td>
<td></td>
</tr>
<tr>
<td>Both reference and sensed images</td>
<td>-</td>
<td>1306.0(x)</td>
<td>-</td>
<td>1680.0(x)</td>
<td></td>
</tr>
<tr>
<td>Reference image only</td>
<td>44.96</td>
<td>71.17(x)</td>
<td>51.84</td>
<td>71.33(x)</td>
<td></td>
</tr>
</tbody>
</table>

B. Test with aerial image dataset

The experiment matches the four aerial image pairs shown in Fig. 8, and compares the performance with ASIFT in terms of time. Table II shows the elapsed time using the proposed method compared to ASIFT, but only simulates the reference images instead of simulating both images as in the original paper. The notation (x) in Table II means matching failure, not able to present a correct matching result. In addition, we have replaced the original SIFT descriptor with RootSIFT [11] for much better results in terms of NCM. RootSIFT uses L1-normalization and square-root to improve the matching performance of the SIFT descriptor.

The proposed method, which combines the sequential transform simulation and the homography matrix evaluation, succeeded matching four aerial image pairs with much faster speed. Applying transform simulation only on the reference images decreased time, but still consumed more time than the proposed method and failed to output desirable results, shown in Table II. The previous methods have only focused on the maximum range of viewpoint differences, not considering the practical cases where the images can have either small or big viewpoint differences. The sequential transform simulation is a flexible method, able to compute the homography matrix faster with images having small viewpoint differences. Many geometric, illumination, and shape differences can be observed between the given aerial image pairs. Because of these apparent changes, NCM is significantly low. However, the homography matrix evaluation is able to filter out the matching failure cases, and matches oblique images accurately with less computation.

Fig. 9 presents the matching results of the proposed method with the aerial image dataset, transforming the reference images
to show the correspondence with the sensed images. For each pair, the sensed image, the reference image and the matching result is shown from (a) through (d). The proposed method is able to produce correct matching results significantly faster. From Table III, ASIFT have more NCM than the proposed method. However, ASIFT performance is significantly degraded in terms of MP. This means that although ASIFT finds more correspondences, this is due to the huge amount of calculation that compares much more feature points, and most of the matches are removed during RANSAC. To summarize, the proposed method achieves successful matching results efficiently with much less calculation.

<table>
<thead>
<tr>
<th>Image pair #1</th>
<th>Image pair #2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Simulate</strong></td>
<td><strong>Proposed</strong></td>
</tr>
<tr>
<td>NCM</td>
<td>24</td>
</tr>
<tr>
<td>MP</td>
<td>0.189</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Image pair #3</th>
<th>Image pair #4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Simulate</strong></td>
<td><strong>Proposed</strong></td>
</tr>
<tr>
<td>NCM</td>
<td>35</td>
</tr>
<tr>
<td>MP</td>
<td>0.057</td>
</tr>
</tbody>
</table>

C. Discussion

We compare the proposed method against various aspects of ASIFT, and summarize them as Table IV. The number of simulations in the proposed method is far fewer than that of ASIFT. The proposed method simulates the reference image only, but the reduced calculation is also due to stopping the sequential transform simulation early. Our proposed method checks for errors during the homography matrix evaluation step, but ASIFT only relies on the NCM. This can be misleading if the overall NCM is very low due to extreme viewpoint and temporal changes. This is why we have used MP as another evaluation criteria in this paper.

<table>
<thead>
<tr>
<th></th>
<th>ASIFT</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulate the reference image</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>Simulate the sensed image</td>
<td>O</td>
<td>X</td>
</tr>
<tr>
<td>Number of simulations</td>
<td>Many</td>
<td>Few</td>
</tr>
<tr>
<td>Computational cost</td>
<td>Very high</td>
<td>Low</td>
</tr>
<tr>
<td>Checking for error</td>
<td>X</td>
<td>O</td>
</tr>
</tbody>
</table>

Finally, we note the matching failure case using the proposed method. The selection of the sensed image and the reference image is important since the proposed method simulates the reference image only. Matching failure may happen if the order of the input images is switched. In such cases, we must perform the whole simulation process twice, by testing each of the two input images as a reference image. In addition, severe non-linear illumination differences in input aerial images from using different sensors can degrade the performance of the proposed method and can result in too few correspondences between the input images. This is due to the limited invariance of SIFT descriptors against severe illumination changes.

IV. Conclusion

In this paper, we proposed an efficient method for matching oblique aerial images, which can be divided into the sequential transform simulation and the homography matrix evaluation. First, the sequential transform simulation reduces calculation by stopping the simulation loop if the NCM peak is found. This is more effective for aerial images having many feature points, as feature computation and comparison are the main factors of computational complexity in image matching. Second, the homography matrix evaluation algorithm based on geometric constraints filters out various failure cases that may occur during aerial image matching process. The experimental results prove that the sequential transform simulation and the homography matrix evaluation are effective. The proposed method show a large gain in speed while achieving successful match results using oblique aerial images.

REFERENCES

Stochastic Modeling of Camera Errors for Stereo Image Processing

1st Florian Particke  
Information Technologies  
Friedrich-Alexander-Universität  
Germany  
florian.particke@fau.de

2nd Christian Hofmann  
Information Technologies  
Friedrich-Alexander-Universität  
Germany  
chris.hofmann@fau.de

3rd Markus Hiller  
Information Technologies  
Friedrich-Alexander-Universität  
Germany  
markus.hiller@fau.de

4th Lucila Patiño-Studencki  
Information Technologies  
Friedrich-Alexander-Universität  
Germany  
lucila.patino@fau.de

5th Jörn Thielecke  
Information Technologies  
Friedrich-Alexander-Universität  
Germany  
joern.thielecke@fau.de

Abstract—Stereo vision algorithms with dense outputs are of great importance in the field of computer vision, virtual reality, environment reconstruction and robot navigation. In the field of robot navigation, mobile robots and fully automated vehicles have to be tracked in an environment they share with pedestrians. For this purpose, an increasing use of stereo cameras is expected in the future due to the significant improvements in the quality of distance measurements that can be achieved compared to mono cameras.

The stochastic properties of such a sensor are crucial for the decision of the optimal tracking algorithm, e.g. Kalman Filter for a Gaussian probability density function. However, in the current research, mostly only the accuracy of the algorithm is analyzed. In this paper, it is tried to close the gap. The stochastic properties of a stereo image processing chain are analyzed. In experimental results, it is shown that the reprojection error is Gaussian. This contribution is very promising, as these results indicate that a Kalman Filter is the proper choice as a tracking algorithm in the image plane. Additionally, first results regarding the distance measurements achieved by the stereo image processing chain are presented. It is shown that the error from triangulation int this setup is Gaussian with an additional systematic error.

Index Terms—Camera, Calibration, Surveillance, Stereo Vision, Dense Output

I. INTRODUCTION

With the rapidity of the technological developments in the area of autonomous driving, it will become more and more important to interact with humans in urban environments. To avoid accidents, it is crucial to track and predict humans in a precise way. One possible user scenario is valet parking in car parking areas, where fully automated vehicles have to operate with manually driving cars and humans [1]. Additional available information sources are environmental cameras, which are often mounted in surveillance areas and are able to enhance the field of view of the fully automated vehicles. Hence, the detection and tracking of pedestrians in surveillance areas by environmental cameras is part of current research. Examples are [2] and [3]. In the past, extensive research was performed to compare the accuracy of different algorithms for stereo vision [4], [5]. In most cases, the error distribution is for simplicity assumed to be Gaussian without proof. Ignoring the actual true error distribution may show major drawbacks, when an unsuited tracking algorithm for the present error distribution is used. One rare paper is [6], which shows first experimental results regarding the analysis of the error distribution. The following paper will considerably expand the experimental results of the previous work regarding the stochastic properties of the error distribution.

As stereo cameras significantly improve distance measurements in comparison to mono cameras, which always need some reference point or area [7], the proposed analysis is performed for stereo cameras. The distribution of the error in the image plane and in the world coordinate system is of crucial interest, because algorithms like Kalman Filter are only optimal in the case of a specific error distribution, e.g. in the case of Kalman Filter for a Gaussian probability density function [8].

In this paper, stochastic properties are analyzed, especially the error distributions in a proposed stereo image processing chain are examined. This paper focuses on the error distribution of the reprojection error and shows first results regarding the distance measurements, which are based on triangulation and depth values from a disparity map. For the experiments, the ZED stereo camera [9] of Stereolabs is used, which is mounted at the wall. The calibration is performed by OpenCV (version 3.1.0) [10].

The paper is organized as follows. In Section II, the stereo image processing chain and the experimental setup are described and the focus of the experimental analysis is set into context. The experimental results are discussed in Section III. In Subsection III-A, the error distribution of the reprojection error is analyzed in detail and first results regarding the
distance measurements are presented in Subsection III-B. In Section IV, the results are summarized and an outlook of future research is given.

II. STEREO IMAGE PROCESSING CHAIN

In this Section, the stereo image processing chain and the experimental setup are presented. The stereo image processing chain is discussed in detail in Subsection II-A, while the experimental setup is pointed out in Subsection II-B.

A. Overview of Processing Chain

Fig. 1. Overview of the stereo image processing chain.

The stereo image processing chain, which is used for the analysis of the stochastic properties, is depicted in Fig. 1. After the calibration of the left and right image, which has been performed using a checkerboard, the reprojection error is calculated. The reprojection error is the focus of this paper and is examined in Section III-A. After the analysis of the reprojection error, the stereo calibration is performed. The calibrated stereo image is rectified and distortion is removed. For the analysis of the distance measurement, two different approaches are examined in detail: triangulation and distance measurement based on a disparity map. This analysis is performed in Section III-B. An additional method is 3D reconstruction from the disparity map, which is not focus of this paper.

B. Experimental Setup

The ZED stereo camera of Stereolabs is used for perception. The camera is mounted at the wall of the laboratory. The communication is based on the Robot Operating System (ROS) [11] and the calibration is based on OpenCV (version 3.1.0). The left and right mono camera (c.f. Fig. 1) are calibrated by the function `calibrateCamera()`, which uses the corners of a checkerboard in different positions and orientations (at least 10-20 pictures) [12]. One example picture for the calibration is depicted in Fig. 2. In the center of Fig. 2, the checkerboard for the calibration task is shown.

Fig. 2. Example picture, which is used for calibration of the left and right mono camera.

III. EXPERIMENTAL RESULTS

In this Section, the experimental results are discussed. In Subsection III-A, the stochastic characteristics of the reprojection error are analyzed. First results comparing the distance measurements by triangulation and disparity map are pointed out in Subsection III-B.

A. Reprojection Error

The reprojection error is the difference of the theoretical camera model and the measurement. The coordinates in the world coordinate system are projected into the image plane and compared with the corresponding image coordinates. As world coordinates, the corners of the checkerboard are used. The error is evaluated after successful calibration, because it can be assumed that the intrinsic and extrinsic parameters are optimal in this case. In the following, the error between the projected coordinates and the true coordinates of the checkerboard is analyzed. For 150 different positions of the checkerboard in the laboratory, pictures are taken for evaluation. In Fig. 3, the results of one example scenario are depicted. The reprojection error of all chessboard corners in the $x$-$y$ image plane is plotted. The results are very promising, as it seems to represent the stochastic characteristics in the form of a Gaussian. A deterministic error would suggest a model error, which would have to be considered in the model. The other pictures, differing in the position of the checkerboard, also suggest a Gaussian distribution.

For actual confirmation, a Quantile-Quantile (QQ)-plot [13] is performed. The QQ-plot compares the quantile of empirical results with the quantile of a theoretical distribution. In this paper, the empirical distribution is normalized and afterwards compared with a standard normal distribution. Points close to the line or on the line (c.f. Fig. 4) indicate that the empirical distribution equals the theoretical distribution. Outliers are indicated by single points, which have greater distance to the line.

The QQ-plot is performed independently in $x$- and $y$-direction. The results for the $x$-direction are depicted in Fig. 4.
for one example scenario. The representation clearly indicates that the empirical distribution equals a normal distribution, as all points are very close to the line and no distinct outliers are present. Additionally, the coefficient of determination $R^2 \approx 0.99$ shows that the distributions are almost equal, which should be the case for $R^2 = 1$.

As additional verification, a Shapiro-Wilk test [14] is performed. Results of the test are the $W$-statistic and the $p$-value, which both must be above a certain threshold for confirming the test and the hypothesis of normal distribution. For normal distribution, the $p$-value must be above an alpha value [15]. The alpha value is typically set to 0.05. The interpretation is as follows. If the normality test fails ($p$-value is below the alpha value), you are able to state with 95% confidence that the data does not fit the normal distribution, but if the normality test is passed ($p$-value is above the alpha value), you may only state that no significant departure from normality is found. The results for the $x$- and $y$-direction of the left and right camera for an example scenario are depicted in Table I. As all $p$-values are well above the chosen significance level, the Shapiro-Wilk test also supports the assumption that the error in $x$- and $y$-direction is normally distributed.

All three evaluations are performed for 150 different pictures (positions in the room) and indicate that the reprojection error can be assumed as a Gaussian normal distribution. These results are very promising, because this suggests that a Kalman filter is the optimal choice for tracking in the image plane.

### B. Distance Measurements

After stereo calibration (c.f. Fig. 1), rectification and distortion removal, the error distribution of the distance measurements is analyzed. As it is part of the current research, first results regarding the accuracy of the distance measurements are presented. Direct triangulation of the corners of the checkerboard [16] are compared with a depth map, which is generated by disparity map creation, Semi-Global-Blockmatching [17]–[20] with a block size of 9 pixels and additional smoothing by the DisparityWLSFilter [21]. For further information regarding the used algorithms, the interested reader is referred to the stated literature.

In Fig. 5, the depth picture after disparity map creation and Semi-Global-Blockmatching is depicted. It is obvious that noise artifacts are present which result from suboptimal block matching. Semi-Global-Blockmatching faces the biggest challenges at smooth surfaces (like walls or tables), because distinct features are not present. Nevertheless, the checkerboard is detected in the center of the camera and depth values of the room are mostly correct.

![Fig. 5. Depth picture, which results from disparity map creation and Semi-Global-Blockmatching without DisparityWLSFilter.](image)

To improve the accuracy of the depth maps, especially for smooth surfaces, a DisparityWLSFilter is used, which smoothes the depth values of the picture. An example image representing the results of the described image chain is depicted in Fig. 6. The DisparityWLSFilter smoothes the output of the Semi-Global-Blockmatching filter. As a consequence, noisy depth values are suppressed, but areas of equal depth are generated. In other words, the depth resolution of the picture is reduced, whereas the resistance against noise is improved. As the checkerboard is used for distance measure-

### TABLE I

<table>
<thead>
<tr>
<th></th>
<th>$W$</th>
<th>$p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>right</td>
<td>0.985</td>
<td>0.785</td>
</tr>
<tr>
<td>left</td>
<td>0.959</td>
<td>0.092</td>
</tr>
<tr>
<td></td>
<td>0.975</td>
<td>0.576</td>
</tr>
</tbody>
</table>

**Table I. Results of Shapiro-Wilk test for normal distribution in $x$- and $y$-direction in the distance of 2.5 m**
ments, the reduced depth resolution is not a problem, because the checkerboard occupies big areas in the picture, even in high distances to the camera.

In Fig. 7, the triangulated and true coordinates for a distance of 3.1 m are plotted. Even the simple triangulation method shows very good results, as most of the corners are located very accurately. A systematic error seems to be present, because all triangulated points are shifted to the bottom left corner in comparison to the true points. To analyze the distribution of the error, the error distribution between the true points and the triangulated points is fitted to a normal distribution in \(x\)- and \(y\)-direction and a QQ-plot is performed.

The results of the QQ-plot in \(x\)-direction are depicted in Fig. 8. The representation clearly indicates that the empirical distribution equals a normal distribution, as all points are very close to the line and no distinct outliers are present. Additionally, the coefficient of determination \(R^2 \approx 0.97\) shows that the distributions are almost equal. Further analysis shows that a systematic error is present with a mean value of \(\mu_x = -0.0154\) m and \(\mu_y = -0.0027\) m. Especially the systematic error in \(x\)-direction is quite significant and should be analyzed in more detail in the future.

In further distances, the systematic error increases, while QQ-plots in \(x\)- and \(y\)-direction still indicate that the error distribution is Gaussian. A Kalman filter would be only optimal in this case, if the mean value is also estimated in the state vector [8].

The depth map, which is generated by disparity map creation, Semi-Global-Blockmatching and additional smoothing by the DisparityWLSFilter is part of current research. Hence, first results regarding the mean value of the distance measurements of all checkerboard corners in different distances are presented. The results of both approaches are compared in Table II. The ground truth is based on accurate laser measurements and compared to triangulation and the depth picture resulting from the disparity map for three different distances 3.1 m, 4.4 m and 8.1 m. The first results indicate that the error increases with increasing distances, which are reasonable results, because with increasing distance the detection of the checkerboard corners gets more and more complicated, as the resolution of the partial image contains the checkerboard. Nevertheless, an error of 0.6 m is high for a distance of 8.1 m in the case of the depth picture. It indicates a nonlinear relation of the error depending on the distance. For simple triangulation, there are no reasonable results in the distance of 8.1 m, as the algorithm is not able to detect the corners with high confidence in both pictures. As a consequence, further research will target the error distributions for higher distances around 8 m.

![Figure 6: Depth picture, which results from disparity map creation, Semi-Global-Blockmatching and additional smoothing by the DisparityWLSFilter.](image)

![Figure 7: Triangulated and true coordinates of the checkerboard corners in the \(x-y\)-plane for a distance of 3.1 m.](image)

![Figure 8: Quantile-Quantile (QQ)-plot for the assumption of a normal distribution in world coordinates in \(x\)-direction. It is obvious that the error corresponds to a normal distribution.](image)

<table>
<thead>
<tr>
<th>Table II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Results of Shapiro-Wilk test for normal distribution in (x)- and (y)-direction in the distance of 2.5 m</td>
</tr>
<tr>
<td>Ground truth</td>
</tr>
<tr>
<td>Triangulation</td>
</tr>
<tr>
<td>Depth picture</td>
</tr>
</tbody>
</table>

**IV. Conclusions and Outlook**

In this contribution, a stereo image processing chain was presented, which was analyzed regarding the stochastic properties of the error distribution in the different steps of the chain. The reprojection error was analyzed in detail. The graphical method QQ-plot and the Shapiro-Wilk test were performed. Both methods strongly indicate that the reprojection error is Gaussian distributed. As a consequence, the Kalman filter can be assumed as optimal in the image plane. Additionally,
first results of distance measurements in the world coordinate system are presented. It was shown that for triangulation the error distribution seems to be Gaussian with a systematic error. The distance measurements clearly show drawbacks for high distances.

In the future, error distributions shall be analyzed for different distances in the world coordinate system in the case of triangulation and depth maps in more detail. Particular attention will be given to high distances.

REFERENCES


Posture and Fall Detection System Using 3D Motion Sensors

Hamza Qassoud *, Miodrag Bolic *, Sreeraman Rajan†
*School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Ontario, Canada
*Department of Systems and Computer Engineering, Carleton University, Ottawa, Ontario, Canada
Email: {hqass076@uottawa.ca, mbolic@eeecs.uottawa.ca, sreeramnr@sce.carleton.ca}

Abstract—This work presents a supervised learning approach for training a posture detection classifier, and implementing a fall detection system using the posture classification results as inputs with a Microsoft Kinect v2 sensor. The Kinect v2 skeleton tracking provides 3D depth coordinates for 25 body parts. We use these depth coordinates to extract seven features consisting of the height of the subject and six angles between certain body parts. These features are then fed into a fully connected neural network that outputs one of three considered postures for the subject: standing, sitting, or lying down. An average classification rate of over 99.30% for all three postures was achieved on test data consisting of multiple subjects where the subjects were not even facing the Kinect depth camera most of the time and were located in different locations. These results show the feasibility to classify human postures with the proposed setup independently of the location of the subject in the room and orientation to the 3D sensor.

Index Terms—Fall detection systems; Posture detection systems; Supervised learning; Machine learning.

I. INTRODUCTION

Two classes of devices can be used for posture classification: wearable (contact-based) and contactless. Wearable devices are attached to different parts of the human body. As an example, a strain sensor was used for measuring strain in the clothing of the subject, eventually leading to classify 27 different upper body postures [1]. In [2], various wearable sensors such as 3-axis accelerometers, 3D gyroscope and 3D magnetometer (3D compass) were placed on different body locations to compute a 3D orientation estimates with high accuracy and with no drift for both static and dynamic movements.

The contactless posture classification is mostly achieved using image and video processing principles. There are several devices such as radars, sonars, and cameras which can be used for posture classification. For instance, in [3] a background subtraction on the depth image of the human body to extract a silhouette contour of a human was presented. A horizontal projection of the silhouette contour was then employed to determine whether or not the subject is kneeling. When not kneeling, star skeleton technique was applied to the silhouette contour to obtain its feature points. These points were then used, alongside with the centre of gravity, to calculate the feature vectors and depth values of the body. These calculated values were then fed into a pre-trained LVQ (learning vector quantization) neural network to obtain one of the four possible postures: standing, sitting, stooping, or lying down. The proposed method in [4] used a head model to locate the human position. This work used image processing techniques such as edge extraction, template matching to achieve human detection. The HOG (histogram of oriented gradients) feature was then extracted from the depth images to get the characteristic vector of the original image. Finally, a generalized regression neural network was used to classify and identify human postures.

Our proposed work presents a method using only the Microsoft Kinect v2 sensor. We present a system to accurately classify the posture of the subject in any frame containing the human body in real-time. As training a neural network using supervised learning to perform classification can take a long time, a general purpose graphics processing units (GP-GPU) are leveraged to accelerate neural network training and runtime by executing in vector form on highly parallel and pipelined processing units rather than sequential processing with a Central Processing Unit (CPU) [5]. To accomplish speedup, TensorFlow or Theano is used to load and store data into and out of the GP-GPU memory and from the hard disk [6] [7]. An NVIDIA GeForce 940M GPU was used in this work.

In this paper, we present an efficient, and real-time posture classification system that uses 3D depth motion sensor inputs. We also implement a fall detection system by detecting whether the detected subject has been lying down on the floor for more than two seconds. We also provide the capability to different a subject from an inanimate object. We achieve our set objective of posture classification in three steps. In the first step, we extract features from the raw data that the motion sensor outputs which is presented in Section II. In the second step, we train a classifier using the features as input to determine the posture of the subject which is described in Section III. In the third step, using the posture classification output obtained second step, we detect falls as described in Section IV).

II. FEATURES EXTRACTION FROM MOTION SENSOR RAW DATA

The Microsoft Kinect V2 sensor uses infrared (IR) camera to recognize up to six users in its field of view and provides skeletal tracking features to up to two of these users. The sensor switches between 15 and 30 FPS (frames per second) depending on lighting conditions. Using the Kinect libraries, one can extract the 3D depth coordinates of the desired body parts.
Figure 1 illustrates the Kinect skeleton. The skeleton has 25 body joints that can be tracked. The X, Y, and Z position of these body joints relative to the camera can be extracted using the Kinect libraries in C++. Thus, we obtain 75 data points per frame per skeleton representing the 3D depth coordinates of each body part, or 2250 data points per second.

A simple approach to the posture detection would be to feed the 75 data points representing body joints locations into a neural network and classify the posture for each frame. However, this does not only slows down the classification process but also would require a very large amount of data, in hundreds of GB, since we would need our training data to be recorded from all possible angles and depths.

In order to circumvent the above issue, we extract location-independent features from the 75 data points representing the body joints depth coordinates. We extract seven features from these body joints. This will not only make the posture classification setting-independent, but it will also drastically reduce the amount of data collected since we now only extract 7 data points per body per frame instead of 75.

The Kinect skeleton provides the X, Y, and Z position of each one of the 25 body joints. As shown in Figure 2, we define the Y-axis to be the height (vertical location), the X-axis to be the width (horizontal location), and the Z-axis to denote the depth.

Figure 3 illustrates six of the seven extracted features from the raw data, named as follows: Left hip angle (1), right hip angle (2), left knee angle (3), right knee angle (4), chest angle (5), and chest-knee angle (6). Algorithm 1 shows us how the computation of the $\text{LEFTHIP}$ angle (one of the seven features) is done from the body joints depth coordinates using the cosine formula. The other five angles are computed in a very similar way. The seventh feature is the height of the subject which can be computed by taking the Y-position of the head and subtracting the lower Y-position value of the right and left foot from it, as shown in the Algorithm 2.

Although the Microsoft Kinect sensor estimates reasonably well the position of the body parts that it is unable to see, it may be good idea to check if the calculated height and angle features are not null before feeding them into the neural network.

### III. Posture Detection

Six people of different heights and shapes were asked to participate in the data collection phase. The data collection phase consisted of four five-minute rounds, where in each round, the subject was allowed to either move or stay in a fixed posture: that is standing for five minutes, sitting for five minutes, and lying down for five minutes. Since the camera had a frame rate of 30 frames per second, 1800 training
Three possible postures: standing, sitting, or lying down were training frames per posture in total. Subject per posture were collected which led to about 54000 frames per minute, or 9000 frames for each 5-minute round per subject.

# Use the cosine formula to determine the left hip angle
$\cos(\theta) = \frac{adjacent}{hypotenuse}$

Algorithm 1: The computation algorithm of the left hip angle feature from the raw body joints data.

Input: The X, Y, and Z position of the HIP_CENTER, the HIP_LEFT, and the KNEE_LEFT body joints.

Output: The left hip angle feature.

1. # Extract the body joints locations.
2. spineBasePos = joints[JointType_SpineBase].Position;
3. hipLeftPos = joints[JointType_HipLeft].Position;
4. kneeLeftPos = joints[JointType_KneeLeft].Position;
5. # Calculate the distance between the body joints.
9. # Use the cosine formula to determine the left hip angle feature.
10. leftHipAngle = acos((pow(hipKneeDistance, 2) + pow(spineHipDistance, 2) - pow(spineKneeDistance, 2))/(2 * hipKneeDistance * spineHipDistance)) * 180/PI;

Algorithm 2: The computation algorithm of the height feature from the raw body joints data.

Input: The X, Y, and Z position of the HEAD, the FOOT_LEFT, and the FOOT_RIGHT body joints.

Output: The height feature.

1. headPos = joints[JointType_Head].Position;
2. footLeftPos = joints[JointType_FootLeft].Position;
3. footRightPos = joints[JointType_FootRight].Position;
4. # $fmin(x, y)$ returns the smaller of its arguments: either $x$ or $y$.
5. height = headPos.Y - fmin(footLeftPos.Y, footRightPos.Y);

Features extraction was performed on the collected data. Three possible postures: standing, sitting, or lying down were considered as classes. For a fast posture classification, we first trained a fully connected 7-input 3-output and 0-(hidden) layer neural network in Tensorflow and trained it on the 54000 frames of each posture, or 378000 data points given that each training frame contains seven features. After the training phase, the posture classification accuracy of the neural network was tested on 20000 frames of labelled data collected on five different people who were not part of the training phase, and 96.08% classification accuracy was obtained.

A 1-layer neural network with 10 neurons in the hidden layer and 2-layer neural network with 10 neurons in each hidden layer were also implemented and trained as a goal to understand the trade-off between the classification accuracy and the processing time. We did not see a significant increase in the classification accuracy (accuracy was always less than 97%) with both the 1-layer and 2-layer networks. For a mere 1% increase in accuracy, significantly higher processing time was required for both neural networks. Given that the Kinect runs at 30 frames per second, and the motivation was to implement a real-time system, we decided to choose the 0-layer fully connected implementation for the posture classification classifier.

Further improvements were applied to the posture detection algorithm to handle real world scenario. One of the improvements was to make the height factor more significant than the other six features. For instance, the angles for both the standing and the lying down position are the same; hence, it is only the height factor that will provide discrimination between these postures and avoid classification errors. A threshold of 45 cm was chosen to differentiate the two postures: That is, if the subject’s height is less than this threshold, then the subject cannot be in the standing position and if it is greater, then the subject cannot be lying down. Defining the height as the Y-position of the head less than the Y-position of the foot as discussed in Algorithm 2 helps us avoid classification errors in many scenarios such as if the subject lays down but raises his hands above the 45cm threshold. In such as scenario, the system will still detect a height closer to 0 because the Y-position of the head and the foot will be close to each other. The improvements to the posture detection algorithm are presented in Algorithm 3.

We were able to achieve an improvement in the posture classification result of 99.30%, from 96.08%, on the same test data after incorporating the improvements as shown in Algorithm 3.

IV. FALL DETECTION

Falls can be detected if two conditions are met: (i) The subject has been lying down for a more than a specified amount of time; (ii) it is confirmed that it is not an object (such as bed, couch) that has been identified as on the floor. Figure 4 represents a summary of the whole fall detection system and Algorithm 4 implements the fall detection system based on the posture detection, and will be executed every time new posture data is made available to the system.
Algorithm 3: Improving the posture classification result based on the height factor.

**Input:** The features extracted from the raw body joints depth data which will give us a certain posture classification result from the neural network. The height feature is given in position 0.

**Output:** The improved posture detection result. Position 0 of the posture array is standing, position 1 is sitting and position 2 is lying down.

1. Extract the seven features for the current frame.
2. \( \text{features} = \text{extractFeatures}(); \)
3. \( \text{posture} = \text{getPostureClassification}(\text{features}); \)
4. \( \text{height} = \text{features}[0]; \)
5. **if** \( \text{height} < 0.45 \) **then**
   6. The subject cannot be standing in this case.
   7. \( \text{posture}[0] = 0; \)
   **else**
   9. The subject cannot be lying down in this case.
  10. \( \text{posture}[2] = 0; \)
**end**

Algorithm 4: Fall detection system based on the posture detection results.

**Input:** Posture detection results over a window of two seconds.

**Output:** Fall detection result

1. Retrieve two seconds of posture data (i.e. 60 frames at 30 FPS).
2. \( \text{postures} = \text{getPostureClassification}(\text{features}); \)
3. \( \text{lyingDownCount} = 0; \)
4. \( \text{fallDetected} = \text{False}; \)
5. **for each** \( \text{posture} \) **in** \( \text{postures} \) **do**
   6. **if** \( \text{posture} == \text{"LYINGDOWN"} \) **then**
      7. \( \text{lyingDownCount} = \text{lyingDownCount} + 1; \)
   **end**
9. **end**
10. A fall will be detected if and only if the subject has been lying down for more than 90% of the time (i.e. \( 90% \times 60 \text{frames} = 54 \text{frames} \)).
11. **if** \( \text{lyingDownCount} > 54 \) **then**
12. \( \text{fallDetected} = \text{True}; \)
13. **end**

As can be observed from Figure 4, a fall detection system requires to detect if a subject is lying down on the floor. In order to reduce possible false alarms, the locations of all the objects currently present in the room should be \textit{a priori} known. If subject’s current location is either overlapping or inside the boundaries of each object, then the subject can be declared as not lying down on the floor and hence no fall will be detected in this condition.

The method suggested above for detecting whether the subject is lying down on the floor or not requires the knowledge of the floor plan of the room. Alternatively, an object detection algorithm may be used to detect all the objects present in the room and their X and Z boundaries may be obtained from the location estimates. Then a decision can be made about fall by checking the subject’s location and comparing with the boundaries of the object. This may not be very efficient real time implementation since every object needs to be detected and located.

We propose a faster and reliable solution to determine whether the subject is lying down on the floor or not. The Algorithm in 5 first determines the floor level Y-coordinate relative to the camera by having an initialization step before the system starts. This initialization step will analyze the first three second of skeleton data and take the lowest recorded Y-coordinate of the left and right foot of the subject then store it. For the subject to be lying down on the floor, the Y-coordinate of his or her body joints cannot be much greater than this lowest Y-point. Using a safety threshold (say 25 centimetres), a decision that a subject is lying down on the floor is made if and only if the Y-coordinate of subject’s foot is less than the floor level Y-coordinate of the room plus the threshold.
In other words, the subject is lying down on the floor if and only if the Y-coordinates of subject's both feet are less than 25 centimetres away from the floor. This algorithm will be executed if the posture is classified as lying down.

**Algorithm 5:** Floor level detection and determining whether the subject is lying down on the floor.

**Input:** The Y-coordinate of the right and left foot of the subject over a window of three seconds. Next detected posture and the Y-coordinates of the right and left foot.

**Output:** Boolean representing whether the subject is lying down on the floor.

1. Initialization step. min() function returns the smallest number in an array.
2. `rightLeftFootYcoordinates = readSkeleton();`
3. `floorLevelY = min(rightLeftFootYcoordinates);`
4. `posture, footLeftY, footRightY = getNextPosture();`
5. if `posture == "LYING DOWN"` then
6.  Is the subject's' feet currently 0.25m away from the floor?
7.  if `footLeftY < (floorLevelY + 0.25)` &
8.     `footRightY < (floorLevelY + 0.25)` then
9.     return `True`;
10. end
11. return `False`;
end

To measure the accuracy of Algorithm 5, two people were asked to lie down on the floor, on a bed, and on a couch for three minutes in three different rooms. The system was able to recognize 100% of the time that the subject was not lying on the floor when the subject was lying on the bed or couch, and also detected 100% of the time that the subject was lying on the floor when the subject was indeed lying on the floor.

**V. CONCLUSION**

This work presents a real time fast and accurate implementation of a posture classification and fall detection system using the 3D body joints depth coordinates extracted from the Microsoft Kinect skeleton v2. These depth coordinates are used to compute seven features of the subject: height, left hip angle, right hip angle, left knee angle, right knee angle, chest angle, and the chest-knee angle. These seven features are used as inputs to a pre-trained neural network which then outputs three possible postures: standing, sitting, and lying down. Furthermore, a fall detection system is also implemented using the outputs of the posture classification system as well as the output of Algorithm 5 that determines whether the subject is lying down on the floor or on an object.

Limitations of this work include that only three posture have been tested (standing, sitting, and lying down). Furthermore, the system was not tested in the case where the subject is covered by a blanket, which may affect the accuracy of the Kinect skeleton.

Future work planned for the system is to implement the object detection algorithm discussed in Section IV and compare its performance and accuracy results with the proposed implementation in Algorithm 5 of determining whether a subject is lying down on the floor. In addition, an additional `transition` class will be added to the posture classifier to allow us to determine when the user switches posture. Lastly, more thorough testing of the system in realistic conditions will be done to potentially discover new limitations of the system.

**REFERENCES**


Fall Detection Based on Deep Saliency Images

Hongjun Li\textsuperscript{a,b,c,*}, Yupeng Ding\textsuperscript{a}, Chaobo Li\textsuperscript{a}, Ze Zhou\textsuperscript{a}
\textsuperscript{a} Nantong University, School of Electronic and Information
\textsuperscript{b} Nantong Research Institute for Advanced Communication Technologies
\textsuperscript{c} TONGKE School of Microelectronics
Nantong, China
lihongjun@ntu.edu.cn

Abstract—Fall detection is a hot topic in the context of aging society, which can reduce the fall-related injuries effectively. In this paper, a fall detection algorithm based on visual saliency via deep feature learning is proposed. Firstly, we use the Kinect to collect and label RGB images and depth images of fall events. Secondly, the 2-stream convolutional neural networks (CNN) is trained with RGB images and their labeled 0-1 maps to capture the saliency images, which can reduce the influences of complex scenes. Thirdly, training a CNN with two convolutional layers is used to verify the superiority of the saliency images in the fall detection. Finally, we experiment on the public library and our self-built dataset, and the averaged accuracy rate of this algorithm reaches 98.92%. Experimental results show that the algorithm can improve the effectiveness and accuracy rate of fall detection significantly.

Keywords—fall detection, visual saliency, deep features, convolutional neural networks

I. INTRODUCTION

Fall detection is a hot issue in human abnormal behavior detection\textsuperscript{[1]}. Especially, it is crucial for the elderly living alone. Statistics show that fall is the primary reason of injury related death for seniors aged 79 or more and the second leading cause of injury related (unintentional) death for all ages\textsuperscript{[2]}. And a survey shows that 35% people fall at least once a year among the seniors aged 65 or more\textsuperscript{[3]}. The demand for surveillance systems, especially for fall detection, has increased within the health care industry with the rapid growth of the population of the elderly in the world. In light of this, there is a great need for a real-time and accurate algorithm to detect a falling person.

The existing methods for fall detection that can be found in the literature may broadly be divided into two groups: those that use a variety of non-visual sensors and those exclusively vision-based\textsuperscript{[4]}. On the one hand, the methods detect a falling person by using a wearable device which has some sensors, but they have poor comfort and expansibility. On the other hand, the methods vision-based detect the fall by processing the image information obtained by the monitoring devices.

The major challenges to vision-based methods are how to determine the location of the objects, distinguishing which objects in a scene are falling and which are not. Most methods analysis the sharp of the object to detect the location, but people are non-rigid objects whose sharp are not consistent. In 2010, Kinect, a game peripheral by Microsoft, provides a possibility for the optimization of the previous human fall detection methods, which can determine the location of people (up to six people). And based on Kinect detection, it has more advantages of comfort. Rougier \textsuperscript{[5]} segmented the human body image from the foreground image and extracted centroid height. The distance between the centroid of the human body and the ground and the speed of the body centroid are used to identify the falling behavior. The algorithm can recognize slow sitting, squatting and other activities in the blocked scene, but it is easy to get false reports for fast action such as squatting rapidly to pick up things, sitting down quickly etc. Kepski and Kwolek \textsuperscript{[6-7]} combined Kinect and acceleration sensor to detect human fall events. This method detects foreground objects by different processing technology on the depth images, and then calculates the gravity center of human body and the distance between the gravity center of human body and the ground to assess whether the fall event occurred according to the state information and the predefined rules obtained from two kinds of sensors. Mastorakis and Makris \textsuperscript{[8]} established the human 3-dimensional bounding box and calculated the speed of height direction and the depth to width direction to detect fall, which has higher detecting accuracy in the process of walking. Ni et al.\textsuperscript{[9]} proposed a method to distinguish and detect human fall events by depth images.

Recently, deep learning methods have achieved excellent results for a variety of computer vision tasks, including action representation\textsuperscript{[10-11]}. Lin et al.\textsuperscript{[12]} presented a continuous deep learning model for fall detection using Microsoft Kinect. The input includes pre-processed high-resolution RGB images, depth images and optical flow images. They combined several deep learning structures including convolutional neural networks and long short-term memory networks for continuous human fallen detection. Feng et al.\textsuperscript{[13]} proposed a novel fall detection system using deep learning methods to analyse the postures in a smart home environment for detecting fall activities. Background subtraction is employed to extract the foreground human body. The binary human body images form the input to the classifier. Two deep learning approaches based on a Boltzmann machine and deep belief network are compared with a support vector machine approach. The decision on the occurrence of a fall is made on the basis of combining the classifier output with certain contextual rules. Jankowski et al.\textsuperscript{[14]} researched fall detection in elderly
II. RELATED WORK

The algorithm we present in this paper relates to three research directions: Kinect, saliency region detection, and convolutional neural networks for fall detection. These three issues are briefly reviewed in this section.

A. Kinect

Kinect is a peripheral that Microsoft generated for its Xbox game console, and it is the motion sensing device which takes natural body movements as input. Through projecting Infra-red on the object and calculating the time that every light beam needs to be received by receiver sensor, we can draw a depth map, which make it possible to the motion sensing technology in 3D environment. A Kinect device includes color, infrared, depth cameras, microphones and so on. It provides depth image, joint point tracking, language identification and other functions.

In Fig. 1, from left to right: RGB image, depth image and their labeled binary image (ground truth) are collected by Kinect in a scene. We can see that depth image is less affected by light and takes up less memory compared with RGB image.

B. Saliency Region Detection

Visual saliency computation can be categorized into bottom-up and top-down methods or a hybrid of the two. Top-down approaches use visual knowledge commonly acquired through learning to detect saliency. Bottom-up methods do not require training and rely on low-level features such as color contrast, pixel/patch locations, histogram, etc., for saliency detection.

In recent years, CNNs have been shown to be very effective on various visual recognition tasks, such as image classification, semantic segmentation and object detection. Wang et al.[16] solved the salient object detection by employing the Fast R-CNN framework. Kim et al. [17] trained a CNN to predict the saliency shape of image patches. The selective search method is first used to localize a stack of images patches, each of which is taken as the input to the CNN. Li et al. [18] leveraged both high-level features from CNNs and low-level features extracted based on hand-crafted methods. To enhance the generalization and learning ability of CNNs, the original R-CNN is redesigned by adding local response normalization to the first two layers.

C. Convolutional Neural Networks

Convolutional neural network is a feed forward neural network whose artificial neurons can respond to a part of the surrounding units in the coverage area, which is excellent for image processing. It includes one or more convolutional layers, pooling layers, full connected layers and one classification layer. The classic convolutional neural network is shown in Fig.2. The width and height of the input layers correspond to the width and height of the input images. After the first convolution layer, the pooling layer takes a sub-sampling of feature maps and gets smaller feature maps. The last two layers are full connection layers. All neurons in two full connection layers connect with each other. The output of the whole network is obtained.
III. FALL DETECTION BASED ON DEEP SALIENCY IMAGES

The algorithm that we present in this paper can be divided into two sides: saliency maps detection and fall detection via deep features. The two issues are reviewed in this section.

A. Saliency Maps Detection

One major challenge to vision-based fall detection methods are how to determine the location of the objects. Saliency object detection, which is the prediction of where a human being will look in an image, has attracted considerable research interest in recent years. It serves as an important pre-processing step for many tasks, such as image classification, image retargeting and object recognition. It is not only useful for RGB images, but also depth images.

In this work, we train a 2-stream CNN model to extract saliency maps, which is shown in Fig. 3. The CNN has one input layer, five convolutional layers, four fully connected layers and one output layer. Firstly, we input a RGB image to the model, which will be cropped into two parts: the entire image and the location of saliency region. Secondly, the two images will be fed into the model respectively. Thirdly, we can get two 4096-vectors (feature a and feature b) from the second fully connected layer. And then connect them in series to fusion vector. Finally, the 2-stream CNN is fed into the neural network with two fully connected layers and one output layer, which is trained a lot of RGB images and their labeled saliency maps with 0 and 1 respectively. This network is used as a regressor to evaluate saliency score of every region in an image in accordance with the features extracted from 2-stream CNN.

During the fusion stage, two different features are connected in series, the process is expressed as:

\[ f_x = \alpha f_a + (1 - \alpha) f_b \]  

where \( f_x \) is the fusion feature, \( f_a \) is feature captured from the entire image, and \( f_b \) captured from the location of saliency region.

Given an image I, we represent it by a set of M-level segmentations \[ S = \{S_1, S_2, ..., S_M\} \], where each segmentation \( S_m \) is a decomposition of the image I and consists of \( K_m \) regions. \( S_1 \) is the finest segmentation consisting of the largest number of regions, and \( S_M \) is the coarsest segmentation consisting of the smallest number of regions. For each level, we assign the saliency score of each region, that gets from the output of 2-stream CNN, to its contained pixels so that we can capture M saliency maps \( S = \{S_1, S_2, ..., S_M\} \), which is shown in Fig. 4. And then the M saliency maps are fused together, which is expressed as:

\[ S_f = \sum_{m=1}^{M} w_m S_m \]  

where \( S_f \) is the final saliency image, and the \( w_m \) is the weight of feature map fusion. The weights are updated with the sum of the losses of all training images:

\[ K^M = S_g - \sum_{n=1}^{M} w_n S_n \]  

where \( K^M \) is the sum of the losses, and \( S_g \) is the ground truth of an image. And then we minimize the losses to get the finest weights.

B. Fall Detection via Deep Features

Through the algorithm in the previous section, we can get saliency images from our self-built dataset and UR Fall Detection Dataset (URFD) [20], which are shown in Fig. 5. In this section, we extract deep features of saliency images based on CNN. In Fig. 6, saliency images are taken as inputs and the recognition of fall or non-fall are produced as output, and the width and height of the input layer correspond to the width and
height of the input image. Our network consists of two convolutional layers followed by a fully connected layer and an output layer with a sigmoid nonlinear function. Following the convolutional layers, we add a max pooling layer respectively for translation invariance. We adopt the sigmoid function as the activation function for the two convolutional layers. After several convolutional and pooling layers, fully connected layers are built and trained using the output of the previous layer as input.

\[ g(x) = g_L(\ldots g_2(g_1(x; w_1); w_2)\ldots w_L) \]  

(4)

where each function \( g_i \) takes as input \( x \) (input image, or the feature map in the previous layers) and a parameter vector \( w_i \) (weight to be trained); where \( L \) is the number of layers and \( g(x) \) is output of the neural network.

For a classification problem with \( N \) training examples and \( M \) classes, the squared-error is expressed as:

\[ E^N = \sum_{n=1}^{N} \sum_{m=1}^{M} (t_m^n - y_m^n)^2 \]  

(5)

where \( t_m^n \) is the value of the \( m \)th dimension about \( n \)th pattern’s corresponding label, and \( y_m^n \) is the \( m \)th output layer unit related to \( n \)th input pattern. In our work, \( M = 2 \), which says fall or non-fall. Though cross-entropy loss is typically used rather than squared loss, there was no significant effect on the results in our experiment.

C. The Flow Chart of Algorithm

The flow chart of algorithm is shown in Fig. 6. Firstly, a Kinect sensor is used to collect and label RGB images and depth images. Secondly, a saliency object detection model is trained with RGB images and their labeled 0-1 maps to capture the saliency images, which can reduce the influences of complex scenes. Thirdly, we train a CNN with two convolutional layers to verify the accuracy rate of our algorithm. Finally, we experiment on the public library and our self-built dataset.

IV. EXPERIMENT

We conducted our experiments on a PC laptop Intel Core i7-6800K 3.40 GHz CPU, NVIDIA GTX 1080 (8 GB on-board memory) GPU and 16GB RAM. Our system is mainly implemented in MATLAB R2014a 64-bit and Caffe [21], an open source framework for CNN training and testing on Windows 10. In the simulation experiments, the algorithm adopts the same parameter configuration. In order to verify the effectiveness, the algorithm is researched in the self-built database.

A. Datasets

UR Fall Detection Dataset [20] contains 70 (30 falls + 40 activities of daily living) sequences. Fall events are recorded with 2 Microsoft Kinect cameras and corresponding accelerometric data. Each fall contains sequence of depth and RGB images for camera 0 and camera 1 (parallel to the floor and ceiling mounted, respectively), synchronization data, and raw accelerometer data. In this work, we just select 30 falls included in the dataset, which is shown in Fig. 7. And the experimental results on the UR Fall Detection show in table I. By comparison, we can find the excellent performances of the saliency images in the fall detection. Whether the camera is in the top or the front, the proposed method has better performance than the methods in [20].
Our self-built dataset contains 2100 images with pixel wise annotation of salient objects. These images are collected by Kinect, and each image can be divided into three types: a RGB image, a depth image and their labeled saliency map (a ground truth). The dataset includes 5 kinds of actions: standing, sitting, bowing, on all fours, and lying on the floor, which is shown in Fig. 8. The first three actions are defined to non-fall, while the rest of actions are defined to fall. In order to illustrate the practicability of the algorithm, we collect data in different environments. We also divide self-built dataset into three parts, 1200 images for training, 300 images for validation and the remaining 600 images for testing. The images for training randomly chosen from the five kinds of actions, 240 images of each kind. And then the remaining are used for validation and testing.

The 2-stream CNN is composed of two deep convolutional neural networks in parallel originally trained over the ImageNet dataset [22] using Caffe. And then we use the selected 1200 RGB images and their labeled saliency maps to train the 2-stream CNN for extracting saliency features. Finally, we can acquire 600 images after testing, which are used to verify the effectiveness of the algorithm based on saliency features.

### B. Accuracy Rate of the Algorithm

In order to prove the accuracy of the algorithm, we performed the experiments with 600 saliency images obtained from RGB images, in which 100 images for train and 500 images for testing. The experimental results show in table II, using saliency images gives better performance compared to RGB images and depth images. It is worthwhile to note that saliency images can improve the accuracy and efficiency of recognition. Since the previous work is aimed at the depth images, it has better recognition performance than the proposed algorithm in the depth images. In order to further verify the reliability of the algorithm, we conducted experiments separately under different training samples and test samples. The experimental results are shown in Fig. 9. By comparison, we can find the excellent performances of the saliency images in the fall detection.

<table>
<thead>
<tr>
<th>Images</th>
<th>RGB image</th>
<th>Depth image</th>
<th>Saliency image</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>40.21%</td>
<td>59.62%</td>
<td>98.92%</td>
</tr>
<tr>
<td>Previous work[23]</td>
<td>42.90%</td>
<td>89.20%</td>
<td>/</td>
</tr>
</tbody>
</table>

Convergence is an important index to evaluate the quality of an algorithm. In the fall detection, the faster the algorithm converges, the faster the detection speed will be, the higher the possibility of reducing the fall injury will be. In order to verify the performance of the algorithm, we conducted experiments on three different kind of images: saliency images, depth images and RGB images. As can be seen from Fig.10, the saliency image is used to detect a person whether is falling much faster than the other two images. When we use the saliency images to detect a falling man, we can reduce the fall injuries because of the improvement of the detection speed.

![Fig. 10. The square-error of different images](image-url)
V. CONCLUSION

According to the human abnormal behavior detection problems, a fall detection algorithm based on saliency features is proposed. Firstly, this algorithm can effectively reduce the interference of complex background. Secondly, using saliency images to detect a falling person can improve the accuracy rate. The efficiency and precision are improved effectively compared with RGB images. In practice, falls are likely to be more sudden than images collected from the laboratories. In future work, we would like to research saliency detection based on natural images and multi-object fall detection.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers and the editors for their valuable suggestions. This work is supported by the National Natural Science Foundation of China (NO.61601248); the University Science Research Project of Jiangsu Province (NO.16KJB510036); the Science and Technology Program of Nantong (NO.MS12016025); Top-notch Academic Programs Project of Jiangsu Higher Education Institutions (NO.PPZY2015B135); Nantong University-Nantong Joint Research Center for Intelligent Information Technology (KFKT2017B04); Innovation Program for College Student (No.201710304021Z).

REFERENCES

Online Semi-Supervised Learning with Adaptive Vector Quantization

Yuan-Yuan Shen, Xu-Yao Zhang, Cheng-Lin Liu

1 National Laboratory of Pattern Recognition, Institute of Automation of Chinese Academy of Sciences
95 Zhongguancun East Road, Beijing, P.R. China
2 University of Chinese Academy of Sciences, Beijing, P.R. China
3 CAS Center for Excellence of Brain Science and Intelligence Technology, Beijing, P.R. China
Email: shenyuanyuan2015@ia.ac.cn, {xyz, liucl}@nlpr.ia.ac.cn

Abstract—This paper considers the online semi-supervised learning (OSSL) problem in which the data are a mixture of both labeled and unlabeled samples and appear in a sequential (stream) manner. OSSL is very common in real applications and similar to the human-like learning process. Prototype-based classifiers, which represent the data of different classes by some prototypes, are natural in a streaming scenario by updating the prototypes with online (incremental) learning. However, most of previous prototype-based models are either designed for supervised or unsupervised learning separately. In this paper, we propose a novel model called online adaptive vector quantization (OAVQ) aiming at improving the classification performance in case of OSSL. Specifically, we use the learning vector quantization (LVQ) criterion for updating the prototypes when the data point is labeled, and the frequency sensitive competitive learning (FSCL) criterion for adjusting the prototypes when the data point is unlabeled. The labeled and unlabeled data are coming randomly in a sequential manner, and these two criteria are used alternatively to learn the positions of prototypes. In this way, we can make full use of both supervised and unsupervised information to further boost the performance. Experiment results on several databases verify the effectiveness and applicability of the proposed method in improving the performance for OSSL.

Index Terms—learning vector quantization, frequency sensitive competitive learning, adaptive vector quantization, online learning, semi-supervised learning

I. INTRODUCTION

Pattern recognition methods can usually be partitioned into supervised and unsupervised models. A more general case is semi-supervised learning (SSL) which makes use of both the labeled and unlabeled data to learn the classifier. Since we can easily collect large amount of unlabeled data compared with the complicated process of preparing labeled data, SSL usually achieves much better performance than supervised and unsupervised learning with the situation of only a small number of labeled samples [1], [2]. Moreover, in many real world applications, the training process has to incrementally learn the model on a stream of patterns, which means the models need to be adjusted over time when new patterns (either labeled or unlabeled) appear, and this kind of process is called online semi-supervised learning (OSSL). In this setting, at each time, we may have only one or a small number of data points and the whole training data are usually not stored in the learning process. Furthermore, in some cases, the algorithms are restricted to a situation of single-pass learning where each data point only appear once. OSSL is similar to the human learning process and becomes very common in real application. How to efficiently and effectively make use of labeled and unlabeled information in an online fashion makes OSSL a challenging problem.

Prototype-based methods represent the training data with a set of points in feature space (called prototypes), and have found their utility in wide range of problems. Generally, there are two kinds of prototype-based learning paradigms, namely supervised classification models which learn several prototypes as typical representatives for each class, and unsupervised clustering models in which a set of prototypes are treated as an approximate representation for the whole dataset.

Learning vector quantization (LVQ) [3] as a well known prototype learning algorithm has been widely developed in supervised learning. Kohonen et al. proposed a number of improved versions of LVQ such as LVQ2, LVQ2.1, and LVQ3 [4]. Crammer et al. [5] showed that LVQ fell into a family of maximal margin learning algorithms providing a rigorous upper bound of generalization error. Jin et al. proposed optimizing the log-likelihood of hypothesis margin (LOGM) [6] for improving the convergence of training and generalization performance.

However, most existing LVQ methods are usually trained in batch mode. In order to cope with online learning, many algorithms [7]-[9] have been proposed. Online learning vector quantization (OLVQ) as an important type of online prototyped learning algorithms has been researched extensively in [10]-[13]. When a new pattern is arriving, two nearest prototypes from the positive class (the prototype which has the same class label with the input pattern) and the rival class (the prototype which has the different class label with the input pattern) are updated, or a new prototype is added in OLVQ methods. The main difference among those OLVQ methods is the manner of how to introduce new prototypes. In [10] and [11], new prototypes are added according to the increase of some error mechanism. When the number of new samples which are misclassified reaches to the predefined error number, a new prototype is added. In [12] and [13], new prototypes are added by a predefined threshold. When a new sample is arriving, the distance between the new sample and the nearest prototype is computed. If the distance is bigger than
In this paper, to deal with the challenging OSSL problem, we propose a new model called online adaptive vector quantization (OAVQ) which combines the supervised and unsupervised prototype based learning methods in an unified framework. Specifically, the model is initialized using several initial labeled training samples, then the incremental mixed labeled and unlabeled samples are continually used for updating the prototypes. The prototypes are updated and shared for both supervised and unsupervised processes, and therefore, the class information of labeled data can be transferred to unlabeled data, while the clustering information from unlabeled data can be used to improve the classification performance. If the coming sample is labeled, then prototypes are updated by the mode of online learning vector quantization (OLVQ); otherwise analogy to the self-training, the nearest prototype from the unlabeled sample is being updated based on a modified objective function. Inspired by [15] which verifies the effectiveness of considering the update time of each prototype in unsupervised learning, the frequency-sensitive strategy is exploited in our paper. In this case, update time (frequency) is maintained for each prototype, the distance between sample and prototypes in unsupervised learning is redefined by the original distance between sample and the prototype and the frequency. When the prototype is computed as the nearest prototype, then the frequency of this prototype is increased by one. The less one prototype is updated in unlabeled situation, the easier it is chosen as the winner prototype for unlabeled sample. The flow diagram of our method is illustrated in Fig. 1.

The rest of the paper is organized as follows. Section II describes the proposed OAVQ model, section III reports the experimental results, and the last section concludes this paper.

II. ONLINE ADAPTIVE VECTOR QUANTIZATION (OAVQ)

In this section, firstly we introduce our proposed online adaptive vector quantization (OAVQ) model, then we give detailed discussions on how to learn from labeled and unlabeled samples respectively.

A. OAVQ Framework

Suppose that we observe a sequence of feature vectors $x_1, x_2, \ldots, x_n$ where $x_i \in \mathbb{R}^d$ is a pre-defined feature representation. The $x_i$ can be either labeled or unlabeled, if it is labeled, a class label $y_i$ is also given. Till time $T$, $N_1$ is the total number of labeled samples we have seen, and we use $N_2$ to represent the total number of unlabeled samples. The model parameters need to be optimized are denoted by $m$. Then, the objective function for online semi-supervised learning (OSSL) through time $T$ can be summarized as:

$$ F = \min_{x_i, y_i, m} \sum_{i=1}^{N_1} S(x_i, y_i, m) + \lambda \sum_{j=1}^{N_2} U(x_j, m), $$

where the hyperparameter $\lambda$ controls the trade-off between supervised learning and unsupervised learning, and $S(\cdot)$ and $U(\cdot)$ are the loss functions for labeled samples and unlabeled samples respectively.
Specifically, in our proposed online adaptive vector quantization (OAVQ) model, two prototype-based criteria (i.e., OLVQ criterion for supervised learning and FSCL criterion for unsupervised learning) are combined to solve the OSSL problem. Here the model parameters refer to the prototypes which are shared in supervised and unsupervised learning. Detailed learning procedures are described in section II.B and section II.C, and we summarize the OAVQ process in Algorithm 1.

**Algorithm 1 OAVQ**

**Require:** labeled sample \((x, y)\) or unlabeled sample \(x\) prototypes \(m\)

1: while receive new pattern \(x\) do
2: if \(x\) is labeled then
3: update prototypes based on OLVQ criterion
4: else
5: update prototypes based on FSCL criterion
6: end if
7: end while

**B. Supervised Model Learning: OLVQ**

In this sub-section, we describe the learning criterion of OLVQ for labeled samples. For a \(M\)-class classification problem, prototype learning is to learn a set of prototype vectors \(m_{i,j} (i = 1, 2, \ldots, M; j = 1, \ldots, n_i)\) for each class. Here \(n_i\) is the number of prototypes in class \(i\). The learning process is usually implemented by minimizing the empirical loss on a training set. An input pattern \(x \in \mathbb{R}^d\) is classified to the class of the nearest prototype:

\[
k = \arg \min_{i=1}^{M} \min_{j=1}^{n_i} \| x - m_{i,j} \|_2^2 = G(x, m),
\]

(2)

There are many variations of LVQ algorithm [3], [4], [6]. In this paper, we use the minimum classification error (MCE) [6], [18] criterion due to its good performance.

Specifically, given that \(m_1\) and \(m_2\) are two nearest prototypes to pattern \(x\) from the positive class and the rival class respectively, the posterior probability of \(x\) belonging to genuine class \(y\) (i.e., the probability of correct classification) can be approximated by the sigmoid function \(\sigma\):

\[
P(c|x; m) = \sigma(\xi_1 d(x)),
\]

(3)

where \(\xi_1 (\xi_1 > 0)\) is a constant for tuning the smoothness of sigmoid function and \(d(x)\) is computed as \(d(x) = \| x - m_{1} \|_2^2 - \| x - m_{2} \|_2^2\). Then the conditional log-likelihood loss of pattern \(x\) is \(S(x, y, m) = 1 - P(c|x; m)\).

As new pattern \(x\) is arriving, OLVQ updates the two prototypes \(m_1\) and \(m_2\) by gradient-based methods [19], [20]:

\[
m_1 = m_1 - \eta \frac{\partial S(x, y, m)}{\partial m_1}, \quad m_2 = m_2 - \eta \frac{\partial S(x, y, m)}{\partial m_2},
\]

(4)

where \(\eta\) is the learning rate.

**C. Unsupervised Model Learning: FSCL**

In this sub-section, we focus on the problem of how to update the prototypes for unlabeled samples. Based on the FSCL criterion, the prototype that provides the highest similarity to the given input pattern is declared as the winner node and is moved closer to the input pattern, whereas the rest of the prototypes are left unchanged.

Specifically, given that \(x\) is the arriving unlabeled sample at current time, then the probability of \(x\) belonging to prototype \(m\) can be approximated by the sigmoid function \(\sigma\) (similar to OLVQ):

\[
P(x \in m) = \sigma(\xi_2 d(x)),
\]

(5)

where \(\xi_2 (\xi_2 > 0)\) is a constant for tuning the smoothness of sigmoid function and \(d(x) = \| x - m_{h^*} \|_2^2\) \((m_{h^*}\) is the winner prototype where \(h^* = \arg \min_h \| x - m_h \|_2^2\)). Based on the winner-take-all strategy, only the nearest prototype is considered in the definition of affiliation function.

When frequency is introduced, suppose that \(m_{h^*}\) is the winner prototype, then \(h^*\) is computed as:

\[
h^* = \arg \min_h n_h \| x - m_h \|_2^2,
\]

(6)

where \(n_h\) is the frequency of the prototype. Once a prototype is updated by the unlabeled sample, then the relative frequency of this prototype \(n_h\) is increased by one. Obviously, this strategy can make a balance in all the prototypes for unsupervised learning. If a prototype is updated for only a few times, then it is easier to be chosen as the winner prototype. In this way, all prototypes can be well activated in the learning process.

Similarly, the loss function for unlabeled sample \(x\) can be defined as:

\[
U(x, m) = -f \ast (1 - P(x \in m_{h^*})),
\]

(7)

where \(f\) is the degree of confidence. \(f\) is calculated as:

\[
f = \sigma(\xi_2 (\| x - m_{1} \|_2^2 - \| x - m_{2} \|_2^2)),
\]

(8)

here \(m_1\) and \(m_2\) are two nearest prototypes with the sample \(x\) from two different classes. Estimating the uncertainty from two best predicted classes [21] has been proved to be very useful in active learning. If the margin for the distance is larger, the confidence should be larger for this sample.

As unlabeled new pattern \(x\) is arriving, the nearest prototype \(m\) is also updated by gradient-based methods [19], [20]:

\[
m_{h^*} = m_{h^*} - \eta \frac{\partial U(x, m)}{\partial m_{h^*}},
\]

(9)

and here \(\eta\) is the learning rate.

Since we have made some modifications to the traditional FSCL, we summarize the learning process in Algorithm 2. Instead of increasing by one for frequency \(n_h\) in each update, a decreasing sequence parameterized by \(L\) is used here to improve the performance. One explanation for this setting is that as the algorithm tends to be stable over time, the effect from the frequency should be decreased.
III. EXPERIMENTS

In this section, we conduct experiments on both artificial dataset and real-world datasets to evaluate the model. An artificially generated dataset is used firstly to illustrate the effectiveness of incorporating unlabeled data into the learning process. The proposed algorithm is then applied to some real-world datasets to further verify its performance.

A. Artificial dataset

The artificial dataset consists of three Gaussian distributions, each representing a class. The dataset consists of 300 points in total with 100 points for each class. The distributions are centred at $[0, 0], [-1.5, 1]$ and $[2, -1.5]$ respectively, with the covariance matrix $[1 \ 0; 0 \ 2], [1 \ 0; 0 \ 2]$ and $[1 \ 0; 0 \ 2]$. The dataset and the learned prototypes for different methods are figured in Fig. 2. Solid bold circles represent the learned cluster centers are viewed as the initial prototypes. For the patterns from each class are clustered into $P$ clusters and the number of prototypes as $K$.

In order to illustrate the benefits of unsupervised learning, we omit the supervised process in this experiment. In the above four figures, Fig. 2a shows scatter plots of all training data. Then we randomly choose two initial prototypes for each class in Fig. 2b. By continuing to learn models from unlabeled samples in the manner of CL and FSCL respectively, Fig. 2c and Fig. 2d are produced. Here we take the learning process of the two prototypes from the class which are painted black as an example. From Fig. 2b we observe that the initial two prototypes are too close to the right and are not benefit to the process of the classification. By learning the distribution information from unlabeled samples with CL criterion, in Fig. 2c we can see that the left one of the two prototypes are shift to the left. Thus the two prototypes become a better representation of the class which are painted black. Furthermore, online learning vector quantization (OLVQ) is sensitive to the initial prototypes. A typical case in our data is the right prototype of the labeled black. In Fig. 2c, it rarely can be updated or wrongly updated by the pattern from the class which are painted blue. In Fig. 2d FSCL strategy is considered and we can obtain more reasonable prototypes by using this approach.

B. Real-world data set

To test our proposed method on realistic data, we conduct experiments with the UCI datasets including: (a) dna, (b) pendigits, (c) usps, (d) mnist. The statistics of the above datasets are listed in Table I. As discussed in previous sections, the results of online learning are usually influenced by the order of the samples. In order to yield stable results, the experiment for each dataset is implemented for 20 times, and for each time the appearing of the samples are arranged randomly. In our experiments, the first 20 patterns from each class are used to train the initial prototypes. Specifically, we predefine the number of the prototypes as $P$. The first 20 patterns from each class are clustered into $P$ clusters and the cluster centers are viewed as the initial prototypes. For the same dataset, the number of prototypes for different methods is identical. For all the remaining data, we randomly choose a certain percentage of samples as labeled data. Then the mixed data with labeled and unlabeled samples are arriving one-by-one randomly. Adagrad algorithm [20] is exploited for updating the model in our method.

For all datasets, the balance parameter $\lambda$ is set as 0.01. The hyperparameters $\xi_1$ and $\xi_2$ are initialized from the training samples. Suppose that the number of the initial training samples is $K$, then $\xi_1$ is estimated as $\frac{1}{K} \sum_{i=1}^{K} ([x_i - m_2]^2 - [x_i - m_1]^2)]$ (m1 and m2 are two nearest prototypes to pattern $x_i$ from the positive class and the rival class respectively) and $\xi_2$ is estimated as $\frac{1}{K} \sum_{i=1}^{K} [\|x_i - m_1\|^2]$.
The number $P$ of prototypes is chosen from 3 to 10 and the initial learning rate of gradient descent is chosen from $\{0.01, 0.1, 0.2\}$. The number of prototypes and the initial learning rate are set as Table II.

Table III shows the results of OLVQ, OLVQ+CL and OA VQ respectively. OLVQ is a purely online supervised classifier, while OLVQ+CL and OA VQ are the online semi-supervised learning models. Learning from unlabeled samples is based on CL strategy and FSCL strategy in OLVQ+CL and OA VQ respectively. Different percentages (from 1% to 100%) of samples are labeled to evaluate the effectiveness of the proposed method. Best performance of the compared methods are given in bold. From Table III, we can observe that when the percentage of labeled samples is less than 10%, OLVQ+CL and OA VQ have a clear advantage over OLVQ. Usually the unlabeled data we use is sampled from the same distribution with the labeled data, then the data distribution information contained in unlabeled data is also helpful to the classification. OLVQ+CL/OAVQ introduces the learning of unlabeled samples in OLVQ and the better classification performance is obtained than OLVQ. This is useful in the situation that the training set is the combination of small portion of labeled samples and a large portion of unlabeled samples. However, as the amount of labeled data increases, OLVQ+CL and OA VQ show a very weak advantage over OLVQ even the classification performance of OLVQ+CL/OAVQ is slightly inferior to OLVQ. This is caused mainly by the accumulative error of learning unsupervised samples. For unlabeled data, the winner prototype is possible to come from the different class with the arriving sample (i.e., the prototype should be updated is not been adjusted, and meanwhile the prototype should not be updated is adjusted). Semi-supervised models balance the advantage of distribution information and the disadvantage of the wrongly updating. When the labeled samples are in the majority, the distribution information contained in the labeled samples is enough. Then the advantage of distribution information is weakened and the accumulative error from unlabeled samples become the main factor. Finally, considering the updating frequency of each prototype from unlabeled samples, OA VQ tends to choose prototypes which have the lower frequency in unsupervised model learning and always has better performance than OLVQ+CL or comparable performance to OLVQ+CL.

**IV. CONCLUSION**

In this paper, we consider an important problem of online semi-supervised learning (OSSL). OSSL is very useful in certain real applications such as online tracking, where potentially unlimited data arrive sequentially, which cannot be entirely stored and only a small fraction of them are labeled. We propose a new model called online adaptive vector quantization (OA VQ) to solve this problem which includes two basic components: the OLVQ criterion for dealing with labeled data and the FSCL criterion for handling unlabeled samples. Experimental results show the effectiveness of OA VQ on both artificial and real datasets. Our future work will consider extending OA VQ to other challenging problems such as class-incremental learning.

**ACKNOWLEDGMENT**

This work has been supported by the National Natural Science Foundation of China (NSFC) Grants 61411136002, 61633021 and 61721004.

**REFERENCES**


Pattern Structure of Human Motion Using Single Channel CW Doppler Radar: An Unsupervised Perspective

Julio J. Valdés
Digital Technologies Research Centre
National Research Council Canada
Ottawa, Canada K1A 0R6
julio.valdes@nrc-cnrc.gc.ca

Zachary Baird
Department of Systems and Computer Engineering
Carleton University
Ottawa, Canada
zachbaird@cmail.carleton.ca

Sreeraman Rajan
Department of Systems and Computer Engineering
Carleton University
Ottawa, Canada K1N 6N5
sreeramanr@sce.carleton.ca

Miodrag Bolic
School of Electrical Engineering and Computer Science
University of Ottawa
Ottawa, Canada
mbolic@site.uottawa.ca

Abstract—The accurate estimation of vital signs in human subjects during periods of gross body movements using Doppler radar is a challenging goal. The degradation in accuracy and confidence of estimates should be anticipated when the radar returns are contaminated by large body movements. Since the presence of body movements may not be known a priori, it is necessary to have algorithms in place to determine the presence of large body movements before estimation of vital signs is attempted. Therefore, knowledge of the internal properties and the structure of the data are crucial. From an unsupervised perspective, this paper makes an initial study of the properties of radar return signals acquired using a single channel CW Doppler radar system, described in terms of 43 features derived from the raw radar information. The intrinsic dimension of the data is estimated using several techniques and used for approximating embedded low-dimensional manifolds containing the data, constructed with linear and nonlinear methods. Three dimensional versions of these mappings are presented for visualization purposes and compared with a class distribution of movement given by sedentary and still, sedentary and moving, or walking. It is found that a substantial dimensionality reduction is possible and that the internal structure of the data distribution in the lower dimension spaces exhibits a good potential for class differentiation.

Index Terms—Dimension reduction, unsupervised learning, human activity monitoring.

I. INTRODUCTION

Since the 1970s researchers have been experimenting with monitoring physiological signs of humans and animals with Doppler radar [1]. Recent advances in radar hardware and breakthroughs in digital signal processing techniques have allowed advances like localization of humans [2], estimation of breathing, heart rate [3] [4], and even heart rate variability [5] with high accuracy and reproducibility. One major limitation is the inability to accurately estimate vital signs during periods of gross body movements [5]. At most, small movements such as typing on a laptop or smart phone while the subject is sedentary are admissible for a radar physiological monitoring system to obtain reliable estimates [6]. However, a vital sign monitoring system in uncontrolled scenarios must be able to provide reliable vital sign estimates irrespective of whether the subject is sedentary or moving.

II. METHODOLOGY AND MATERIALS

1) Radar: The radar used in the data collection for this paper is a single channel binary phase coded Continuous Wave (CW) Doppler radar prototype model built by K&G Spectrum with an operating frequency of 24.125GHz. The radar has four transmit and receive antenna pairs adjacent to one another, each with beamwidth of 20 degrees horizontal and 70 degrees vertical, ensuring that the entire room can be covered. The antennas are connected to the radar transceiver through a single pole four throw switch meaning only one pair of antennas is on at a time, thus the radar operates as a single input-single output radar. During the data recording only one antenna pair was used and the subject was only in the direction of that antenna. The radar software allows for up to 50 0.75m overlapping zones to be observed and recorded simultaneously. In this experiment only 9 zones were recorded which covered the distance from the antenna to the opposite corner of the room. The sampling rate of the radar is 905 Samples/sec when 9 zones are observed.

2) Classes: Three classes of activity level were considered:
   • Sedentary and still (SS-class): the subject is sitting, standing or lying in place, breathing normally or holding their breath and are not moving any body part.
   • Sedentary and moving (SM-class): the subject is sitting, standing, or lying in place and breathing normally or holding their breath and are moving one or more body parts.
parts. The level of movement as well as the body part(s) being moved varies throughout the recording; there was not a specific protocol for moving, it was kept random so the developed classifier would not suffer from generalization problems.

- **Walking (W-class):** the subject is breathing normally and is walking towards and away from the radar in a straight line at approximately 0.5m/s.

With the exception of the W-class, each activity class is independent of posture and location within the room. Because of this, the relative angle and distance between the subject’s thorax and the radar varies. This was done so that the samples represent as closely as possible what would be encountered in a real environment where the subject is not following controlled test cases.

3) **Data Collection:** Data collection was performed in a simulated penitentiary cell at Carleton University (Ottawa, ON) after obtaining necessary ethics approval. The room measured 3.15x3.35x2.95m (Fig. 1) and was made of masonry units opaque to electromagnetic signals at the operating frequency of the radar, which was mounted on the wall at an angle of approximately 30° above the floor.

Fig. 1: Radar room with location markers (A, B, C) used for specifying different activity protocols.

antenna was operating and only one human was present in the radar field.

4) **Preprocessing:** The radar data was segmented and saved into one minute files. Each one consisted of 54300 samples in each of the 9 zones. The zone in which the subject was present was estimated by finding the highest energy zone (after the mean of the signal was removed). Non-overlapping data segments of ten seconds were used for processing. Of the 642 segments obtained, 67 were removed because they were either corrupted with noise, they were transitional samples (i.e. the sample was composed of data that belonged to two classes), or had been zero padded during the segmentation process. This left 575 segments for classification purposes. The samples were then filtered with a second order 0.08Hz high pass Butterworth filter (for removal of DC bias due to clutter) and a second order 20Hz low pass Butterworth filter (for removal of possible high frequency noise including 60Hz Mains vibrations).

5) **Feature Extraction:** A total of 43 features were extracted from each signal. The features included 11 time domain features and 32 frequency domain features. The time domain features included: correlation in embedding space of original signal, correlation in embedding space of signal in 0.08-0.333Hz band, correlation in embedding space of signal in 0.667-3Hz band, root mean square value, zero crossing rate, turns count, variance, skewness, kurtosis, mobility, and formfactor. Correlation was found by computing the auto-mutual information for the signal, finding the first local minimum and then calculating the correlation between the signal and the signal shifted by the number of samples corresponding to the local minimum in the auto-mutual information. The frequency domain features included: total power, mean frequency, median frequency, spectral variance, spectral skewness, fractional power in 0.2-0.667Hz band (corresponding to breathing range and second harmonic), fractional power in 0.667-3Hz band (corresponding to heart beat range), fractional power in three noise frequency bands, fractional power in 3-5Hz band, fractional power in 5-11Hz band, fractional power of signal greater than 11Hz, all 15 possible ratios of the aforementioned fractional powers, fractional power in 0.08-0.35Hz band (corresponding to range of fundamental breathing frequency), fractional power in 0.36-0.7Hz band (corresponding to range of second breathing frequency harmonic), the ratio of the fundamental and second breathing frequency harmonic, Shannon entropy of entire spectrum, Shannon entropy of breathing range, and Shannon entropy of heart beat range. The frequency domain features were extracted from the signal spectrum generated from a 216 point Welch-Periodogram. The final data matrix consisted of 576 43-dimensional vectors and was converted to a z-scores matrix by subtracting the means and dividing by the standard deviation.

III. **INTRINSIC DIMENSIONALITY**

Observations are described in terms of collections of variables/attributes, having several kinds of mutual dependencies, redundancies and noise. The developments in sensor, communication and computing technologies increase the number of descriptor variables and consequently, the dimension of the observation space. However, such an increase affects the performance of statistical and machine learning procedures because of the curse of dimensionality. Often the data concentrate in low dimension nonlinear manifolds, embedded within the high dimensional space in which the data is represented. The implication is that in fact the data is not really high dimensional. The dimension of those manifolds is considered to be the intrinsic dimension and usually it is much smaller than that of the original data representation space. Learning and uncovering these manifolds is important and useful for understanding the internal structure of the data, as well as for improving the performance of data analytic methods like clustering, classification and regression. Different approaches...
have been proposed for finding the intrinsic dimensionality and for learning the subspace.

A classical approach to estimate intrinsic dimensionality has been based on the eigenvalues obtained when performing Principal Components Analysis (PCA) [7]. In this case the dimension is determined by the number of important eigenvalues. A typical criterion is to retain those normalized eigenvalues larger than 0.025.

A maximum likelihood estimator (MLE) of the dimension is presented in [8]. The idea is to analyze small hyperspheres around each data point and consider the occurrence of points within as a Poisson process, under the assumption that the probability distribution of the points is constant within the sphere. A log-likelihood function is derived for the process from which an estimate for the dimension around a point \( x \) is obtained. This estimate is expressed in terms of the number of neighbours \( k \) as \( \hat{m}_k(x) = \left[ \frac{1}{k-2} \sum_{i=1}^{k} \log \frac{T_i(x)}{\gamma_i} \right]^{-1} \) where \( k \) is the number of neighbours considered and \( T_k(x) \) is the Euclidean distance from the point \( x \) to its \( k \)-th neighbour.

The estimation for the whole dataset is obtained by averaging \( \hat{m}_k(x) \) for all points \( x \in X = \{x_1, \cdots, x_n\} \). Since this is valid only for a certain \( k \), several estimates are produced for a range of values of \( k \in [k_1, k_2] \) and then an overall average over that range is considered as the final estimate (usual choices are \( k_1 = 6, k_2 = 12 \)):

\[
\hat{m}_k = \frac{1}{n} \sum_{i=1}^{n} \hat{m}_k(x_i), \quad \nu = \frac{1}{(k_2-k_1+1)} \sum_{k=k_1}^{k_2} \hat{m}_k
\]

Another approach is based on estimating the dimension of the attractor of a chaotic dynamical system, using the correlation integral [9]. The assumption is that the volume of the attractor of a chaotic dynamical system, using the correlation integral [9]. The idea is based on a modified version of the correlation integral using a general kernel function. The estimation is based on the correlation integral rate of the U-statistic, defined as

\[
U_{n,h}(K) = \frac{2}{n(n-1)} \sum_{1 \leq i < j \leq n} K_h(||x_i - x_j||^2)
\]

where \( K_h(||x_i - x_j||) = \frac{1}{\sqrt{h}} \mathcal{K}(||x_i - x_j||^2/h^2) \), \( \mathcal{K} \) is a measurable, non-negative, bounded kernel \( \mathcal{K} : \mathbb{R} \to \mathbb{R} \), \( h \) is a kernel parameter, \( m \) is the dimension of the submanifold contained in the high dimensional space and \( n \) the number of vectors in the sample. In particular, a simple kernel \( K(x) = (1 - x)_+ \) is used and five samples of sizes \( \{N/5, N/4, N/3, N/2, N\} \) are considered. For them empirical estimates of the U-statistic are produced for a collection of tentative dimensions \( l \in [1, l_{\text{max}}] \). Individual estimates of intrinsic dimension result from applying weighted least squares to linear fits for the obtained U-values and the tentative dimensions. The slope with the smallest absolute value is considered to represent \( \nu \) (see [13] for details).

Another approach from a dynamic systems perspective is the Takens estimator [14], given by

\[
\nu = \frac{1}{M_{h_{\text{Takens}}} \log(||x_i - x_j||/h_{\text{Takens}})}
\]

where \( \log \) is the Euclidean norm, \( h_{\text{Takens}} = \bar{d} + \sigma \), with \( \bar{d} \) and \( \sigma \) being the mean and the standard deviation of the nearest neighbor distances. \( M_{h_{\text{Takens}}} \) is the mean over all distances smaller than \( h_{\text{Takens}} \), which is a kind of maximal scale when consider neighbourhoods.

IV. MAPPING METHODS

Mapping the original space to a low-dimensional one allows a simpler representation of the data (mitigating the curse of dimensionality). If the intrinsic dimension is sufficiently low,
a direct visualization of the data structure provides appropriate insights into the data structure. This approach has been successfully applied to the analysis of sensor data [15], [16].

A. Principal Components

Principal Components Analysis [17] is one of the classical methods that have been used for dimensionality reduction. It computes a linear transformation that projects the training inputs into a variance-maximizing subspace in a way that ensures a monotonically decreasing variance with the number of components. The linear transformation maximizes the variance of the projected inputs, and is a projection matrix.

B. Sammon nonlinear mapping

The preservation of dissimilarities/differences between the original and the target spaces is expressed as minimizing different formulations of an error mapping function. Different methods are derived from this general concept, among them, Sammon’s nonlinear mapping [18]. It considers the transformation of vectors of two spaces of different dimension \((D > m)\) by means of the transformation \(\varphi : \mathbb{R}^D \rightarrow \mathbb{R}^m\) which maps vectors \(\vec{x} \in \mathbb{R}^D\) to vectors \(\vec{y} \in \mathbb{R}^m\), \(\vec{y} = \varphi(\vec{x})\) minimizing Sammon error \(E = \frac{1}{\sum_{i < j} \delta_{ij}} \sum_{i < j} \left(\delta_{ij} - d_{ij}^2(\vec{y}, \vec{y})\right)^2\), where \(\delta_{ij}\) is a dissimilarity measure for objects \(i, j\) in the original space and \(d_{ij}\) is a corresponding distance in \(\mathbb{R}^m\) (typically Euclidean) between the images of \(i, j\). The weight term \(\delta_{ij}^{-1}\) gives more importance to smaller dissimilarities.

C. Isomap

Isomap [12], [19], [20] is based on the rationale that points that are close Euclidean-wise on the manifold not necessarily are close along the underlying manifold containing the data. The procedure follows three steps: i) Construction of the neighborhood graph \(G\) that connects all points \(i, j\) based on their pairwise distances \(d_X(i, j)\) (usually in Euclidean metric). A distance threshold \(\epsilon\) is used to determine close points given a length of \(d_X(i, j)\) (the \(\epsilon - \text{Isomap}\) approach). The \(K - \text{Isomap}\) approach, works with the number of neighbors associated to each point, where a number of neighbors \(K\) is given and two points \(i, j\) are connected with an edge in \(G\) only if \(i\) is one of the \(K\)-neighbours of \(j\). ii) Geodesic distances \(d_M(i, j)\) are estimated between all pairs of points by computing their shortest path distances \(d_G(i, j)\) in \(G\). Initially, \(d_G(i, j)\) is considered \(d_X(i, j)\) only if there is an edge between \(i\) and \(j\) otherwise it set it as \(\infty\). For each \(k \in [1, N]\) where \(N\) is the number of data points, \(d_G(i, j)\) is replaced by \(\min\{d_G(i, j), d_G(i, k) + d_G(k, j)\}\) to find the shortest path distances. iii) The construction of the \(d\)-dimensional embedding is performed by applying classical multidimensional scaling to the \(d_G\) geodesic distance matrix.

D. t-Distributed Stochastic Neighbour Embedding (t-SNE)

This technique is an enhancement of SNE [21], where the mapping from higher dimensional space to lower dimensional space is based on the consideration of the similarity of conditional probabilities between datapoints. A conditional probability \(p_{j|i}\) is the probability of datapoint \(x_i\) to have \(x_j\) as a neighbor based on a Gaussian distribution \(p_{j|i} = \frac{1}{\sqrt{2\pi}\sigma_j^2} \exp\left(-\frac{||x_i - x_j||^2}{2\sigma_j^2}\right)\), where \(\sigma_j^2\) is the variance of datapoint \(x_i\) and \(k\) is a perplexity parameter related to selected local neighbors size. For the lower dimensional space, SNE utilizes conditional probabilities \(q_{ji}\) of datapoints \(x_i\) based on another Gaussian distribution. The goal is to minimize the difference between the probability distributions of the two spaces, expressed as the sum of Kullback-Leibler divergences:

\[ C = \sum_{i} \sum_{j} \left(p_{j|i} \log \frac{p_{j|i}}{q_{ji}}\right) \]

One drawback of SNE, is the low cost when representing widely separated points. t-SNE applies a symmetric cost function by considering \(p_{ij} = p_{ji}\) and \(q_{ij} = q_{ji}\) and minimizes the mismatch of a joint probability distribution \(P\) of higher dimension and \(Q\) a joint probability distribution of low dimension space as \(C = \sum_{i} \sum_{j} p_{ij} \log \frac{p_{ij}}{q_{ij}}\). The crowding problem of SNE is addressed in t-SNE by using the Student’s \(t\)-distribution in the target space, which has a heavier tail [22].

V. RESULTS

The Intrinsic Dimensionality estimations for the radar data obtained with the techniques described in Section III are shown in Table I. Their mean and median are 4.68 and 4.994 respectively and despite of their variability, they coincide in indicating that the dimension of the manifold containing the data is much smaller (≈ 5) than that of the space determined by the descriptor variables (43). Accordingly, a considerable dimensionality reduction could be possible without loosing much information. On the other hand, with an intrinsic dimension not much larger than 3, the visualizations obtained with mappings targeting 3D spaces could provide useful insight at understanding the underlying structure of the data and its relation with the class distributions.

The \(\mathbb{R}^{43} \rightarrow \mathbb{R}^3\) mappings of the radar data obtained with the methods from Section IV are shown in Fig. 2. It is impossible to represent 3-dimensional data in full accuracy on hard media. Only snapshots from selected perspectives can be used for illustrating the structure of these spaces. On the one hand, their dimension is in general smaller than the intrinsic dimension, with the inevitable information loss. On the other hand, no single perspective can reveal all of the details and the substructures that an interactive inspection tool would. All mappings applied to the radar data are unsupervised (that is, the data distribution is completely independent from the class distribution). For comparison and to aid in the interpretation, on Fig. 2 the class information was overlaid \textit{a posteriori}.

<table>
<thead>
<tr>
<th>Method</th>
<th>Intrinsic Dimension Estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eigenvalues</td>
<td>7.090</td>
</tr>
<tr>
<td>Maximum Likelihood Estimator (MLE)</td>
<td>5.988</td>
</tr>
<tr>
<td>Correlation Dimension</td>
<td>1.346</td>
</tr>
<tr>
<td>Takens</td>
<td>3.571</td>
</tr>
<tr>
<td>Geodesic Minimal Spanning Tree (GMST)</td>
<td>6.175</td>
</tr>
<tr>
<td>U-statistic</td>
<td>4</td>
</tr>
</tbody>
</table>

TABLE I: Intrinsic dimensionality estimations.
For the PCA, the estimated 7 intrinsic dimensions capture only 81.78% of the variance, whereas the space of the first 3 components captures much less (62.30%). Therefore, the structure shown in Fig. 2a is only orientative. However, it clearly shows that the W-class (red objects) is completely separable from the others by a suitable oriented plane. Accordingly, a linear classifier would resolve it if working on that reduced dimension space. This pattern appears rather clearly in the 3D space obtained with Sammon’s nonlinear mapping, shown in Fig. 2b. In this case, Euclidean distance was the dissimilarity measure used on both the original and the target spaces. The W-class (red) exhibits a bimodal character, with one mode having a much higher density. The SM-class (blue) appears as a very compact subset, with some intersection with the more scattered SS-class (green), where a higher density core can be identified. With Isomap (Fig. 2c) the three main structures appear more widespread than with Sammon, which is likely the effect of the geodesic expansion of the distances.
that Isomap follows along the subspace manifold containing the data. However, the same structures are easily identified: separability and bimodality of the W-class, compacity of the SM-class and its slight intersection with the SS-class, which keeps revealing a higher density core. However, Isomap reveals more clearly some interesting features: (i) the fact that the SM-class lies in between the W-class and (ii) the intersection with the SS-class happens with the lower density elements of the latter. Considering the experimental difficulties of keeping all subjects in a sedentary and still state of activity while conducting the radar observations, it is likely that these subjects would have exhibited some degree of movement during the recordings. The sharp division of the activities into discrete classes does not capture such borderline situations. The t-SNE mapping concentrates the W-class, while preserving its bimodality and provides more a detail view of the relationships between the SM and the SS classes, which is directly related to the physics behind the different levels of activity. In the latter case, the space resolves in a more clear fashion the higher and the lower density subareas, as two modes within the class.

Altogether, these results provide useful insights into the internal structure of the CW Doppler radar data and its relationship with physiological and behavioral states in humans. They also provide details that would help in the design of appropriate recognition or classification systems.

VI. CONCLUSIONS

Non-contact physiological monitoring in real life applications requires a system to be able to process signals resulting from subjects performing many every day activities that have different levels of movement. The level of activity of the subject affects the radar signal greatly. Single channel CW Doppler radar measurements were preprocessed and described by 43 feature vectors associated to subjects in various positions, distances, orientations and with various levels of activity.

Different estimates of its intrinsic dimension show that the data is contained in an embedded submanifold of much lower dimension (approximately 5), indicating that there is a high level of redundancy in the information provided by the descriptor variables. Four unsupervised mapping techniques, linear and nonlinear, based on different principles, were used for constructing three dimensional mappings. A posteriori, class information was overlaid on these spaces for comparing with the structure provided solely by the descriptor variables. The integrated analysis of the patterns emerging from the different unsupervised mappings reveals well defined regions in good correspondence with the different levels of activity for the subjects in the study. Altogether, these results provide useful insights into the internal structure of the CW Doppler radar data and its relationship with physiological and behavioral states in humans. They would help in the design of appropriate recognition systems with supervised approaches.

ACKNOWLEDGMENT

The authors would like to thank Correctional Service Canada for funding this project. The project is additionally funded through NSERC Discovery and OGS grants.

REFERENCES

Sequential Minimal Optimization Extended to General Quadratic Programming

1st William Brendel  
Snap Research  
Snap Inc.  
Los Angeles, USA  
william.brendel@snap.com

2nd Luis Marujo  
Snap Research  
Snap Inc.  
Los Angeles, USA  
luis.marujo@snap.com

Abstract— Nearly two decades ago Platt introduced the sequential minimal optimization (SMO) algorithm [1] to solve the Support Vector Machine (SVM) dual quadratic programming optimization problem. SMO belongs to the family of Sequential Quadratic Programming (SQP) algorithms, and specifically aims to reduce the quadratic programming (QP) problem to its minimum at every iteration. As a result, SQP can be solved analytically and leads to an algorithm with linear time and space complexity. In 2005, Fan et al. [2] summarized most of the optimization strategies that can be applied to solve the SVM QP problem with SMO in the well known LIBSVM library. Presently, other QP problems with similar form as the SVM QP dual problem are solved using more time and space consuming algorithms than mobile computational requirements allow. This research strives to discern the conditions that allow SMO to be extended to other QP problems and its complexity of solving the minimal QP at each iteration.

Index Terms—Quadratic optimization, Support vector machines, Natural language processing.

I. INTRODUCTION

It is not surprising that problems involving quadratic programming (QP) occur frequently in the fields of Machine Learning, Natural Language Processing, Computer Graphics, and Computer Vision. Many applications are therefore formulated as graph theory problems, where typical objective functions to be optimized contain unary potentials related to nodes and binary potentials related to edges. For the last 50 years, graph theory has simply supported an ocean of computer science applications [3], [4]. To name a few examples, image segmentation as been formulated as minimum cut [5], [6], maximum weight independent set [7], maximum weight clique [8], [9] and minimum spanning tree [10] problems [11]. Multi-object tracking has been formulated as maximum weight independent set [12] and generalized minimum and maximum clique [13], [14] problems. In addition, SVM and Support Vector Clustering (SVC) formulation [15] can be viewed as an independent set problem [16], [17]. That said not all QP problem are directly related to graph theory, for example [18]–[22]. In the same way that graph problems can be reduced to one another, QP problems can be reformulated into problems that can be solved more efficiently. For example, maximum clique, maximum independent set and graph cut problems are linked to the vertex separator problem QP formulation [17]. Various cut problems are reformulated as spectral clustering problems that can be optimized via weighted kernel k-means algorithms [6], [23], achieving real-time computation performances. In Tsang et al.’s work [16], SVM and SVC are formulated as minimum enclosing ball problems, obtaining provably approximate optimal solutions in linear time, with a space complexity independent of the problem size. Since graph-theory problems (and their QP formulation) can be reduced to one another, many QP problems are re-formulated to optimize a simple quadratic objective function with linear constraints such as Eqn. (2). Consequently, it is natural to investigate the extent to which SMO can be generalized to solve such QP problems while considering high-speed and low-memory requirements typically found on mobile platforms.

A. Related Work to Sequential Minimal Optimization

SMO was originally designed to train a support vector machine that requires the solution of the very large SVM dual QP optimization problem [1] defined below:

$$\alpha^* \leftarrow \arg \max_{\alpha \in [0, C]^n} F(\alpha) : 1^T\alpha - \frac{1}{2}\alpha^T H \alpha$$

s.t. \(y^i\alpha = 0\) with \(y \in \{-1, 1\}^n\)

$$H_{ij} = y_i y_j K(x_i, x_j) \quad \text{and} \quad x^k \in \mathbb{R}^d$$

SMO belongs to the family of SQP algorithms [24]. It breaks a large QP problem into a series of smaller QP problems, each of which optimizes a quadratic model of the objective subject to a linearization of the constraints. The method is equivalent to applying Newton’s method to the KarushKuhnTucker (KKT) conditions of the QP problem. In SMO, the small QP problems involve only two variables. These small QP problems are solved analytically, thus avoiding the use of a time-consuming numerical QP optimization at each iteration. The amount of memory required for SMO is linear in the training set size \(n\), which allows SMO to handle very large input sets.

SMO is not the only algorithm reducing the large SVM QP problem into a series of smaller problems [25], [26]. However, methods of [25], [26] are not scalable for very large training set as the small QP problems cannot fit into memory [1], which is still a problem nowadays, for example as the scientific community moved research tasks from MNIST to IMAGENET. A more radical approach than SMO is to avoid the QP altogether [27]. However, non-linear kernels still...
require the inversion of an \( n \times n \) matrix. SMO is also closely related to a family of optimization algorithms called Bregman methods [28] or row-action methods [29]. These methods solve convex programming problems with linear constraints. They are iterative methods where each step projects the solution at each iteration onto each constraint. However, as stated in [1], unmodified Bregman methods cannot solve general non-convex QP problems with linear equality constraints directly. Other algorithms with analytical solutions for the small QP problems, like in [8], [30], could also be extended to solve the SVM QP formulation or more generally Eqn. (2). However, SMO presents a unique advantage: the direction in which the solution is updated at each iteration is extremely sparse (involving only two variables). Consequently, SMO reduces the time complexity of the overall process from \( O(n^2) \) for standards algorithms to \( O(n) \) while maintaining a linear space complexity. When the final solution is assumed to be sparse, SMO only focuses on optimizing the few dimensions that are not null. However, when the final solution is known to be dense, it optimizes only the best two dimensions at each iteration, leading to a sub-optimum solution. While we can always use other SQP algorithms [24], we offer in this paper a solution to escape poor local optimum. An extensive study on SMO applied to SVM can be found in [31]. We summarize the main contributions of the paper below.

B. Contributions

In the context of SVM, SMO was already extended, for example, to learn jointly support vectors and kernels [32]. This paper aims at extending SMO to the general form of QP described in Eqn. (2). More precisely: (1) We prove that the complexity of the smallest possible problem to be optimized at each iteration depends on the number of independent linear equality constraints. (2) We prove that optimizing the small problem at each iteration is equivalent to analytically solving a linear system and a bounded D-dimensional conic equation. (3) We extend the notion of a working set [2], [26] in order to escape local sub-optimal solutions. (4) We extend the notion of dimension selection [2], [30] to the general form of QP in Eqn. (2). (5) We show that SMO can be applied to QP problems other than the dual formulation of SVM and SVC, like document summarization. Below, we conclude our introduction with the set of notations and definitions used throughout the paper. For the remaining parts of the paper, we detail our contributions, focus on selected applications, demonstrate the results of our experiments, and finally present the conclusions.

C. Notations and definitions

Vectors and matrices are noted in bold, and matrices are in capital letters. \( \mathbf{x} \) and \( \mathbf{C}_{ij} \) are values at position \( i \) and \( j \) in vector \( \mathbf{x} \) and matrix \( \mathbf{C} \). \( \mathbf{C}_{ij} \) is the \( j^{th} \) column of \( \mathbf{C} \), \( \mathbf{C}^T \) is the \( j^{th} \) row of \( \mathbf{C} \). \( \mathcal{F} \) is an objective function to be optimized with respect to \( \mathbf{x} \), \( \mathbf{x}^{(t)} \) represents the solution at iteration \( t \) and \( \mathbf{x}^* \) the final solution when the KKT conditions are met. In the remaining part of the paper we simplify the notation: \( \mathcal{F} = \mathcal{F}(\mathbf{x}), \mathcal{F}^{(t)} = \mathcal{F}(\mathbf{x}^{(t)}), \mathcal{F}^* = \mathcal{F}(\mathbf{x}^*), \delta \mathcal{F}^{(t)} = \delta \mathcal{F}^{(t)}(\delta \mathbf{x}) \) and \( \delta \mathbf{x} = \mathbf{x}^{(t+1)} - \mathbf{x}^{(t)} \). We denote by \( \nabla \mathcal{F} \) the gradient of \( \mathcal{F} \) with respect to \( \mathbf{x} \) and by \( \delta \mathcal{F}^{(t)} = \mathcal{F}^{(t+1)} - \mathcal{F}^{(t)} = \mathcal{F}(\mathbf{x}^{(t+1)}) - \mathcal{F}(\mathbf{x}^{(t)}) \) the objective function of the small QP problem at iteration \( t \). In the following, \( \mathbf{x}, \mathbf{x}^{(t)}, \mathbf{x}^* \) and \( \delta \mathbf{x} \in \mathbb{R}^n \). \( \mathbf{D}_t = \text{diag}(\mathbf{x}) \) is the diagonal matrix where \( (\mathbf{D}_t)_{ii} = x_i \). We define \( \mathcal{S} = \{i_k\}_{k=1}^K \) as a set of \( K \) columns of \( \mathbf{C} \), with \( \mathcal{C}_{ik} = \mathbf{C}_{i_k} \). Respectively \( \mathcal{C}_{SS} = (\mathcal{C}_{S}^T)_{S}^T \in \mathbb{R}^{K \times K} \) and \( \mathbf{x}_S = [x_{i_1}, \ldots, x_{i_K}]^T \in \mathbb{R}^K \).

Definition 1. We call \( \text{Row}(\mathcal{C}) \) the vector space spanned by the rows of \( \mathcal{C} \), and \( \text{rank}(\mathcal{C}) = |\text{Row}(\mathcal{C})| \) is the dimension \( \text{Row}(\mathcal{C}) \), i.e. the number of independent rows of \( \mathcal{C} \).

Definition 2. We call \( \text{Ker}(\mathcal{C}) = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{C}\mathbf{x} = \mathbf{0}\} \) the null space of \( \mathcal{C} \), and \( \text{nulity}(\mathcal{C}) = |\text{Ker}(\mathcal{C})| \). Then \( n = \text{rank}(\mathcal{C}) + \text{nulity}(\mathcal{C}) \).

II. Problem Setup

We aim to generalize SMO to optimize the QP problem:

\[
\mathbf{x}^* \leftarrow \text{optimize } \mathcal{F} : \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} \text{ s.t. } \mathbf{C} \mathbf{x} = \mathbf{d}
\]

(2)

where \( \mathbf{C} \in \mathbb{R}^{m \times n} \) and \( \mathbf{C} \mathbf{x} = \mathbf{d} \) forms a “non-over-constrained” system. Inequality constraints of the form \( \mathbf{C} \mathbf{x} \leq \mathbf{d} \) can be directly added to the objective function \( \mathcal{F} \) via Lagrangian multipliers [33], solving a dual QP problem of the same form as Eqn. (2). The problem can also be reformulated using slack variables, leaving only equality constraints [33]. The iterative formulation of Equn. (2) aims at solving:

\[
\delta \mathbf{x}^* \leftarrow \text{optimize } \delta \mathcal{F}^{(t)} : \delta \mathbf{x}^T \mathbf{A} \delta \mathbf{x} + \nabla \mathcal{F}^{(t)} \delta \mathbf{x} \text{ s.t. } \mathbf{C} \delta \mathbf{x} = \mathbf{0}, \mathbf{x}^{(t)} + \delta \mathbf{x} \in \mathbb{R}^n \text{ and } \nabla \mathcal{F}^{(t)} = (\mathbf{A} + \mathbf{A}^T) \mathbf{x}^{(t)} + \mathbf{b}
\]

(3)

where we assume that \( \mathbf{x}^{(0)} \) is initialized so that \( \mathbf{C} \mathbf{x}^{(0)} = \mathbf{d} \), and \( \mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} + \delta \mathbf{x}^* \). The goal of SMO is to constrain \( \delta \mathbf{x} \) to be extremely sparse so that Eqn. (3) can be solved analytically. While traditional formulation would control the sparsity of \( \delta \mathbf{x} \) by incorporating an additional constraint like \( ||\delta \mathbf{x}||_1 < \nu \), SMO explicitly controls the sparsity of \( \delta \mathbf{x} \) via a set \( \mathcal{S} \) of \( K \) indices:

\[
\delta \mathbf{x}_S^* \leftarrow \text{optimize } \delta \mathcal{F}^{(t)} : \delta \mathbf{x}_S^T \mathbf{A}_{SS} \delta \mathbf{x}_S + \nabla \mathcal{F}^{(t)}^T \delta \mathbf{x}_S \text{ s.t. } \mathbf{C}_S \delta \mathbf{x}_S = \mathbf{0}, \mathbf{x}_S^{(t)} + \delta \mathbf{x}_S \in \mathbb{R}^n \text{ and } \nabla \mathcal{F}^{(t)} = (\mathbf{A} + \mathbf{A}^T) \mathbf{x}_S^{(t)} + \mathbf{b}_S
\]

(4)

In [1], [2], the later system is first optimized with respect to \( \mathcal{S} \), and then with respect to \( \delta \mathbf{x} \). The update rule is then \( \forall i_k \in \mathcal{S}^*, \mathbf{x}_S^{(t+1)} \leftarrow \mathbf{x}_S^{(t)} + \delta \mathbf{x}_S^* \). In summary, designing a SMO algorithm involves 3 steps: we need (i) to fix the number of indices \( K = |\mathcal{S}| \), (ii) to design a heuristic to select the best set of indices \( \mathcal{S}^* \), and (iii) to solve Eqn. (4) analytically with respect to \( \delta \mathbf{x}_S^* \) once \( \mathcal{S}^* \) is found.
III. FINDING THE MINIMAL QP PROBLEM SIZE

The minimal QP size is directly related to \( K \), the number of indices in \( S \). In the context of solving the SVM dual Eqn. (1), [1] fixed \( K = 2 \). If we add a sparsity constraint \( \Gamma \alpha = \nu \) to Eqn. (1) to have fewer support vectors, the linear constraints are summarized by the system \( C \mathbf{x} = d \) with \( C = [y, 1]^T = \begin{bmatrix} -1 & -1 & \cdots & -1 \end{bmatrix} \) and \( d = [0, \nu]^T \). However, if we maintain \( K = 2 \) and if we inadverently pick the index set \( S \) such that \( C_S = \begin{bmatrix} -1 & -1 \end{bmatrix} \) then \( C_S \mathbf{x}_S = 0 \Rightarrow \mathbf{x}_S = 0 \), since the columns of \( C \) are orthogonal, hence independent. This means the iterative system becomes stationary and the solution may never converge to an optimum. Previous research [2] solved the issue by selecting \( S = \{i_1, i_2\} \) so that \( y_{i_1} = y_{i_2} \), and extended the selection of \( K \) to any matrix \( C \).

**Lemma 1.** Let \( K \) be the number indices in \( S \) needed to optimize Eqn. (4), \( \text{rank}(C) \leq n - 1 \Rightarrow K \geq \text{rank}(C) + 1 \).

**Proof.** \( K \leq n \), hence \( \text{rank}(C_S) \leq \text{rank}(C) \) as we may have removed some of the independent columns of \( C \) during the projection onto the dimensions selected by \( S \). This leads to \( \text{rank}(C_S) \leq \text{rank}(C) \leq K - 1 \). Based on Def. 2, we have \( C_S \in \mathbb{R}^{m \times K} \Rightarrow \text{nullity}(C_S) + \text{rank}(C_S) \). Hence nullity\( (C_S) = K - \text{rank}(C_S) \geq \text{rank}(C_S) + 1 - \text{rank}(C_S) \) leading to nullity\( (C_S) \geq 1 \). This means that \( \forall S \) with \( |S| = K \), \( \text{nullity}(C_S) \leq |S| - 1 \) and \( \text{nullity}(C_S) \geq |S| - |S| \), i.e., the system Eqn. (4) is not stationary until it reaches an optimum.

Note that when \( \text{rank}(C) = n \), we can directly solve \( \mathbf{x}^* = C^{-1} \mathbf{d} \) if \( \forall \mathbf{v} \in \{1, n\}, \mathbf{x}_i^* \in [r, R] \) (otherwise no valid solution exists). When \( C \) is full rank, we can get \( C^{-1} \) analytically by using the Jordan-Gauss elimination method. In addition, we show in the supplemental material that Eqn. (4) can be formulated so that \( K \geq 1 \) by parameterizing \( \delta \mathbf{x} \) as a linear combination of vectors spanning \( \text{Ker}(C) \). However, it is computationally expensive when \( |\text{Ker}(C)| \) is large.

IV. DIMENSION SELECTION AND WORKING SET

Once we have chosen \( K \), we need to find the set \( S = \{i_k\}_{k=1}^K \) before optimizing Eqn. (4) with respect to \( \delta \mathbf{x}_S \).

[2], [26] introduced the notion of “working set” related to the most violating pairs and second order information. Both are linked to the first and second order approximation of the objective function \( F \). [30] also chooses the dimensions to be optimized based on the highest gradient magnitude along a specific dimension. We propose the following heuristic to extend the notion of working set. At each iteration \( t \), we maintain two sets, an active set \( S_A^{(t)} \) and an inactive set \( S_I^{(t)} \). Let \( \nabla F_i^{(t)} \) be the gradient of the objective function at time \( t \), and \( \eta_i^* \) be the best potential displacement along the \( i \)th direction of the gradient:

\[
\eta_i^* \leftarrow \text{optimize } \eta_i \nabla F_i^{(t)} \Leftrightarrow w^* \leftarrow \text{optimize } (w - x_i^{(t)}) \nabla F_i^{(t)} \quad \text{with } \eta_i^* = w^* - x_i^{(t)} \quad \text{and } \nabla F_i^{(t)} = ((A + A^T)x_i^{(t)} + b) \quad (5)
\]

In other words, when the QP is maximized, we pick \( \eta_i^* \) such that \( \eta_i^* \nabla F_i^{(t)} \geq 0 \), and when the QP is minimized we pick \( \eta_i^* \) such that \( \eta_i^* \nabla F_i^{(t)} \leq 0 \). Then we define \( S_A^{(t)} = \{i_k\} \) such that \( \eta_i^* \nabla F_i^{(t)} \neq 0 \) and \( S_I^{(t)} = \{j_1\} \) such that \( \eta_i \nabla F_j^{(t)} = 0 \). The first \( K - 1 \) indices of \( S \) are chosen among \( S_A^{(t)} \) and the last index is chosen among \( S_A^{(t)} \cup S_I^{(t)} \). When \( |S_A^{(t)}| \leq K - 1 \), all the indices of \( S_A^{(t)} \) are part of \( S \) and the remaining \( K - 1 - |S_A^{(t)}| \) indices are picked randomly from \( S_I^{(t)} \). When \( |S_A^{(t)}| > K - 1 \), we pick the best \( K - 1 - \rho \) indices \( i_k \in S_A^{(t)} \) corresponding to the largest \( |\eta_i^* | \nabla F_i^{(t)} | \). The other \( \rho \) indices are picked randomly in \( S_A^{(t)} \). While randomizing a portion of the indices for \( S_A^{(t)} \) may slow down the convergence rate a little bit, it prevents falling into poor local optimum. Typically we choose \( \rho \ll |S_A^{(t)}| \). Note that when \( x^{(t)} \) is sparse and when \( x^* \) is expected to be sparse, \( |S_A^{(t)}| \) decreases quickly, reducing the search space for \( S \) dramatically. For the last index \( i_K \), we run a search over all the unpicked indices and retain the one that offers the best optimizer \( \delta \mathbf{x}_{S_{A_{i_{<K}}}} \) of Eqn. (4). Our heuristic generalizes [2], [30] as it not only introduces a random selection process in order to avoid poor local optimum, but also takes into account the potential placement \( \eta^* \) along each gradient direction.

\[ \delta \mathbf{x} = U \mathbf{c} \quad \text{such that } C \delta \mathbf{x} = 0 \Leftrightarrow \delta \mathbf{x} \in \text{Ker}(C) \]

Then Eqn. (3) can also be reformulated to avoid the working set selection process, leading to the next lemma:

**Lemma 2.** Eqn. (3) can be reformulated in the null space of the constraints so that \( K \geq 1 \).

**Proof.** Let \( |\text{Ker}(C)| = D \) and \( U \in \mathbb{R}^{m \times D} \) such that the columns \( \{u_i\}_{i=1}^D \) of \( U \) span \( \text{Ker}(C) \). We can parameterize \( \delta \mathbf{x} = U \mathbf{c} \) since \( C \delta \mathbf{x} = 0 \Leftrightarrow \delta \mathbf{x} \in \text{Ker}(C) \). Then Eqn. (3) becomes:

\[
\alpha^* \leftarrow \text{optimize } \alpha^* U^T A U \alpha + (A x_i^{(t)} + A^T x_i^{(t)} + b)^T U \alpha \quad \text{such that } r \leq x_i^{(t)} + (U \alpha)_i \leq R \quad \forall i \in \{1, n\} \quad (6)
\]
Indeed $\forall \alpha \in IR^D$, $CU\alpha = 0$, i.e. $CU = 0$ by construction. From Lemma 1 $CU = 0 \Leftrightarrow \text{rank}(CU) = 0 \Rightarrow K \geq 1$. □

The downside of this reformulation is that we need to pay the space and computation price of processing $U$ and $U^TAU$, unless $A$ is extremely sparse or $D$ extremely low. One clear advantage is when $D = |Ker(C)| \leq 2$, optimizing Eqn. (6) is equivalent to optimizing a bounded $D$-dimensional conic equation, as demonstrated by Eqn. (13) and Eqn. (15) that have closed form solutions (see next two sections).

V. SOLVING THE MINIMAL QP PROBLEM

Many applications (min/max clique, cut, independent set, etc) have constraints with rank($C$) ≤ 1. Therefore, in the following we will show how to solve the small QP problem of Eqn. (4) for special cases where rank($C$) = 0 and for the general case of rank($C$) ≥ 2. In the future, we will set $K = \text{rank}(C) + 1$.

A. Special Case for rank($C$) = 0, $K = 1$

rank($C$) = 0 means the linear constraint $CX = d$ is removed from Eqn. (2). Since $K = 1$, $S = \{k\}$ and $\delta x_{S} = \alpha \in IR$. At each iteration $t$ Eqn. (4) becomes:

$$(k^{*}, \alpha^{*}) \leftarrow \text{optimize } \alpha^2 A_{kk} + \alpha (Ax^{(t)} + A^T x^{(t)} + b)_k \quad \text{s.t. } x^{(t)}_k + \alpha \in [r, R] \quad (7)$$

We solve Eqn. (7) with respect to $\alpha$ for every index $k \in \{1, n\}$. For each fixed $k$, Eqn. (7) reduces to optimizing a bounded second degree equation of the form $\alpha^* \leftarrow \text{optimize } \beta \alpha^2 + \gamma \alpha$, where alpha’s bounds are defined as $\alpha_{\min} = r - x^{(t)}_k$ and $\alpha_{\max} = R - x^{(t)}_k$. The next section gives the closed form solution for $\alpha^*$. At iteration $t$, the complexity of solving Eqn. (7) is linear.

B. Special Case for rank($C$) = 1, $K = 2$

With rank($C$) = 1 the linear constraints in Eqn. (2) reduces to $c^T x = d$, with $c$ and $d \in IR^n$, $S = \{k, l\}$ and $\delta x_{S} = [\alpha_k, \alpha_l]^T \in IR^2$. As described in Section IV, we first conduct a linear search to find $l$, then we perform a second linear search to find $k, \alpha_k$ and $\alpha_l$. In order to find $k$ we search the dimension that would lead to highest potential gradient magnitude along the dimension $k$, taking into account the bounds on $x_k \in [r, R]$:

$$k^* \leftarrow \arg \max_{k \in S^{(1)}_A} \left\| (w - x_k) (Ax + A^T x + b)_k \right\| \quad (8)$$

With a high probability we keep $k^*$, and with a low probability we simply randomly pick $k \in S^{(1)}_A$ such that $|\eta \nabla F(x)_k| \neq 0$. We then perform a second linear search to find the second dimension $l$ and the coefficients $[\alpha_k, \alpha_l]$

$$(l^*, \alpha_k^*, \alpha_l^*) \leftarrow \text{optimize } \alpha_k^2 A_{kk} + \alpha_l^2 A_{ll} + \alpha_k \alpha_l (A_{kl} + A_{lk}) + \alpha_k (Ax^{(t)} + A^T x^{(t)} + b)_k + \alpha_l (Ax^{(t)} + A^T x^{(t)} + b)_l \quad \text{s.t. } \alpha_k c_k + \alpha_l c_l = 0, \text{ with } x^{(t)}_k + \alpha_k \text{ and } x^{(t)}_l + \alpha_l \in [r, R] \quad (9)$$

First let’s assume both $c_k, l \neq 0$. Solving the constraints gives:

$$\alpha_k = -\frac{\alpha_l c_l}{c_k} \text{ and } \alpha_{\min} \leq \alpha \leq \alpha_{\max}$$

where $\alpha_{\min} = \max \left( r - x^{(t)}_k, \frac{(x^{(t)}_k - w)_c}{c_k} \right)$ and $\alpha_{\max} = \min \left( R - x^{(t)}_l, \frac{(x^{(t)}_l - w)_c}{c_l} \right)$ where $(w, W) = (r, R)$ if $c_k c_l > 0$ or $(w, W) = (R, r)$ if $c_k c_l < 0$. Then, solving Eqn. (9) reduces to optimizing a bounded second degree equation of the form:

$$(l^*, \alpha_k^*, \alpha_l^*) \leftarrow \text{optimize } \beta \alpha_l^2 + \gamma \alpha_l \quad \text{s.t. } \alpha_l \in [\alpha_{\min}, \alpha_{\max}] \text{ and } \alpha_k^* = -\frac{\alpha_l c_l}{c_k} \quad (11)$$

Now let us assume that only one of the coefficients ($c_k, c_l$) is null. We deduce from the linear constraint that $c_k = 0 \Rightarrow \alpha_l = 0$ and $c_l = 0 \Rightarrow \alpha_k = 0$. In both cases Eqn. (9) reduces again to optimizing a bounded second degree equation. Finally when $[c_k, c_l]^T = 0$ Eqn. (9) reduces to optimizing a bounded conic equation. The closed form solutions for both the bounded second degree and the bounded conic equations are shown in Section VI.

C. General Case for rank($C$) ≥ 2, $K = \text{rank}(C) + 1$

As described in Section IV, our algorithm first estimates the $K$–1 indices of $S$ and next conducts a line search on the remaining dimension $i_K$ while optimizing Eqn. (4) with respect to $\delta x_{S}$. As a result, for every candidate index $i_K$, we simply need to optimize Eqn. (4) with respect to $\delta x_{S}$. This ensures that both $i^*_K$ and $\delta x^*_S$ will be chosen as the best optimizer of Eqn. (4) at the end.

Lemma 3. If $K = \text{rank}(C) + 1$ and rank($C_S$) = $K$ then the complexity of optimizing Eqn. (4) with respect to $\delta x_{S}$ is reduced to the complexity of (i) solving the linear system $C_S \delta x_{S} = 0$ and (ii) optimizing a $D$-dimensional bounded conic equation with $D = \text{nullity}(C_S) = \text{rank}(C) - K + 1$.

Proof. $C_S \in IR^{m \times K}$ hence from Def. 2 nullity($C_S$) = $K - \text{rank}(C_S) = \text{rank}(C) - K + 1$. Solving $C_S \delta x_{S} = 0$ is equivalent to finding the vector space $[u_1, \ldots, u_D]$ spanning $Ker(C_S)$ with $D = \text{nullity}(C_S)$. We can then parametrize $\delta x_{S} = \sum_{k=1}^{D} \alpha_k u_k$ and solve Eqn. (4) with respect to $\alpha = [\alpha_1, \ldots, \alpha_D] \in IR^D$

$$(\alpha^*, \alpha_1^*, \ldots, \alpha_D^*) \leftarrow \text{optimize } \sum_{k=1}^{D} \alpha_k^2 A_{ik_{1}} + \sum_{k=1, k_{1}=1}^{D} \alpha_k \alpha_{k_{1}} A_{ik_{1}} + \sum_{k=1}^{D} \alpha_k \nabla F^{(t)}_{i_k} \quad \text{s.t. } \alpha_k \in [\alpha_{\min}, \alpha_{\max}] \text{ and } \nabla F^{(t)} = (Ax^{(t)} + A^T x^{(t)} + b) \quad (12)$$

Eqn. (12) is a $D$-dimensional bounded conic equation. □

For $D = 1, 2$ we provide a closed form solution in Section VI. When $D > 2$ we need to optimize iteratively the $D$–dimensional bounded conic equation, i.e. we need to solve a sub-QP problem of dimension $D$. Note that we can always randomly pick two linear combinations of vectors.
in \( \text{Ker}(C_S) \) and solve a bounded 2D conic equation, which is the equivalent of one optimization step in the random projection algorithm in [34]. Finally finding the vector space spanning \( \text{Ker}(C_S) \) is done analytically via QR decomposition or SVD of \( C_S \). Optimizing the \( D \)-dimensional bounded conic equation can be achieved with time complexity \( O(D) \) where the time complexity for the QR decomposition and Gram-Schmidt basis completion process is \( O(m^3 + m(K - m)^2) \) or for the SVD is \( O(m^3K + mK^2 + K^3) \). Solving the small QP problem is summarized in Alg. 2 and our final SMO algorithm is described in Alg. 3.

**Algorithm 2** Solve the small QP problem.

1: `function SOLVEQ2DQP(x, \nabla F(x), A, C, S)`
2: \[ \text{Find } \{u_k\} \in \text{Ker}(C_S) \]  \( \triangleright \) Using for example QR on \( C_S \)
3: \[ \text{Parametrize } \delta x_S = \sum u_k \alpha_{Sk} \]
4: \[ \text{Solve Eqn. (4) w.r.t. the } \{\alpha_{Sk}\} \]
5: \[ \text{return } (\delta x_S^*) \]

Note that when \( \text{rank}(C) \geq n - 2 \), the sub-QP problem can be solved analytically. More generally when \( \text{rank}(C_S) \geq K - 2 \), i.e. when \( \text{rank}(C) \leq \text{rank}(C_S) + 1 \), the bounded conic (or second degree) equation has a closed form solution described in Section VI. For \( D > 2 \), iterative approaches to optimize the \( D \)-dimensional bounded conic equation are more efficient than closed form solutions, and one can simply reuse the same SMO algorithm to solve the sub-QP problem.

VI. Optimizing Bounded Second Degree and Bounded 2D Conic Equations

The bounded second degree equation from the previous section has the following form:

\[
\alpha^* \leftarrow \text{optimize } \beta \alpha^2 + \gamma \alpha \quad \text{s.t. } \alpha \in [\alpha_{\text{min}}, \alpha_{\text{max}}] \quad (13)
\]

where \( \alpha^* \) has a closed form solution depending on the sign of \( \beta \) and whether the optimization problem Eqn. (2) is a maximization or minimization problem. When solving the maximization problem \( x^* \leftarrow \text{arg max } F(x) \), \( \alpha^* \) has the closed form solution:

\[
\alpha^* \leftarrow \begin{cases} \frac{-\frac{\gamma}{2\beta}} & \text{if } -\frac{\gamma}{2\beta} \in [\alpha_{\text{min}}, \alpha_{\text{max}}] \text{ and } \beta < 0 \\ \text{arg max } \beta \alpha^2 + \gamma \alpha & \text{otherwise} \end{cases}
\]

(14)

When solving the minimization \( x^* \leftarrow \text{arg min } F(x) \), \( \alpha^* \) has the same closed form solution as Eqn. (14), except that \( \text{arg max} \) becomes \( \text{arg min} \) and \( \beta < 0 \) becomes \( \beta > 0 \). In some cases, for \( \text{rank}(C_S) = K - 2 \), the QP minimal problem cannot be reduced to a bounded second degree equation, but instead to a bounded conic equation of the form:

\[
\alpha^* = [\alpha_1^*, \alpha_2^*]^T \leftarrow \text{optimize } \frac{\beta_1 \alpha_1^2 + \beta_2 \alpha_2^2 + (\beta_{12} + \beta_{21})\alpha_1\alpha_2 + 2\gamma \alpha_1 + \gamma_2 \alpha_2}{\alpha_1 \alpha_2} \quad (15)
\]

s.t. \( \alpha_1 \in [\alpha_{\text{min}1}, \alpha_{\text{max1}}] \) and \( \alpha_2 \in [\alpha_{\text{min}2}, \alpha_{\text{max2}}] \)

In order to solve Eqn. (15), we first compute the optimum on the boundaries, i.e. we fix \( \alpha_1 = \alpha_{\text{min}1} \) and optimize the second degree equation with respect to \( \alpha_2 \). We then repeat the procedure by fixing \( \alpha_1 = \alpha_{\text{max}1} \), \( \alpha_2 = \alpha_{\text{min}2} \) and \( \alpha_2 = \alpha_{\text{max}2} \). Finally, if \( (\beta_{12} + \beta_{21})^2 - 4\beta_{11}\beta_{22} < 0 \) we check if there’s an optimum inside the bounds by solving the \( 2 \times 2 \) linear system: \( 2\beta \alpha = -\gamma \). Note that \( (\beta_{12} + \beta_{21})^2 - 4\beta_{11}\beta_{22} < 0 \Leftrightarrow 0 \leq \beta_{12} - \beta_{21} < \text{det} (\beta) \), which guarantees \( \beta \) is invertible. Our final SMO algorithm is described in Alg. 3.

**Algorithm 3** General SMO (G-SMO)

1: `Initialize:

2: \( K = \text{rank}(C) + 1 \)
3: \( x^{(0)} \) such that \( Cx^{(0)} = d \) and \( \forall i \in \{1, n\}, x_i^{(0)} \in [r, R] \)
4: \( \nabla F(x) \leftarrow (A + A^T)x^{(0)} + b \)

5: `Main loop:

6: for \( t \leftarrow 1 \) to \( t_{\text{max}} \) do

7: \( S \leftarrow \text{INITWORKINGSET}(x^{(t)}, \nabla F(x), \rho) \)
8: \( \delta F^* \leftarrow 0 \)
9: for \( j \leftarrow 1, n \) \( \setminus S \) do  \( \triangleright \) Search for the last index of \( S \)
10: \( S \leftarrow S \cup j \)
11: \( \delta x_S^* \leftarrow \text{SOLVEQ2DQP}(x^{(t)}, \nabla F(x), A, C, S) \)
12: \( \delta F \leftarrow \| \delta x_S^* A_S \delta x_S + \nabla F(x) \|_2 \)
13: if \( \delta F > \delta F^* \) then \( j^*, \delta F^*, \delta x_S^* \leftarrow j, \delta F, \delta x_S^* \)
14: \( S \leftarrow S \setminus j \)
15: \( S^* \leftarrow S \cup j^* \)
16: \( x^{(t+1)} \leftarrow x^{(t)} + \delta x_S^* \)
17: \( \nabla F(x^{(t+1)}) \leftarrow \nabla F(x^{(t)} + (A + A^T + \text{diag}(b))S^*) \delta x_S^* \)
18: if \( \delta F^* \in \epsilon \) then break
19: return \( x^{(t+1)} \)

VII. Applications and Results

In this section, we apply our algorithm to several problems. For all experiments, the code was written in C++ with SSE optimization and ran on an 2.8 GHz Intel Core i7, with 16 GB 1600 MHz DDR3 RAM.

A. Single Doc. Summarization as a Sparse Max Cut

In order to generate summaries of a single document with the most \( N \) relevant sentences, we model the problem as a sparse maximum cut problem [35], [36]:

\[
x^* \leftarrow \text{arg max } x \quad F(x) = (1 - x)^T(A - D)x \\
\text{s.t. } 1^T x = N \text{ and } x \in [0, 1]^n \quad (16)
\]

where \( A_{ij} \) is the similarity between sentences \( i \) and \( j \), with \( |A_{ij}| \leq 1 \). \( D \) is a diagonal matrix such that \( D_{ii} = \max_j A_{ij} \). Hager et al. [37] proved that introducing \( D \) yields to a proper binary solution \( x \) for the aforementioned maximum cut problem. Also note that when \( D_{ii} = \sum_j A_{ij} \) the QP problem becomes equivalent to solving the minimum dominant set formulation of [38]. In the following we will report results for \( D_{ii} = \sum_j A_{ij} \) (MCUT), for \( D_{ii} = \sum_j A_{ij} \) (MDOM) and for \( D_{ii} = 1 \) (CONST1). We used the following similarity metric between sentences:

\[
A_{ij} = \sum_k h_i^k \land h_j^k \text{ with } h_k = \begin{cases} 1, & \text{if sentence } i \text{ contains } \text{k}^{th} \text{ word of the } \text{document dictionary} \\ 0, & \text{otherwise} \end{cases} \quad (17)
\]
In order to evaluate our algorithm adapted to perform summarization, we used the English sub-corpus of Concisus Corpus of Event Summaries [39], which contains 78 event reports and respective summaries, distributed across three different types of events: aviation accidents, earthquakes, and train accidents. To evaluate the quality of the summary, we used standard ROUGE [40], namely ROUGE-1, which is the most widely used evaluation measure for this scenario. We generate 3 sentence summaries, commonly found in online news web sites, like Google News. We compared our algorithms against LexRank [41] and Centrality [42] for a better understanding of the improvements. We also measured Max ROUGE-1 which was obtained by selecting the highest ROUGE-1 value among all possible combination of three sentences.

Table I outlines the result of our algorithms and the baselines on the Concisus dataset. Method CONST1 outperforms the LexRank and Centrality baselines, as well as the other two methods MDOM and MCUT.

<table>
<thead>
<tr>
<th>Alg.</th>
<th>LexRank</th>
<th>Centrality</th>
<th>MDOM</th>
<th>MCUT</th>
<th>CONST1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROUGE-1</td>
<td>0.428</td>
<td>0.443</td>
<td>0.482</td>
<td>0.489</td>
<td><strong>0.506</strong></td>
</tr>
</tbody>
</table>

**TABLE I**

ROUGE-1 RESULTS ON THE CONCISUS DATASET. MAX ROUGE-1: 0.646.

B. Support Vector Machine

We wanted to make sure that our algorithm was still performing comparably to the LIBSVM for the problem of SVM, even though we extended SMO to a more general class of problems. In order to evaluate our algorithm we used the datasets “adult” (adu.), “webpage” (web.), “cod-rna” (cod.), and “splice” (spl.) from [43]. The adult dataset is composed of nine partitions: a1a → a9a and the webpage dataset is composed of height partitions: w1a → w8a. For both datasets, each partition starts with a small training set and a large testing one, and ends with a large training set and a small testing one. We run our experiments on each partition. Implementation-wise, we used the standard LIBSVM code wrapped in the OpenCv library with the RBF kernel $K_{RBF}(\mathbf{x}_1, \mathbf{x}_2) = e^{-\gamma \|\mathbf{x}_1 - \mathbf{x}_2\|^2}$. For our approach we used $\hat{K}(\mathbf{x}_1, \mathbf{x}_2) = 1 - \hat{K}_{RBF}(\mathbf{x}_1, \mathbf{x}_2)$, $\varepsilon = 10^{-5}$. We initialize our system with ≤10 random support vectors. We also used the same grid search ($\gamma \in [0, 1]$, $\gamma_{incr} = 10^{-5}$) for both methods for fair comparison. Note that our implementation did not account for further optimizations including caching and shrinking, other heuristics based on the support vector margins or decision constraints [2]. Table II shows the baseline on the aforementioned datasets and Table III shows the results of our algorithm. While our approach provides slightly better accuracy, our solution is also sparser in terms of numbers of support vectors. Hence our approach only takes fewer iterations to converge, leading to a training time smaller than with the standard LIBSVM approach.

C. Toy Experiments for rank(C) > 2

In order to test our approach for rank(C) > 2, we used an extended version of the maximum cut problem described in Eqn. (16). The new QP problem is formulated as follows:

\[\mathbf{x}^* \leftarrow \arg\max_{\mathbf{x}} \mathcal{F}(\mathbf{x}) = (1 - \mathbf{x})^T (\mathbf{A} - \mathbf{I}) \mathbf{x}\]

s.t. \(\mathbf{C} \mathbf{x} = \mathbf{d}, \mathbf{I} = \text{identity}, \mathbf{x} \in [0, 1]^n\)

\(\mathbf{A} = \{0, 1\}^{m \times n}, \mathbf{C} = \{-1, 1\}^{m \times n}\) (18)

where \(C_i^j = 1\) and \(d_i = N\) and \(\forall i \geq 1\)

\(C_{ij} \in \{0, 1\}^m, (C1), = 0\) and \(d_i = 0\)

\(\mathbf{A}\) is randomly initialized with a level of sparsity of 70%, i.e., only 30% of the values of \(\mathbf{A}\) are 1s. We fix rank(C) = m, and to guarantee that \(\mathbf{C}\) has \(m\) independent columns, we build \(\mathbf{C}\) as follows:

\(\mathbf{C} = \begin{bmatrix} 1^T \\ 2I - 11^T \\ 1 \end{bmatrix} \mathbf{R} \in \{-1, 1\}^{m \times m - 1}

\(\mathbf{R} \in \{-1, 1\}^{m \times m - 1}\) (19)

Here the first \(m\) columns of \(\mathbf{C}\) are guaranteed to be independent from the \(m - 1 \times m - 1\) sub-matrix \(2I - 11^T\). It is also clear that the \(m_{th}\) column is also independent from the first \(m - 1\)'s. In order to guarantee that \((\mathbf{C} \mathbf{1}^T)_{i \geq 1} = 0\), we define \(\mathbf{R}\) such that \((\mathbf{R} \mathbf{1})_i = m - 4,\ i.e., each row of \(\mathbf{R}\) has \((m - 2 + \frac{n - m}{2})\) 1’s and \((2 + \frac{n - m}{2})\) -1’s. Since we also want to guarantee that the constraints \(\mathbf{C} \mathbf{x} = \mathbf{d}\) can be respected in the box space \(\mathbf{x} \in [0, 1]^n\), for \(k < m\) we choose \(\mathbf{R}_k = -(2I - 11^T)_k\) and for \(k \geq m\) we simply set \(\mathbf{R}_k\) to be a random column of \(2I - 11^T\) and \(\mathbf{R}_{k+1} = -\mathbf{R}_k\). Finally we randomly swap columns and rows of \(\mathbf{C}\) to minimize the biased introduced by the sequential construction of \(\mathbf{C}\). The identity matrix \(\mathbf{I}\) is introduced to push
the values of $x$ to be binary. The additional $m - 1$ constraints of $C$ can be viewed at compatibility constraints enforced on the solution, i.e. while $A$ represents the node connectivity in the graph, $C_{i<i<\pi}$ represent sets of hyper-graph edges of node affinities for the final solution.

We set $N = 5$ and test two additional variants of Alg. 3 for $m = 4$ and $m = n - 2$ with $n = 100, 1000, 10000$. For each $(m, n)$ combination we run 100 instances of the QP problem with different $A$ and $C$. The algorithm variant “random working set” (RWS-SMO) is replacing the working set selection by picking randomly unique indices for $S^\ast$. The variant “null space reformulation” (NSR-SMO) is based on the problem reformulation Eqn. (6) of Lemma 2. Note that we do not compare with simplex or interior point methods because naive use of general LP solvers would be computationally too expensive for large $n$ and $m$. Table IV and V show the results of the different algorithms for $m = 4$ and $m = n - 2$.

The RWS-SMO is suboptimal as it picks dimensions that don’t necessarily need to be optimized, hence it takes more iterations to converge and the local optimum is worst than the other two approaches, showing the importance of the working set selection. For $m = 4$, the G-SMO algorithm is more efficient than its variants as it quickly recursively decomposes the small QP problem into smaller ones until a closed form solution can be used since $m \ll n$. For $m = n - 2$ the NSR-SMO variant is more efficient as it directly reduces to a bounded 2D conic equation.

**TABLE IV**

<table>
<thead>
<tr>
<th></th>
<th>$n = 100$</th>
<th>$n = 1000$</th>
<th>$n = 10000$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$F_{avg}$</td>
<td>Time</td>
<td>$F_{avg}$</td>
</tr>
<tr>
<td>RWS-SMO</td>
<td>1.325</td>
<td>1.21</td>
<td>1.938</td>
</tr>
<tr>
<td>NSR-SMO</td>
<td>128.2</td>
<td>5.71</td>
<td>132.19</td>
</tr>
<tr>
<td>G-SMO</td>
<td>132.1</td>
<td>2.82</td>
<td>139.5</td>
</tr>
</tbody>
</table>

**TABLE V**

<table>
<thead>
<tr>
<th></th>
<th>$n = 100$</th>
<th>$n = 1000$</th>
<th>$n = 10000$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$F_{avg}$</td>
<td>Time</td>
<td>$F_{avg}$</td>
</tr>
<tr>
<td>RWS-SMO</td>
<td>101.2</td>
<td>3.28</td>
<td>1252.5</td>
</tr>
<tr>
<td>NSR-SMO</td>
<td>115.1</td>
<td>8.12</td>
<td>1289.0</td>
</tr>
<tr>
<td>G-SMO</td>
<td>113.5</td>
<td>7.90</td>
<td>1317.1</td>
</tr>
</tbody>
</table>

**REFERENCES**


Stacked Kernel Extreme Learning Machine for Hyperspectral Image Classification

1st Yantao Wei  
School of Educational Information Technology  
Central China Normal University  
Wuhan 430079, China  
yantaowei@mail.ccnu.edu.cn

2nd Peng Zhang  
School of Electronic Information and Communications  
Huazhong University of Science and Technology  
Wuhan 430074, China  
zp_zhg@hust.edu.cn

3rd Huang Yao, 4th Jiazhen Xu  
School of Educational Information Technology  
Central China Normal University  
Wuhan 430079, China  
yaohuang@mail.ccnu.edu.cn, xujiazhen@mail.ccnu.edu.cn

5th Xinge You*  
School of Electronic Information and Communications  
Huazhong University of Science and Technology  
Wuhan 430074, China  
youxg@hust.edu.cn

Abstract—This paper proposes a new spectral-spatial hyperspectral image (HSI) classification method called stacked kernel extreme learning machine (SKELM). Specifically, it combines wavelet, KELM and Gabor filters into a hierarchical structure. The proposed method can take advantage of the abstract and invariant features provided by the hierarchical architecture with multiple layers. First of all, wavelet is used to reduce the observation noise of HSIs. Then KELM is adopted to obtain a set of pixelwise probability maps from the input data, and Gabor filters are used to explore the spatial information by refining these probability maps. These two operations (KELM and Gabor filtering) are alternated to form a hierarchical architecture. The discriminative spectral-spatial features can be learned at each layer of the hierarchical architecture. Finally, the learned spectral-spatial features are fed into the standard KELM for classification. The experimental results on three widely used HSIs reveal that SKELM outperforms some state-of-the-art methods in terms of classification accuracy, and has low sample complexity.

Index Terms—Hyperspectral image classification, hierarchical feature learning, spectral-spatial feature, kernel extreme learning machine, wavelet

I. INTRODUCTION

Hyperspectral images (HSIs) contain rich spatial and spectral information [1], and different objects contained in HSIs exhibit different spectral and spatial signatures. So the abundant spectral and spatial information given by HSIs provides the possibilities for high accuracy HSI classification. Currently, HSI classification has been widely used in many fields, such as ocean monitoring, ground elements identifying, and mineral exploration. Consequently, how to efficiently classify the HSIs has become a critical task [2].

Over the past few decades, a large number of methods have been proposed for HSI classification. Feature extraction plays an important role in image classification [3], [4]. Conventional HSI classification methods often belong to spectral-based methods. For example, dimensionality reduction methods have been used to reduce the redundant spectral information of HSIs [5]–[7]. However, pixels belonging to different objects may share similar spectral properties. So it is difficult for spectral-based methods to correctly classify HSIs. And these methods often lead to noisy classification maps. Aiming to overcome this shortcoming, the structural and contextual information can be used for HSI classification [2], [8]. Spatial information provides additional discriminant information related to the shape and size of different structures [9]. Consequently, various spectral-spatial classification techniques have been used for classification, such as morphological profiles [10], maximizer of the posterior marginal by loopy belief propagation (MPM-LBP) [11], edge-preserving filtering (EPF) [12], and spatial aware dictionary learning (SADL) [13].

Other than feature learning methods, effective classifier- al so play an important role in HSI classification. Many classifiers have been successfully used in HSI classification. SVM has been commonly used in HSI classification. However, it is difficult to choose suitable kernel functions, parameters [14]. Furthermore, SVM classifier cannot directly provide the probability outputs in the probability-based methods, so an estimation procedure is required [15]. Sparse representation classification (SRC) has also been used for HSI classification [16], [17]. However, SRC is time-consuming as, for each test sample, it needs an expensive coding process. Multinomial logistic regression (MLR) has also been successfully used in the filed of HSI classification [18]. Recently, Huang et al. proposed kernel-based extreme learning machine (KELM) for classification, which is faster and has good generalization ability [19], [20]. It has received a lot of attention in HSI classification [21]. It is related to the works presented in [22].
However, most of these methods are not based on “hierarchical” fashion. The hierarchical architecture can potentially produce more abstract and robust features at high levels [23], [24]. Consequently, it is a promising direction to design a hierarchical model for spectral and spatial information exploration. Recently, stacked autoencoder (SAE) and deep belief network (DBN) have been used for HSI classification in [25] and [26], respectively. However, these deep models are not designed according to the characteristics of HSIs, and most of these models have high sample complexity [27]. So they cannot deal with the small sample problem in the HSI classification.

To cope with these challenges and perform HSI classification effectively, this paper presents a hierarchical probability-based method called Stacked KELM (SKELM). It could provide more effective spectral-spatial features by using hierarchical structure. The contributions of our work can be summarized as follows.

- SKELM combines the wavelet, KELM and Gabor filters in a hierarchical fashion. It can explicitly learn discriminative spectral-spatial features by making use of the label information.
- SKELM is easy to implement. In the proposed method, learning can be done without iteratively tuning hidden nodes. Consequently, the SKELM is a feedforward method.
- SKELM has low sample complexity. In HSI classification, the acquisition of training samples requires expensive human labor, and the available training samples are limited in most cases. Consequently, our method can alleviate this problem.

The structure of this paper is as follows. Section II describes SKELM in detail. It combines the wavelets, KELM and Gabor filters to obtain discriminative spectral-spatial features. Section III shows experimental results and analysis on two widely-used HSIs. Finally, Section IV concludes this paper.

II. THE PROPOSED METHOD

The flowchart of the proposed SKELM is given in Fig. 1. The layers of it can be divided into three parts, namely, the preprocessing, stacked spectral-spatial feature learning and the KELM classifier. The preprocessing layer is composed of normalization and wavelet decomposition. The spectral-spatial feature learning is composed of stacked probabilistic output layers and spatial feature learning layers. The probabilistic outputs are determined by KELM over the previous layer pixel by pixel, and the spatial features are obtained by sliding the Gabor filters over the previous probabilistic output layer. In addition, the obtained spatial features are concatenated with spectral features obtained on the preprocessing layer.

A. Image Preprocessing

In this layer, two operations are implemented: Normalization and wavelet decomposition. Let $U$ and $L$ be the maximum and the minimum of the HSI, respectively. Then the normalized HSI can be obtained in the following way:

$$I_{ij}(n) = \frac{I_{ij}(n) - L}{U - L},$$

where $I_{ij}(n)$ is the value on the $n$th band of the pixel in the $i$th row and $j$th column. Additionally, image noise may be generated unavoidably in the HSI acquisition process and has a negative effect on subsequent image analysis. Then wavelet is used to reduce the noise of the normalized HSI in the proposed method. Wavelet has been widely used in filed of image analysis [28], [29]. In the proposed method, 1D-db5 wavelet is used. In our experiments, approximations $a_1$ and $a_2$ have been used. In this way, the noise in the details can be removed.

In order to reduce the influence caused by illumination and atmospheric noise, spectral derivative analysis is carried out on HSIs. In this paper, the derivatives of the $a_2$ are used (denoted by $d_0$). $d_0$ can describe the shape of the spectral curve [30]. Finally, the output is obtained by concatenating three parts ($a_1$, $a_2$ and $d_0$) in this layer. This process can be shown in Fig. 2.

B. Hierarchical Spectral-Spatial Feature Learning

This stage is composed of pixelwise classifiers and spatial filters. Specifically, the alternating between KELM and Gabor filters leads to a hierarchical architecture for HSI classification. Firstly, KELM produces a set of classification probability maps using the output of previous layer. Then Gabor filters are used to explore the spatial information from the outputs of KELM.

1) Probabilistic Output: Let $\{x_i, y_i\} (i = 1, \ldots, N)$ be $N$ training samples coming from previous layer, where $x_i \in \mathbb{R}^d$ and $y_i = (y_{i,1}, \ldots, y_{i,C}) \in \mathbb{R}^C$ is the label of the $i$th training sample, where

$$y_{i,j} = \begin{cases} 1, & x_i \in \mathbb{C}_j; \\ 0, & \text{otherwise}, \end{cases}$$

where $\mathbb{C}$ is the total number of the classes.
where $h()$ is the activation function, and $H$ is the hidden layer output matrix.

The $f_M(x_i)$ is the output vector and reflect the “probabilities” of belonging to different classes. Then the probability can be defined as

$$
\hat{f}_M(x_i) = \left( \frac{\min_{m\in\{1,2,...,C\}} f_M(x_i)}{\sum_{m=1}^C f_M(x_i)} \right) \ \text{for} \ \text{all} \ i \ \text{and} \ M \ \text{and} \ \text{all} \ x \ \text{and} \ \text{all} \ \text{classes} \ \text{C},
$$

Finally, the probabilistic output is

$$
\hat{f}_M(x_i) = (\hat{f}_M(x_i)_1 \ldots \hat{f}_M(x_i)_C)^T.
$$

KELM has the advantages of efficient computation and good classification performance. The input features of pixels can be processed one by one. In this way, a pixelwise probability maps that reflect the probabilities for each pixel belonging to different classes can be obtained. In this stage, the label information is used to obtain discriminative outputs.

2) Spatial Feature Learning: Furthermore, the spatial information can be used to refine the classification maps. Spatial filtering is carried out to postprocess the probabilistic outputs of KELM. Currently, there are many technologies can be used for spatial filtering. In this paper, Gabor filters is used due to its easy implementation.

Gabor functions have been shown to provide a good model of cortical simple cell receptive fields and are described by

$$
G(x, y) = \exp \left( -\frac{(x_0^2 + \gamma^2 y_0^2)}{2\delta^2} \right) \times \cos \left( \frac{2\pi \lambda}{\delta} x_0 \right) \ \text{s.t.} \ (6)
$$

$$
x_0 = x \cos \theta + y \sin \theta \ \text{and} \ \ y_0 = -x \sin \theta + y \cos \theta, \ \text{for} \ \text{all} \ x \ \text{and} \ \text{all} \ \text{classes} \ \text{C}, \ \text{for} \ \text{all} \ \text{orientations} \ \{0, \frac{\pi}{8}, \frac{\pi}{8}, \frac{3\pi}{8}, \frac{3\pi}{8}, \frac{\pi}{4}, \frac{7\pi}{8} \}, \ \text{and} \ \delta \ \text{is the effective width, and} \ \lambda \ \text{is the wavelength.}
$$

Once the filters are obtained, then each map of the previous layer can be filtered pixel by pixel. Then the new feature maps will be obtained. In order to overcome the oversmooth problem, the features obtained in the preprocessing layer are concatenated with the outputs of the Gabor filters. Note that the probabilistic outputs layer and spatial feature learning layer can be taken as a spectral-spatial feature learning unit.

3) Stacking Spectral-Spatial Feature Learning: Existing researches show that stacking unsupervised modules on top of each other can lead to deep feature hierarchy. In each layer of the hierarchy, the input is the output features of the layer below. Higher layer representations capture high-level dependencies between input features. Based on this observation, we stack the spectral-spatial feature learning units to form a hierarchical architecture in this paper.

C. Classification Layer

Finally, the spectral-spatial features produced by the hierarchical model are fed into a standard KELM for classification. The labels of the test samples are determined by the maximum value of the $f_M$.

In summary, the proposed hierarchical spectral-spatial method is not only simple but also effective. In the SKELM, label information is used in each probabilistic output layer. Consequently, the proposed SKELM with hierarchical structure is fit for HSI classification.

III. EXPERIMENTS AND DISCUSSIONS

In this section, the SKELM is compared with other spectral-only algorithms and some state-of-the-art spatial-spectral methods. Here, three commonly preferred performance indexes are used to assess the classification performance of each method. They are overall accuracy (OA), average accuracy (AA), and $k$ coefficient [31]. All the experiments were carried out using Matlab R2014a on a PC with an Intel i7-4790 3.60 GHz CPU and 12 GB RAM.

A. Data Sets and Experimental Setups

For comparison and validation, two well-known publicly available images were used in our experiments. They can be described as follows:

The first image called Indian Pines was collected by the AVIRIS sensor over North-western Indiana (see Fig. 3). This
image with $145 \times 145$ pixels contains 16 exclusive classes (see Fig. 3, there are 10366 labeled pixels). After removing 20 water-absorption bands, 200 bands were used for HSI classification. This image constitutes mixed pixels and the available data is unbalance [11]. Consequently, this image is challenging.

Fig. 3. (a) Pseudocolor Image of the AVIRIS Indian Pines scene. (b) Ground-truth.

The second image is called University of Pavia, which was collected by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor during a flight campaign over University of Pavia, northern Italy. In the experiments, 103 bands after removing 12 noisy bands were used, and this image covers a region of $610 \times 340$ pixels. There are 9 land-cover classes in total (see Fig. 4, there are 43923 labeled pixels).

Fig. 4. (a) Pseudocolor Image of the University of Pavia scene. (b) Ground-truth.

B. Experimental Results on the Indian Pines Image

In this section, we perform a comparative evaluation of our proposed method against several state-of-the-art methods on Indian Pines image. Table I shows classification results of different methods (2% of the labeled samples for each of the 16 classes were randomly chosen for training, and the rest (98%) was used for testing), where LBP-ELM is classification method based on local binary patterns and ELM, and MFL is the multiple feature learning [32]. For the sake of stability, each experiment is performed 10 times, and the reported results are the average values of these results. The bold numbers in the tables denote the greatest accuracy. Obviously, the proposed SKELM performs the best. The experimental results of SVM and KELM are obtained by inputting the original pixels into the classifier directly. These results show that exploiting spatial information of HSI can significantly promote classification accuracy.

Fig. 5 shows that SKELM significantly outperforms other methods, especially when 2% of the labeled samples are chosen as training samples. These results indicate that the proposed KELM has low sample complexity. This is very important due to that the labeled samples are often difficult and expensive to be collected in practice.

C. Experimental Results on the University of Pavia Image

On this dataset, 1% of labeled samples from each class are used for training. The experimental results given in Table II are the averaged results over 10 random runs. We have marked in bold typeface the best result for each class and the best results of OA, AA, and $\kappa$. Experimental results show that the proposed method obtains the best OA of 98.07%,
TABLE I
CLASSIFICATION ACCURACIES FOR THE AVIRIS INDIAN PINES IMAGE.

<table>
<thead>
<tr>
<th>No.</th>
<th>Class Name</th>
<th>SVM</th>
<th>KELM</th>
<th>EPF</th>
<th>MPM-LBP</th>
<th>SADL</th>
<th>MFL</th>
<th>LBP-ELM</th>
<th>SKELM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Alfalfa</td>
<td>33.27</td>
<td>35.63</td>
<td>10.00</td>
<td>51.73</td>
<td>75.77</td>
<td>66.92</td>
<td>92.88</td>
<td>87.31</td>
</tr>
<tr>
<td>2</td>
<td>Corn-notill</td>
<td>55.99</td>
<td>78.90</td>
<td>74.41</td>
<td>80.98</td>
<td>83.89</td>
<td>80.58</td>
<td>89.32</td>
<td>87.21</td>
</tr>
<tr>
<td>3</td>
<td>Corn-mintill</td>
<td>44.24</td>
<td>60.13</td>
<td>89.35</td>
<td>68.46</td>
<td>64.15</td>
<td>26.46</td>
<td>88.38</td>
<td>82.01</td>
</tr>
<tr>
<td>4</td>
<td>Corn</td>
<td>27.69</td>
<td>45.52</td>
<td>79.01</td>
<td>44.76</td>
<td>64.15</td>
<td>26.46</td>
<td>88.38</td>
<td>90.41</td>
</tr>
<tr>
<td>5</td>
<td>Grass-pasture</td>
<td>73.35</td>
<td>88.61</td>
<td><strong>95.15</strong></td>
<td>80.86</td>
<td>90.12</td>
<td>80.76</td>
<td>84.55</td>
<td>90.41</td>
</tr>
<tr>
<td>6</td>
<td>Grass-trees</td>
<td>83.48</td>
<td>95.70</td>
<td>73.93</td>
<td>97.87</td>
<td>97.34</td>
<td>91.20</td>
<td>98.10</td>
<td>90.90</td>
</tr>
<tr>
<td>7</td>
<td>Grass-pasture-mowed</td>
<td>66.00</td>
<td>62.17</td>
<td>50.00</td>
<td>29.20</td>
<td>100.0</td>
<td>67.20</td>
<td>92.80</td>
<td>88.80</td>
</tr>
<tr>
<td>8</td>
<td>Hay-windrowed</td>
<td>88.18</td>
<td>98.02</td>
<td>86.57</td>
<td>98.37</td>
<td>99.65</td>
<td>96.41</td>
<td>98.79</td>
<td>99.98</td>
</tr>
<tr>
<td>9</td>
<td>Oats</td>
<td>28.42</td>
<td>21.67</td>
<td>10.00</td>
<td>40.53</td>
<td>93.16</td>
<td>32.11</td>
<td>94.22</td>
<td>96.32</td>
</tr>
<tr>
<td>10</td>
<td>Soybean-notill</td>
<td>57.04</td>
<td>73.54</td>
<td>77.92</td>
<td>75.79</td>
<td>85.79</td>
<td>82.59</td>
<td>89.40</td>
<td>91.45</td>
</tr>
<tr>
<td>11</td>
<td>Soybean-mintill</td>
<td>70.89</td>
<td>83.89</td>
<td>71.76</td>
<td>90.25</td>
<td>91.55</td>
<td>90.50</td>
<td>93.24</td>
<td>93.44</td>
</tr>
<tr>
<td>12</td>
<td>Soybean-clean</td>
<td>35.91</td>
<td>71.20</td>
<td>74.80</td>
<td>79.53</td>
<td>66.27</td>
<td>70.70</td>
<td>83.58</td>
<td>91.23</td>
</tr>
<tr>
<td>13</td>
<td>Wheat</td>
<td>90.63</td>
<td>98.05</td>
<td>98.02</td>
<td>99.47</td>
<td>98.55</td>
<td>96.28</td>
<td>95.22</td>
<td>99.23</td>
</tr>
<tr>
<td>14</td>
<td>Woods</td>
<td>90.58</td>
<td>96.40</td>
<td>91.21</td>
<td>96.36</td>
<td>96.34</td>
<td>96.47</td>
<td>97.80</td>
<td>97.51</td>
</tr>
<tr>
<td>15</td>
<td>Buildings-Grass-Trees-Drives</td>
<td>33.47</td>
<td>59.77</td>
<td>78.84</td>
<td>63.17</td>
<td>84.78</td>
<td>74.11</td>
<td>86.99</td>
<td>80.06</td>
</tr>
<tr>
<td>16</td>
<td>Stone-Steel-Towers</td>
<td>77.74</td>
<td>70.59</td>
<td>87.15</td>
<td>50.54</td>
<td>90.32</td>
<td>73.01</td>
<td>69.35</td>
<td>92.58</td>
</tr>
</tbody>
</table>

OA 65.46 65.97 78.44 83.84 88.19 84.32 90.80 92.78
AA 59.80 56.55 71.76 71.74 87.45 75.11 89.03 92.23
κ 0.605 0.609 0.750 0.814 0.865 0.821 0.895 0.918

which is 2.24% higher than the second best (95.83%) achieved by MFL. So we can conclude that SKELM significantly outperforms other methods. The reason for this is that SKELM can effectively exploit the spectral-spatial information.

Finally, we analyze the impact of the number of the training samples on classification accuracy. Different percentages from 1% to 3% of labeled pixels in each class are randomly selected as training samples. Fig. 6 shows the OAs as a function of the number of training samples per class obtained by different methods. These experimental results show that SKELM obtains the highest accuracies in all cases. This confirms the conclusions made on Indian Pines data set.

IV. CONCLUSIONS

In this paper, for the purpose of improving HSI classification, an efficient hierarchical spectral-spatial-based method has been proposed that makes full use of the spectral and spatial information contained within HSI data. First, wavelet is used to reduce the observation noise of HSIs, and the derivative is used to extract spectral information. Then KELM is used to obtain a set of classification probability maps and Gabor filters are utilized to explore the spatial information by refining the obtained pixelwise probability maps. This two operations (KELM and Gabor filtering) are alternated to form a hierarchical architecture. Finally, KELM is used to classify image pixels. Experimental results have demonstrated that the proposed SKELM achieves higher accuracy with fewer training samples. However, it is difficult to determine the optimal number of layers previously. So we plan to develop a method to determine the optimal number of layers of the SKELM for a given HSI.

ACKNOWLEDGMENT

The authors would like to thank the University of Pavia and Prof. P. Gamba for kindly providing the ROSIS images of University of Pavia, Prof. D. Landgrebe for making the AVIRIS Indian Pines hyperspectral data set available to the community, Prof. G. B. Huang for sharing KELM source code, Prof. J. Li for sharing MFL source code, Dr. X. Kang for sharing EPF source code, Dr. C. Chen for sharing MH source code and Dr. A. Soltani-Farani for sharing SADL source code.

REFERENCES

TABLE II
Classification accuracies for the Pavia University image.

<table>
<thead>
<tr>
<th>No.</th>
<th>Class Name</th>
<th>SVM</th>
<th>KELM</th>
<th>EPF</th>
<th>MPM-LBP</th>
<th>SADL</th>
<th>MFL</th>
<th>LBP-ELM</th>
<th>SKELM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Asphalt</td>
<td>86.72</td>
<td>84.93</td>
<td>93.05</td>
<td>97.38</td>
<td>92.66</td>
<td>98.05</td>
<td>98.54</td>
<td>98.76</td>
</tr>
<tr>
<td>2</td>
<td>Meadows</td>
<td>96.71</td>
<td>97.19</td>
<td>95.37</td>
<td>99.45</td>
<td>98.92</td>
<td>99.61</td>
<td>99.23</td>
<td>99.57</td>
</tr>
<tr>
<td>3</td>
<td>Gravel</td>
<td>64.41</td>
<td>64.16</td>
<td>96.87</td>
<td>79.02</td>
<td>74.65</td>
<td>74.35</td>
<td>90.21</td>
<td>89.88</td>
</tr>
<tr>
<td>4</td>
<td>Trees</td>
<td>84.93</td>
<td>85.90</td>
<td>99.46</td>
<td>98.92</td>
<td>93.50</td>
<td>89.75</td>
<td>91.44</td>
<td>95.86</td>
</tr>
<tr>
<td>5</td>
<td>Painted metal sheets</td>
<td>98.51</td>
<td>97.28</td>
<td>98.09</td>
<td>97.52</td>
<td>99.38</td>
<td>98.45</td>
<td>94.33</td>
<td>98.94</td>
</tr>
<tr>
<td>6</td>
<td>Bare Soil</td>
<td>77.70</td>
<td>77.55</td>
<td>98.52</td>
<td>93.37</td>
<td>94.57</td>
<td>95.04</td>
<td>99.02</td>
<td>98.95</td>
</tr>
<tr>
<td>7</td>
<td>Bitumen</td>
<td>80.37</td>
<td>71.32</td>
<td>99.54</td>
<td>82.95</td>
<td>77.26</td>
<td>94.12</td>
<td>87.85</td>
<td>98.73</td>
</tr>
<tr>
<td>8</td>
<td>Self-Blocking Bricks</td>
<td>83.74</td>
<td>81.92</td>
<td>87.63</td>
<td>91.52</td>
<td>77.73</td>
<td>93.11</td>
<td>92.47</td>
<td>96.10</td>
</tr>
<tr>
<td>9</td>
<td>Shadows</td>
<td>95.15</td>
<td>96.27</td>
<td>96.88</td>
<td>98.94</td>
<td>99.14</td>
<td>91.54</td>
<td>47.09</td>
<td>98.66</td>
</tr>
</tbody>
</table>

OA = 87.80, 88.29, 95.06, 95.42, 93.28, 95.83, 91.47

AA = 85.36, 84.06, 96.16, 92.31, 89.76, 92.67, 84.22

κ = 0.851, 0.844, 0.935, 0.940, 0.912, 0.945, 0.887, 0.975
A Block-based Path Recognition of Slag Removal Using Convolutional Neural Network

Jeong-Soo Kim  
School of Computer Science and Engineering, Kyungpook National University  
Daegu, South Korea  
hot1074@naver.com

Geon-Tae Ahn  
Research Institute of Industrial Science and Technology  
Pohang, South Korea  
seeahn@rist.re.kr

Soon-Yong Park  
School of Computer Science and Engineering, Kyungpook National University  
Daegu, South Korea  
sypark@knu.ac.kr

Abstract—De-slagging is the method of removing slag on a ladle using a de-slagging machine. In general, only an experienced operator can control such a machine directly. In this paper, we propose a slag removal path estimation method using CNN (convolution neural network) to automate de-slagging task. We extract the block region based on the removal path of the experienced operator from the ladle image, and then learn CNN path after path labeling. Then, we divide the ladle image into blocks and obtain the probability value of each label, which is the CNN output value for each block, using the learned data. The slag removal path is estimated by applying the backward tracing method from the lower center end-point of the ladle after applying the posterior probability. Then we apply curve fitting to the slag removal path estimated by backward tracing. Finally, we compare the proposed slag removal path with the experienced operator’s slag removal path.

Keywords—De-slagging, backward tracing, convolutional neural network, posterior probability, Bezier curve fitting

I. INTRODUCTION

With the advent of Google AlphaGo and autonomous vehicle systems, interest in deep learning [1] has attracted the attention of many researchers. Deep learning is a field of machine learning. In general, machine learning directly extracts the features of data required for learning by human. However, deep learning uses the input data as it is and learns important features from the data on its own. Therefore, deep learning is also called end-to-end machine learning. In deep learning, neural networks are structures with hidden layers in several levels. Recently, the deep learning has many hidden layers, and the maximum number of nodes connecting nodes is several billion. In the past, it was difficult to process optimization and computation costs if there were many hidden layers. However, it is now possible due to using excellent hardware and GPU.

There are various neural networks such as CNN (convolution neural network), RNN (recurrent neural network), DPN (deep belief network) and GAN (generative adversarial network). Among them, CNN is used for image and speech recognition, and especially because of its excellent performance in image recognition, many studies are being conducted recently [2-3]. Therefore, in this paper, CNN is used to recognize and classify slag in ladle images.

Slag, dust, sludge, and waste refractory are the by-products that are generated in steelworks, and the biggest by-product is slag. Slag is produced because silicon, aluminum, calcium, etc. contained in iron ores are not reduced during iron making process. These slag are removed by experienced operator directly controlling the de-slagging machine as shown in Fig. 1. However, around the ladle containing the molten iron and slag is dangerous environment that can cause accidents to the de-slagging operator. Therefore, automation of de-slagging task is necessary for the safety of the operator. To automate the de-slagging task, the slag removal path estimation is first required.

In the path estimation method, there is a method of estimating the rock mining path using KNN (K-Nearest Neighbors) classification method of machine learning [4]. The KNN classification method simply saves learning data and does not require learning time. However, since the test sample data classification is compared with all the learning data, the classification speed reduces if the number of learning data increases.

In this paper, we propose a slag removal path estimation method using CNN in deep learning. Unlike the KNN classification method, CNN requires high computational

![Fig. 1. De-slagging task image.](image)
when learning, but it can be performed with low computational to classify new test samples after learning.

The structure of this paper is as follows. Section 2 describes the CNN learning data acquisition and path labeling method. Section 3 describes the posterior probability method, backward tracing method, and Bezier curve fitting [5]. In section 4, we analyze the CNN path learning results and compare our proposed slag removal path estimation results with the experienced operator’s path. Lastly, section 5 concludes the paper.

II. OBTAINING LEARNING BLOCK IMAGE AND PATH DATA

The proposed learning-block image and path data acquisition are shown in Fig. 2. First, the ladle region ellipse is extracted from the ladle image using the Open Source Computer Vision Library (OpenCV) [6] ellipse fitting function, and the ladle region of interest (ROI) is set based on the extracted ellipse. Then, the ladle region of interest is binarized using dithering [7]. Finally, a certain size block region of the experienced operator slag path is extracted from the ladle binarization image and path labeling is performed.

![Flow of the proposed learning-block image and path data acquisition](image)

Fig. 2. Flow of the proposed learning-block image and path data acquisition.

A. Extracting the ladle region and setting the ROI

The process of extracting the ladle region and setting the ROI is shown in Fig. 3. First, we convert the input ladle image into grayscale image and detect the boundary of the ladle image using a simple structure and an excellent performance Kenny edge detector [8]. Then we use the Open Source Computer Vision Library ellipse fitting function to detect the ellipse region in the image where the boundary line is detected. Since the ellipse of the ladle region is the largest in the input ladle image, we determine the largest ellipse as the ladle region. Finally, we set the ladle region of interest in the input ladle image using the center-point and major axis and minor axis of the ladder region ellipse.

B. Ladle image binarization using dithering

Original RGB ladle image does not provide significant feature information about the slag (other than slag thickness). Therefore, we use the CNN learning image which is converted from the color image to the binary image. When converting a ladle image into a monochrome image, if the pixel value is binarized to 0 or 255 according to the threshold value, the detailed feature of the slag disappears as shown in Fig. 4 (b). To solve this problem, we perform binarization using dithering. Dithering is a technique for representing shading or color using a limited color. Fig. 4 (c) shows the result of binarization using the dithering.

![Apply ladle image dithering binarization](image)

(a) (b) (c)

Fig. 4. Apply ladle image dithering binarization, (a): original image, (b): general binarization, (c): dithering binarization.

C. Select and extract learning-block region

CNN learning image is a block image of a certain size of the experienced operator slag removal path. First, as shown in Fig. 5, the experienced operator removal path is selected from start-point to the end-point in the input ladle image based on the experienced operator slag removal video. Then, a block region of a certain size is extracted as the center-point of the selected removal path. However, if all the points from the start-point to the end-point are extracted as a block region of a certain size, there is no difference between the near block

![The process of experienced operator slag removal path block extraction](image)

Fig. 5. The process of experienced operator slag removal path block extraction.
region images, so that there are many overlapping images. Thus, we extract the block image of a certain size by dividing the selected points from the start-point to the end-point at certain intervals.

D. Define block-based removal path label

CNN learns using labeled learning data. Label means that defines the properties of the learning data. Therefore, we define the direction of the slag removal path as a label.

As shown in Fig. 6, there are innumerable removal paths to define all the removal paths corresponding to the input ladle image as the removal path labels. Thus, we define the internal removal path of the previously extracted block region as a label. Assuming that the path inside the block is constant, the block-based removal path is defined as four removal path labels as shown in Fig. 7.

![Fig. 6. Slag removal path label.](image)

Fig. 6. Slag removal path label.

![Fig. 7. Block-based removal path label. L1 = 45°, L2 = 90°, L3 = 135°, L4 = STOP.](image)

Fig. 7. Block-based removal path label. L1 = 45°, L2 = 90°, L3 = 135°, L4 = STOP.

E. Labeling the path of the block region

The path labeling process of the block region extracted from the experienced operator removal path is shown in Fig. 8. As shown in Fig. 8 (a), the path of the selected block region in the experienced operator removal path calculates the direction vector angle between the coordinates of the center of each block region, and labels it with the previously defined block-based removal path label. As shown in Fig. 8 (b), the removal path of the last block region calculates the direction vector angle between the last pixel coordinate and the end-point coordinate, and labels the previously defined removal path label.

The removal path angle $d_i$ of each block region is calculated using (1). $i$ is the $i^{th}$ block region corresponding to the slag removal path, and $x_i$ and $y_i$ correspond to the center coordinates of the $i^{th}$ block region.

$$d_i = \frac{\cos^{-1}\left(\frac{x_{i+1}-x_i}{\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}}\right) \times 180}{\pi}$$

(1)

There is a problem that the calculated $d_i$ does not match the removal path label angle of the previously defined block-based. To solve this problem, We convert the angle to 45 if the calculated $d_i$ is in the range 0 $d_i \leq 70$, to 90 if the range 70 $d_i \leq 115$ and to 135 of the range 115 $d_i \leq 180$. Labeled with the label corresponding to the previously defined block-based removal path angle using the converted angle.

The labeling of the extracted block region as above is only for the L1, L2, L3 labels. Therefore, in this study, block image of a certain size is extracted from the slag-removed ladle image as shown in Fig. 9 (b) and the extracted block image is labeled with L4.

III. SLAG REMOVAL PATH ESTIMATION

The slag removal path estimation process is shown in Fig. 10. First, the ladle region is extracted from the input ladle image and the ROI is set, and the set ladle ROI region is dithered binarized. Then, the dithering binarized image is divided into certain size blocks at certain pixel intervals at the center of the block. Next, the posterior probability method is applied to the label probability value, which is the CNN result value of each divided block. The slag removal path is estimated by backward tracing from the end-point using the label probability values of each block to which the posterior probability method is applied. Finally, we apply Bezier curve fitting to the elimination path estimated by backward tracing.

![Fig. 8. The process of removal path labeling, (a): 1 point removal path labeling, (b): 10 point removal path labeling.](image)

Fig. 8. The process of removal path labeling, (a): 1 point removal path labeling, (b): 10 point removal path labeling.

![Fig. 9. Slag image, (a): Ladle image covered with slag, (b): Image of all slag removed.](image)

Fig. 9. Slag image, (a): Ladle image covered with slag, (b): Image of all slag removed.

![Fig. 10. Flow of the proposed slag removal path estimation.](image)

Fig. 10. Flow of the proposed slag removal path estimation.
A. Apply posterior probability method

We proposed a method to estimate the slag removal path to the region where the slag is mostly distributed by applying the posterior probability. The posterior probability method is applied using (2).

\[ P(Class) = P(Class|Slag) \times P(Slag) \quad (2) \]

\( P(Slag) \) is the proportion of slag in each divided block. The slag ratio of each block can be calculated using (3) in the dithering binary image. In the dithering binary image, pixel value 0 is molten iron and 255 is slag. That is, the number of pixel values 0 in each block is divided by the total number of pixels. \( P(Label|Slag) \) is the value of each label probability, which is the CNN output value. If the label probability values of each block are multiplied by the ratio of slag of each block, the label probability value decreases as the slag ratio is lower.

\[ P(Slag) = \text{Numbers of black pixel/total pixels} \quad (3) \]

B. Backward tracing method

In this study, we propose a method to estimate the slag removal path by using all of the label probability values of each block applied posterior probability. The proposed removal path estimation proceeds as shown in Fig. 11. \( B_1, B_2, \) and \( B_3 \) correspond to the top three blocks connected in the current block when the removal path is determined by backward tracing. \( P_1 \) is the probability value of the L3 label corresponding to the removal path from \( B_1 \) to the current block. \( P_2 \) is the probability value of the L2 label corresponding to the removal path from \( B_2 \) to the current block. \( P_3 \) is the probability value of the L1 label corresponding to the removal path from \( B_3 \) to the current block. If the L4 label of each block of \( B_1, B_2, \) and \( B_3 \) is the highest probability value, the values of \( P_1, P_2, \) and \( P_3 \) are 0. The slag removal path proceeds to the block having the highest value among \( P_1, P_2, \) and \( P_3 \) values, and when the values of \( P_1, P_2, \) and \( P_3 \) are all 0, it is decided that the slag at the upper part of the current block is completely removed, so that the process does not proceed anymore. The center point of the block that is no longer in proceed is used as the starting point for the slag removal path.

![Fig. 11. The process of removal path determination using backward tracing method.](image)

C. Bezier curve fitting

The slag removal path estimated by backward tracing is inefficient for proceeding the path of the de-slagging machine as shown in Fig. 12 (a). Thus, we apply the Bezier curve fitting to convert the slag removal path estimated by backward tracing into an efficient path to the de-slagging machine. The Bezier curve fitting can be calculated from (4). \( N \) is the step of the Bezier curve, and \( P_i \) is the control point of the Bezier curve. Fig. 12 (b) shows the results of 9th-order Bezier curve fitting.

\[ B(t) = P_0(1-t)^n + \binom{n}{1} P_1(1-t)^{n-1}t + \cdots + P_n t^n, t \in [0,1] \quad (4) \]

IV. EXPERIMENTAL RESULT

In this section, we analyze the CNN path learning results and compare our proposed slag removal path estimation results with the experienced operator’s path. We used a 352 x 240 resolution video data that recorded the process of removing slag directly by the experienced operator at the de-slagging work site. The size of the block image used for CNN path learning is 90 x 90. We obtained a total of 2614 data through extraction of learning-block image, among which 1568 training data and 1046 test data.

We applied the mini-batch method [9] to CNN path learning. Mini-batch is method of extracting and learning a certain number of learning data. For example, if there are 1000 learning data, only 100 learning data are randomly extracted and learned. In this experiment, we extract 100 randomly out of 1568 learning data with a batch size of 100, and learn through Lecun’s network for MNIST [10]. Table 1 shows the results of 3000 epoch training using mini-batch method. The learning accuracy is 98.6% and the test accuracy is 91.96%.

| TABLE I. The result of CNN path learning. |
|---------------------------------|------------------|
| Train                          | 98.60            |
| Test                           | 91.96            |
The proposed slag removal path analysis is compared with the experienced operator slag removal path according to the block region division interval. Fig. 13 shows the comparison results of slag removal path of the experienced operator and the slag removal path proposed in this study. Fig. 13 (b) shows the result of backward tracing by dividing the block region into 30 x 30 blocks at intervals of 1 pixel. Similarly, Fig. 13 (c), (d) and (e) show results of backward tracing by dividing the block region into 30 x 30 blocks at intervals of 5, 10, and 15 pixels. The blue solid line represents the 9th order Bezier curve fitting result, where the red solid line represents the slag removal path estimation result using backward tracing.

V. CONCLUSIONS

In this paper, we proposed CNN-based slag removal path estimation method for automation of de-slagging task. First, we obtained and labelled CNN learning block images based on experienced operator removal path, and learned CNN path using labeled 1 learning data. Then, the input ladle image is divided into block regions of a certain size for slag removal path estimation and the posterior probability method is applied to the probability of each label, which is the CNN output value of the divided block region using the learned data. Finally we estimated the slag removal path by applying Bezier curve fitting to the path obtained by backward tracing.

As future work, we are aiming to improve the learning accuracy and test accuracy by obtaining more learning block image data, and measure the error and accuracy of the experienced operator slag removal path and the proposed slag removal path estimation results. Also we are aiming to study the algorithm for real-time path determination because it takes much time to estimate the slag removal path.

ACKNOWLEDGMENT

This work was supported by the Convergence R&D Development Project from the Small and Medium Administration, Republic of Korea (S2392741)

REFERENCES

Local Binary Pattern Mapping on Graph-based Image Representation for Texture Classification

Srisupang Thewsuwan and Keiichi Horio
Graduate School of Life Science and Systems Engineering, Kyushu Institute of Technology, 2-4 Hibikino, Wakamatsu, Kitakyushu, Fukuoka 808-0196, Japan
Email: thewsuwan-srisupang@edu.brain.kyutech.ac.jp
Email: horio@brain.kyutech.ac.jp

Abstract—This paper proposes a general framework to develop feature descriptors based on the synergy between graph theory and local binary pattern mapping. Finding relevant information in an image is a significant issue for image discrimination. Recently, a graph-based representation has been used for texture characterization which aids in texture analysis and remains a challenging task. In this work, we propose a weighted neighborhood graph with multi-scale of radial distance pattern mapping for texture classification. The advantage of a rotation invariance of local binary pattern mapping is employed for extracting the local texture information from the graphs. These approaches are evaluated using three standard texture databases: Brodatz, Outex, and UIUC. As results, the proposed method is shown to be effective in texture classification as compared to other methods.

Keywords—LBP, graph theory, texture analysis, texture classification

I. INTRODUCTION

Texture is important information for characterizing the appearance of an image. Texture analysis has played an important role in texture classification by quantifying image properties. At present, there are growing number techniques which described in the literature for texture characterization. These techniques are able to extract and to characterize texture information using a local binary pattern (LBP) [1], [2], Fourier descriptors [3], Gabor filters [4], graph theory and complex networks [5], [6].

In recent years, image representation by graph theory has been employed for texture analysis [5], [7]. Graph-based representations have been used to characterize the topological of networks [6], whereas this structure can also apply for describing image structures. The numerical measure of connectivity between a vertex and its neighbors can be used to describe the texture attributes. Based on related works, Backes et al. have proposed the complex network model for texture analysis [7]. The graph theory was used to represent image pixels as a network. The coarseness and orientation of an image structure can be described regarding the topological properties of the network. Although this model was capable of employing for texture analysis and classification, there are some of important challenges. In our previous work [8], we proposed the LBP mapping approach for analyzing the spatial information. The approach adopted a scheme idea of the local binary pattern [2] to investigate the spatial arrangement of vertices for improving the original complex network model [7]. The results can ensure the efficiency of this method by using the interaction between the spatial arrangement in which was inspired by the LBP and the complex network model for texture classification. Therefore, the empirical synergy between the standard pattern recognition technique and graph theory is a promising direction for this work.

Based on the graph theory, the image pixels can be represented by the set of vertices and the set of edges. The weighted graph is used to describe the image structure by pairwise connection. This value importantly employs for extracting the information on the image texture and hence the deterministic of the weighted graph is focused on this paper. The difference of local pixels are generally used to obtain the weight of edges [7], [8]. However, the important property vector such as the direction is discarded from the numerical value of the weighted edges. Regarding the standard pattern recognition techniques such as LBP operator, there were many methods which have the purpose of extending the basic LBP operator. A scheme idea method that inspiring us for developing this work is CLBP method. Zhenhua et al. [9] proposed completed modeling of LBP or CLBP operator for texture classification. The fundamental idea of this technique is a decomposition two complementary component which including the sign and the magnitude features for extracting the texture information. Accordingly, the weight of edge property is defined as a spatial vector which described by a magnitude and a direction in order to obtain the weighted graph.

In this paper, we propose a method for characterizing the image texture which aids in discrimination by the synergy between the graph theory and LBP mapping. The radial graph represents image pixels with multiple radial distance patterns which are applied for generating feature vectors for texture analysis. The weighted graph thresholding and non-thresholding approached are employed to illustrate the spatial vector property. The riu2 mapping [2] is used to analyze the spatial arrangement of a central pixel and its neighbors in a texture element. Three standard texture databases, Brodatz [10], Outex [11] and UIUC [12] are used for evaluation. The experimental results show the effectiveness of the proposed method when compared to the complex
network model approaches and texture analysis based on conventional methods by the accuracy classification.

To describe the proposed method, the remainder of the paper is organized as follows. Section 2 explains our proposed method, which includes the main topic of pixel-based representation by graph and binary pattern transformation process. The feature descriptors are defined in section 3, followed by experimental results and discussions in sections 4. Finally, section 5 concludes this work.

II. PROPOSED METHOD

This proposed method section can be described by a pixel-based representation using the graph and binary pattern transformation. Fig. 1 shows an overview of our proposed approach. The system can be separated into two parts, radial distance pattern mapping, and the weighted graph. The first part illustrates the process of the radial distance mapping which constructs by using the Euclidean distance between a node and pairwise connections. We have set radial distance $r$ equal to 1, 2 and 3 as fixed window mapping with the multi-scale of the neighborhood. Fig. 1 is indicated the term of radial distances as colors, including green, dark blue, and orange according to the radius $r$, respectively. The weighted graph can be explained by two processes, deterministic of weighted graphs, and binary pattern transformation.

Fig. 1. Overview of the proposed approach

network model approaches and texture analysis based on conventional methods by the accuracy classification.

To describe the proposed method, the remainder of the paper is organized as follows. Section 2 explains our proposed method, which includes the main topic of pixel-based representation by graph and binary pattern transformation process. The feature descriptors are defined in section 3, followed by experimental results and discussions in sections 4. Finally, section 5 concludes this work.

II. PROPOSED METHOD

This proposed method section can be described by a pixel-based representation using the graph and binary pattern transformation. Fig. 1 shows an overview of our proposed approach. The system can be separated into two parts, radial distance pattern mapping, and the weighted graph. The first part illustrates the process of the radial distance mapping which constructs by using the Euclidean distance between a node and pairwise connections. We have set radial distance $r$ equal to 1, 2 and 3 as fixed window mapping with the multi-scale of the neighborhood. Fig. 1 is indicated the term of radial distances as colors, including green, dark blue, and orange according to the radius $r$, respectively. The weighted graph can be explained by two processes, deterministic of weighted graphs, and binary pattern transformation.

To describe the proposed method, the remainder of the paper is organized as follows. Section 2 explains our proposed method, which includes the main topic of pixel-based representation by graph and binary pattern transformation process. The feature descriptors are defined in section 3, followed by experimental results and discussions in sections 4. Finally, section 5 concludes this work.

Fig. 1. Overview of the proposed approach

II. PROPOSED METHOD

This proposed method section can be described by a pixel-based representation using the graph and binary pattern transformation. Fig. 1 shows an overview of our proposed approach. The system can be separated into two parts, radial distance pattern mapping, and the weighted graph. The first part illustrates the process of the radial distance mapping which constructs by using the Euclidean distance between a node and pairwise connections. We have set radial distance $r$ equal to 1, 2 and 3 as fixed window mapping with the multi-scale of the neighborhood. Fig. 1 is indicated the term of radial distances as colors, including green, dark blue, and orange according to the radius $r$, respectively. The weighted graph can be explained by two processes, deterministic of weighted graphs, and binary pattern transformation. On

the deterministic of the weighted graph, we propose two approaches for extracting the local texture information which performs the weighted graph thresholding and the weighted graph non-thresholding. Then, the three binary patterns are obtained as shown in Fig. 1. Feature descriptors are derived by $riu2$ mapping technique [2], which the final histogram is concatenated histograms from pattern_1, pattern_2, and pattern_3.

A. Pixel-based representation by the graph

1) Radial distance pattern mapping: Graph theory [6] is employed to represent an image structure in this work. The radial distance pattern mapping is generated for an increasing scale of the pixel connectivity as shown in Fig. 2. The radial graph can expand into other vertices based on the distance measurement and multi-scale feature extraction. Each pixel of an image is denoted by a vertex $(v_i)$ in the graph. Two vertices are connected when Euclidean distance $d(v_i, v_j) \leq r$ where $r = 1, 2$ and 3 are indicated by gray color.

Fig. 2. An example of pixel-based radial distance mapping based on graph representation. The $v_i$ and $v_j$ are represented vertices corresponding to pixel $i$ and $j$, and $W(e_{ij})$ is a weight of edge between pixel $i$ and $j$ (the first row). Based on radial distance mapping, we can generate a radial graph into three patterns which based on their Euclidean distance, specifically, in the term of radial distance $r = 1, 2$ and 3 (the second row). Then, image-based graph representation can show in the last row. The local pixels that have Euclidean distance $d(v_i, v_j) \leq r$ where $r = 1, 2$ and 3 are indicated by gray color.

2) A weighted graph: The weight of edges is used to represent a structure in which pairwise connections have some numerical values. In case of the image represented by the graph, a difference between the intensity of pixels can be used to generate the weighted graph and consequently this approach can apply for characterizing local image textures. The deterministic the weighted graph aims in this paper. We approach to construct the weighted graph based on the difference between the intensity of pixels and its neighbors.
Suppose that \( I(i) \) is a pixel on the image which connecting to its neighbors, \( I(N) \). The weight of edge is \( I(i) - I(j) \). Moreover, we approach to modify a pixel image for the purpose of variance reducing. Three weighted graphs are proposed to compare performance as shown in Fig. 3. First, an original pixel is used as a numerical value, which is denoted by weight-center (WC). Second, the average value of the neighborhood \( I(N) \) is used instead of the center pixel, which is denoted by weight-mean (\( W_\mu \)). The third approach is the thresholding which is denoted by weight-median (\( W_M \)). Fig. 3 illustrates an example of three weighted graphs.

The weight of edge \( W(e_{ij}) \) is given by:

\[
W(e_{ij}) = \begin{bmatrix} WC(e_{ij}) \\ W_\mu(e_{ij}) \\ W_M(e_{ij}) \end{bmatrix}
\] (1)

where:

\[
WC(e_{ij}) = I(i) - I(j) \\
W_\mu(e_{ij}) = \mu(I(N)) - I(j) \\
W_M(e_{ij}) = med(I(N)) - I(j),
\]

where \( I(i) \) and \( I(j) \) are intensity of pixel \( i \) and \( j \). The neighbors of \( I(i) \) is denoted as \( I(N) \) where \( N = \{1, 2, \ldots, p_n\} \). Based on radial distance \( r_n \), a set of three weighted graphs are given by

\[
W_{r_n}(e_{ij}) = \begin{bmatrix} [WC(e_{i1}), WC(e_{i2}), \ldots, WC(e_{ip_n})] \\ [W_\mu(e_{i1}), W_\mu(e_{i2}), \ldots, W_\mu(e_{ip_n})] \\ [W_M(e_{i1}), W_M(e_{i2}), \ldots, W_M(e_{ip_n})] \end{bmatrix}
\] (3)

where \( n = 1, 2, 3 \). The number of neighbors within a radial distance \( r_n \) is denoted by \( p_n \) or \( r_n, p_n \). In this paper, \( p_1, p_2, \) and \( p_3 \) are equal to 4, 8, and 16 neighbors or mapping dimensions and \( r_1, r_2, \) and \( r_3 \) are equal to 1, 2, and 3. The different radial patterns are denoted by \( P(1, 4), P(2, 8) \) and \( P(3, 16) \), respectively (see Fig. 1 as an example).

### B. Binary pattern transformation

This paper considers the edge as a spatial vector that has a magnitude and a direction. We define the magnitude of the weighted graph by obtaining an absolute weight of edge value. The sign of weight of edge value is used for determining the direction of an edge. Thresholding can generate the binary pattern transformation. For the magnitude value, this approach has required a threshold for generating the binary pattern, whereas the sign can directly present. Accordingly, the weighted-graph thresholding and the weighted graph non-thresholding are defined as the magnitude and the direction of the edges in this paper. Fig. 3(a)-(c) show an example of the deterministic of weighted graphs, \( WC, W_\mu, \) and \( W_M \), which including the weighted graph thresholding and the weighted graph non-thresholding. These two approaches can extract varied information on local pixels which are essential for texture analysis.

1) Weighted-graph thresholding (\( Wb^t \)):

\[
Wb^t(e_{ij}) = \begin{cases} 1 & \text{if } |W(e_{ij})| \geq t^A \\ 0 & \text{otherwise} \end{cases}
\] (4)

where \( t^A \) is auto local thresholding which is generated by an average of the absolute weighted graphs value in the equation (3). This approach performs by converting the pixels whose weights are more than or equal to threshold \( t^A \) to 1, while the remaining pixels are converted to 0. A set of binary pattern of the three weighted graph thresholding for a radius \( r_n \) are defined as follows:

\[
Wb_{r_n}^t(e_{ij}) = \begin{bmatrix} [WC^t(e_{i1}), WC^t(e_{i2}), \ldots, WC^t(e_{ip_n})] \\ [W_\mu^t(e_{i1}), W_\mu^t(e_{i2}), \ldots, W_\mu^t(e_{ip_n})] \\ [W_M^t(e_{i1}), W_M^t(e_{i2}), \ldots, W_M^t(e_{ip_n})] \end{bmatrix}
\] (5)

2) Weighted-graph non-thresholding (\( Wb^n \)):

\[
Wb^n(e_{ij}) = \begin{cases} 1 & \text{if } sign(W(e_{ij})) > 0 \\ 0 & \text{otherwise} \end{cases}
\] (6)

A set of binary pattern of the three weighted graph non-thresholding for a radius \( r_n \) are defined as follows:

\[
Wb_{r_n}^n(e_{ij}) = \begin{bmatrix} [WC^n(e_{i1}), WC^n(e_{i2}), \ldots, WC^n(e_{ip_n})] \\ [W_\mu^n(e_{i1}), W_\mu^n(e_{i2}), \ldots, W_\mu^n(e_{ip_n})] \\ [W_M^n(e_{i1}), W_M^n(e_{i2}), \ldots, W_M^n(e_{ip_n})] \end{bmatrix}
\] (7)
III. FEATURE DESCRIPTORS

The $r_{iu2}$ mapping is employed for describing the binary uniformity patterns in this paper. The uniformity LBP mapping has proposed by Ojala et al. [2] for improving rotation invariant. The uniformity of texture primitive is captured when the binary pattern contains at most two-wise transitions between 0 and 1. Accordingly, this step evaluates the uniform patterns which can be used to analyze the spatial structure of local image texture. The vertex characterizing of the weighted graph thresholding and non-thresholding based on the $r_{iu2}$ mapping are denoted by $W^{b^t}-LPB_{riu2}(r_n, p_n)$ and $W^{b^o}-LPB_{riu2}(r_n, p_n)$, where $p$ is defined by a number of neighborhood and $r$ is radial distance value. In practice, the mapping from $W^{b^t}-LPB_{riu2}$ and $W^{b^o}-LPB_{riu2}$ have $p_n+2$ output values. The result is implemented with a lookup table of $2^{p_n}$. Then, the concatenated histogram of $r_1$, $r_2$ and $r_3$ are generated as a final feature vector. The weighted graph thresholding $W^{b^t}$ as feature descriptor is given by

$$
\Phi(v_i) = \begin{bmatrix}
W^{b^t} - LPB_{riu2}(1, 4) \\
W^{b^t} - LPB_{riu2}(2, 8) \\
W^{b^t} - LPB_{riu2}(3, 16)
\end{bmatrix}^T,
$$

(8)

The weighted graph non-thresholding $W^{b^o}$ as feature descriptor is given by

$$
\Psi(v_i) = \begin{bmatrix}
W^{b^o} - LPB_{riu2}(1, 4) \\
W^{b^o} - LPB_{riu2}(2, 8) \\
W^{b^o} - LPB_{riu2}(3, 16)
\end{bmatrix}^T,
$$

(9)

The final feature vector is given by

$$
\theta = [h(\Phi), h(\Psi)].
$$

(10)

IV. RESULTS AND DISCUSSION

To evaluate the proposed method, the nearest neighborhood with Euclidean distance is used as a discrimination function, following 10-fold cross-validations for texture classification. In the present study, three experiments were conducted to compare the results among the weighted thresholding, the weighted non-thresholding and the combination between the weighted graph thresholding and non-thresholding with different radial distance.

A. Databases

Three standard texture databases are used for evaluation as illustrated in Fig. 4, which are included as follows:

- The Brodatz Texture Album dataset [10] is used in texture analysis, and is a benchmark for evaluating methods. These data arranged in 100 classes, each class containing 10 grayscale samples of $128 \times 128$ pixels obtained by splitting the data of each class into 10 non-overlapping sub-images.
- The Suite Outex_TC_0013, or simply Outex [11] These data included 68 classes, each class containing 20 images.
- The experimental data contain total 1360 gray-scale texture images each comprising $128 \times 128$ pixels.
- The UIUC database is a very challenging database [12]. The images have significantly different viewpoints and scales due to perspective distortion and non-rigid transformation. The image size is $128 \times 128$ pixels. For each of 25 classes, 40 grayscale images were considered in the experiments.

B. Results obtained by the weighted graph thresholding and the weighted graph non-thresholding

The success rates obtained by employing the features which extracted using three weighted graph approaches with different radial distance which have listed in Table 1 and 2. In the experiments, we set radius $r_{max} = \sqrt{5}, \sqrt{7}$ and $\sqrt{9}$, where $r = \{1, \ldots, r_{max}\}$. It means that radial distance of $r_{max}$ is used to imply the scale of the radial distance pattern mapping as represented in Fig. 2. In the experiments, three weighted graphs based on $r_{iu2}$ mapping are denoted by $WC^{t^i}-LPB_{riu2}$, $WC^{t^o}-LPB_{riu2}$, $WC^{m^i}-LPB_{riu2}$ for thresholding approach and $WC^{m^o}-LPB_{riu2}$, $WM^{t^o}-LPB_{riu2}$ and $WM^{m^o}-LPB_{riu2}$ for non-thresholding approach as shown in Table 1, 2 and 3. Three databases are used to evaluate the proposed system.

As the results in Table 1 and 2, the $WC^{t^i}-LPB_{riu2}$ and the $WC^{m^o}-LPB_{riu2}$ descriptors produced significantly better results when compared with $WC^{t^o}-LPB_{riu2}$, $WC^{m^i}-LPB_{riu2}$, $WM^{t^o}-LPB_{riu2}$ and $WM^{m^o}-LPB_{riu2}$ descriptors, respectively, although the challenging database such as UIUC was applied for evaluation. These proposed approaches can demonstrate the importance of deterministic the weights of edges for characterizing the texture by using the graph theory. For the radial distance approach, this factor has influenced the performance to a feature descriptor. Table 1 shows the results of $WC^{t^i}-LPB_{riu2}$ and $WM^{m^o}-LPB_{riu2}$ descriptors with radius $r_{max}$ equal to $\sqrt{5}$ and $\sqrt{7}$ have better results than $\sqrt{9}$, whereas $WC^{t^i}-LPB_{riu2}$ showed the best results when radius $r_{max}$ was $\sqrt{9}$. These effects could be informed us about the local pixels with high radial distance can affect the performance because of the variation of pixel values, consequently which might be a variance with the illumination condition. For the results of the weighted graph thresholding and the weighted graph non-thresholding in Table 1 and 2, these approaches showed the weighted graph by non-thresholding is more efficient than thresholding in preserving the local difference information. The results performed in the same direction as discussed in [9]. Table 3 shows the success rate of the weighted graph which combined between thresholding and non-thresholding processes. The $\{WC^{t^i}-LPB_{riu2}, WC^{m^o}-LPB_{riu2}\}$ descriptor obtained outperforming results for all feature descriptors on Brodatz, Outex and UIUC databases. Accordingly, it can assure us that the magnitude and the direction of the spatial vector have influences on the texture classification by using graph-based representation.
C. Comparison of the proposed method and other methods

For more evaluation of our approaches, the additional conventional texture analysis methods chosen for comparison were including Fourier descriptors [3], Gabor filters [4], co-occurrence matrices [13], the complex network-based texture analysis [7], [8], LBP, LBP^ru2 texture operators [1], [2] and CLBP [9] methods. The considered methods are as follows:

**Fourier descriptors:** The 2D Fourier transform was applied to each image following with the shifting operator for the resulting spectrum. The feature descriptor obtained by summing all the absolute values of the coefficients from the shifted spectrum at the same radial distance from the image center [3].

**Gabor filters:** This method provides the spatial localization difference frequency, and orientation by using a sinusoidal plane wave [4]. From the convolution of these filters over an input image, we used energy as a descriptor. A total of 40 filters (combinations of 8 rotation filters and 5 scale filters) and a frequency range from 1.2 to 1.4 were applied in this experiment.

**Co-occurrence matrices:** Each matrix represents the joint probability of a pair of pixels which separated by determining distance d and direction θ. The co-occurrence matrix for d =
1 and 2 with angles $\theta = 0, 45, 90, \text{ and } 135$, in a non-systemic version for each image were computed in this experiment. Energy and entropy descriptors were approached as feature descriptor from each co-occurrence matrix to compose an image feature vector [13].

**Complex network-based approaches:** The complex network texture descriptor (CNTD) approach [7] proposed for texture analysis and classification by performing on the complex network model. The local spatial pattern mapping (LSPM) method [8] was proposed for improving the performance of the original complex network model [7] by applying the spatial arrangement system. The energy and entropy descriptors are used for evaluation in this experiment.

**LBP operators:** The LBP and LBP$^{riu2}$ operators were chosen in this experiment [1], [2]. In the experiment, LBP$^{riu2}$ descriptor was computed by the concatenation of the histograms when $(P, R) = (8, 1), (16, 2), (24, 3)$ to characterize a texture pattern, a total of 54 descriptors.

**CLBP:** This scheme method [9] is used the different local sign-magnitude to build CLBP_C, CLBP_S, and CLBP_M operators. In the experiments, the joint 3D histogram was employed to obtain CLBP_S/M/C. We used $(P, R) = (16, 2)$ with $riu2$ mapping, totaling 648 descriptors.

Table 4 shows the success rate of the proposed method and other texture analysis methods on Brodatz, UIUC, and Outex databases. The $\{W_C^{riu2}\text{-LBP}, W_C^{riu2}\text{-LBP}\}$ result in Table 3 selected as the proposed method for comparing with other methods. The results showed that the proposed method outperformed the other methods on the Brodatz database which is the standard validation for texture images. Fig. 4 illustrated samples in the Brodatz database which obtained by splitting the data of each class into 10 non-overlapping sub-images. These samples were made more challenging discrimination. In the Outex database, we used this database for validation the illumination change. The experimental results showed the illumination changes had influenced the performance of the proposed methods. Thus the adaptive variance analysis on pixels should be considered in future work. Although this illumination has an affected on the system, this approach was good to compare with other texture analysis methods. For the challenging UIUC, the proposed method result was more efficient than the complex network model approaches [7] [8] and LBP$^{riu2}$ operator [2]. This result confirmed the proposed approach has efficiency for texture classification. As the results in Table 4, the CLBP achieved the highest success rate on the UIUC database. However, this method required a high number of descriptors, whereas the proposed method is small. Accordingly, we can conclude that the proposed method can achieve good performances for characterizing and extracting features of image textures based on the synergy between the graph structure and the LBP mapping.

**V. CONCLUSION**

In this paper, we proposed a method for texture classification based on the synergy between LBP mapping and the graph theory. We used approaches which we refer to as the weighted graph thresholding and the weighted non-thresholding for illustrating the spatial vectors in which considering the magnitude and the sign. The experimental results show that the performance of our method in analyzing the weighted graph as spatial vectors improves the accuracy of texture classification as compared to the related methods and conventional texture analysis methods. Therefore, the synergy between the LBP and graph theory is a promising direction and research opportunity for future work.

**REFERENCES**


**TABLE IV**

<table>
<thead>
<tr>
<th>Method</th>
<th>No. of descriptors</th>
<th>Success rate [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brodatz</td>
<td>UIUC</td>
<td></td>
</tr>
<tr>
<td>Fourier descriptors</td>
<td>90</td>
<td>78.02</td>
</tr>
<tr>
<td>Gabor filters</td>
<td>40</td>
<td>76.58</td>
</tr>
<tr>
<td>Co-occurrence matrices</td>
<td>16</td>
<td>73.69</td>
</tr>
<tr>
<td>CNTD</td>
<td>54</td>
<td>78.38</td>
</tr>
<tr>
<td>LSPM</td>
<td>60</td>
<td>81.80</td>
</tr>
<tr>
<td>LBP</td>
<td>256</td>
<td>82.52</td>
</tr>
<tr>
<td>LBP$^{riu2}$</td>
<td>54</td>
<td>82.34</td>
</tr>
<tr>
<td>CLBP</td>
<td>648</td>
<td>85.46</td>
</tr>
<tr>
<td>Proposed method</td>
<td>68</td>
<td>91.36</td>
</tr>
</tbody>
</table>
Ontology-driven Acquisition of Verbal and Nominalization Patterns for Criminal Events

José A. Reyes-Ortiz  
Departamento de Sistemas  
Universidad Autónoma Metropolitana  
Ciudad de México, México  
jaro@azc.uam.mx

Maricela Bravo  
Departamento de Sistemas  
Universidad Autónoma Metropolitana  
Ciudad de México, México  
mcbc@azc.uam.mx

Leonardo Sánchez  
Departamento de Sistemas  
Universidad Autónoma Metropolitana  
Ciudad de México, México  
ldsm@azc.uam.mx

Abstract—Criminal events are present in the most of the crime news. They are described using verbal and nominalization phrases, which show an important variety of lexical structures in the natural language text of the newspapers. Such phrases need to be extracted and clustered in criminal events in order to use them to extract several criminal events from the news. In this paper, we present an approach to identify patterns for the characterization and, later, extraction of criminal events. We focus our representation on three categories of criminal events, such as: killing, violation and assault. The entire process involves the following issues: an ontological model for criminal events representation used to guide the patterns acquisition; a method to identify verbal and nominalization phrases from Spanish newspapers; and a clustering of phrases in each criminal events of interest.

Keywords—criminal events, verbal and nominal patterns, event ontology, computational linguistics, natural language processing

I. INTRODUCTION

Newspapers offer real-time information what is happening in a society and among citizens. The question what happened? represents an event (accident, speech or findings), additionally, we can find when it happened?, where it happened?, why it happened?, who is (are) involved? and much more data about the events. Every day, millions of event descriptions are generated around the world and a lot of them are written in Spanish. For a user interested in location information about a particular event, it represents a tedious and time-consuming task.

Information extraction about events from newspapers has generated many research opportunities for the Natural Language Processing (NLP) research community. However, it is considered a major challenge, to identify and extract events from text. Event extraction process needs a priori knowledge and, in addition, extracted events need to be represented and preserved for future requirements. Thus, ontologies play an important role, they provide mechanisms for representing real-world knowledge and make inferences with such information in computing environments [1, 2]. Prior knowledge must be created such as: rules, patterns, knowledge base, linguistic information or probabilistic values.

In this paper, we present an approach for acquisition of verbal and nominalization patterns for criminal event characterization from newspapers in Spanish. In the future, criminal events from news can be extracted or recognized.

Three criminal events are considered in our approach (killing, violation and assault) because they are important for crime prevention in a geographical Spanish-speaking region. Computational approach for automatic identification of criminal events extracted from daily news is crucial for the government to take decisions regarding the implementation of crime prevention policies and strategies to avoid violent events in the near future. However, to reach such relevant extraction of criminal events, it is necessary a knowledge base with linguistic patterns in Spanish delimited to a certain number of criminal events.

In this manner, we have delimited our approach for three criminal events with roles closely related with each event of interest as follows.

A Killing event is surrounded by murderer(s) and murdered, defined as:

(i) Murdered is someone who has been killed.
(ii) Murderer is someone who has killed a person.

A Violation event involves rapist(s) and victim(s), described as:

(i) Rapist is someone who has sexually abused a person.
(ii) Victim is someone who has been sexually abused by a rapist.

Finally, in an attack, specifically an Assault event, the roles closely related are attacker(s) and assaulted, which are described as:

(i) Attacker is someone who has committed an assault.
(ii) Assaulted is someone who has been stolen by an attacker.

Based on such criminal events of interest, which are represented in an ontology, several patterns are discovered using a semi-supervised method. These patterns grouped by criminal event are the resulting knowledge base in this work. It can be applied in Spanish newspapers in order to extract instances of three criminal events.
The rest of this paper is organized as follows. Section 2 presents the related work, which is briefly described in order to compare them with the approach presented in this paper. Section 3 exposes the ontological model used for representing criminal events. Section 4 describes the basis of the proposed approach for patterns acquisition, which is focused on verbal and nominal phrases. The acquisition of patterns approach is presented in Section 5. The conclusions and future work are presented in Section 6.

II. RELATED WORK

Acquisition of patterns is a task that is applied to a wide variety of areas. In Natural Language Processing, it has been the basis of several tasks, such as: information extraction, text classification, text clustering or knowledge acquisition. However, some works address the acquisition of patterns. As in [3, 4] presents a new method of automatic acquisition of linguistic patterns for Information Extraction. In [5] a method for paraphrase acquisition is presented, which use a single monolingual corpus often produce only syntactic paraphrases. Also, in [6] a pivot approach is presented by authors for extracting paraphrase patterns from bilingual parallel corpora, whereby the English paraphrase patterns are extracted using the sentences in a foreign language as pivots.

Information extraction is the most popular task, where they are used inside approaches for patterns acquisition. Event extraction from text has been proposed for several works. They can be organized by different features: starting points, language, detected events and representation method.

Regarding starting points, there are approaches that have been proposed in order to extract events from texts in social media ([7-9]), news articles ([10-16]), and scientific documents [17].

Many reported works detect events from text in English ([18], [19]), few works have been proposed for other languages, such as: Spanish [20], Russian [21], French [22], and Chinese [23-25].

We have decided merge “detected events” and “event extraction approach” as relevant features in order to explain in more details each work analysed in this state of the art, which are described as follows.

Several events are extracted from texts, however, the approach used for event extraction is critical for promising results. Thus, we have conducted an analysis of the works that have been developed for the event extraction. Such analysis is driven on the approach used and event detection. In [17] uses corpus-based linguistic resources as vocabulary, n-grams and frame-based patterns, which play a central role in financial events extraction. Real-world events independent of the domain are identified in [7], by using time-dependent semantic similarity measures that are consistent with static measures of similarity but provides high temporal resolution. The classification of events as event detection is presented in [8], who find the distinguishing terms between tweets in events and measure their similarities with learning language models such as ConceptNet. Ontologies as a representation model or to guide the event extraction task, as in [10] that construct a local ontology from news stories based on events, in [12] that populate an ontology, and finally, in [18] an ontology is used to drive the event recognition.

Social media is a resource plagued with events. Therefore, they have been widely used as a starting point for event extraction approaches, which are presented as follows. In [9] a weakly supervised approach is proposed for extraction new categories of events from Twitter, using seed-based machine learning.

In the other hand, also, news texts contain events, as shown in [11, 15] that investigate the global social event extraction by processing online news, authors build an event model and extract event information by classifying English news text using lexical and semantic patterns. Events related to finances are extracted in [13] from news articles, exploiting a framework with components as Gazetteers and Word Sense Disambiguation (WSD), a domain-specific ontology is used as a seed for the event detection process. A system’s architecture is described in [14] for integrating event information in a global crisis monitoring system, addressing several NLP-involved tasks: news geo-tagging, automatic pattern learning, pattern specification language, information aggregation. Security-related events extraction from online news is introduced in [16] through event classification method based on domain-specific inference rules, an approach for event geo-tagging based on utilization of lexical or semantic patterns, a simple method for cross-lingual event information fusion, and techniques for scoring the relevance rank.

In the biomedical domain, events involve biomolecules such as genes, transcription factors, or enzymes, for example, have a central role in biological processes and functions and provide valuable information for describing physiological and pathogenesis mechanisms. Event extraction from biomedical literatue has a broad range of applications [17, 19], predictive model inferred from a manually tagged corpus [20], including support for information retrieval, knowledge summarization, and information extraction and discovery.

In recent years, linguist approaches [22], deep learning techniques [23], approaches that consider language phenomena [24] and methods of news event extraction based on subject elements, which mixes the study of news topic sentence extraction and the research of event extraction together [25].

Under the review of the state of the art in event extraction from text, we have detected a lack of resources for event extraction from Spanish text. Therefore, this paper has considered an ontology model as a starting point, where criminal events are represented; also, three categories are considered because they are highly related to with citizens’ health and crime prevention in Spanish communities as our essential feature.
III. CRIMINAL EVENT ONTOLOGY

A system ontology for representing criminal events is presented in [26], which is modularized into three individual ontologies: space ontology, time ontology and event ontology. Authors consider the event module as core ontology for criminal events and time and space modules as peripheral ontologies.

The core ontology is based on CriminalEvent class from which subclasses are released in order to populate three classes of criminal events. We adapt the ontology offered in [26] to represent the three criminal events in our interest.

The Manchester syntax for OWL 1.1 [27] is used for presenting the adapted ontology for the semantic representation of knowledge in a user-friendly syntax. In this manner, the resulting ontology is composed as follows.

```
Class: Killing
  SubClassOf: CriminalEvent

Class: Violation
  SubClassOf: CriminalEvent

Class: Assault
  SubClassOf: CriminalEvent
```

The three criminal events and involved persons have semantic relationships between them. These relationships are presented below.

The **Killing** event can be qualified, guilty or in a fight. This crime incorporates all those situations in which a human is deprived of his life by any means,

This crime incorporates all situations in which a human being is deprived of his life by any means, without a relationship or affinity that links the victim with the murderer. It is characterized with two participants: murderer and murdered. It generates two ontological relations shown as follow.

```
ObjectProperty: hasMurderer
  Domain: Killing
  Range: Murderer

ObjectProperty: hasMurdered
  Domain: Killing
  Range: Murdered
```

In the **Violation** event, also called, rape, sexual abuse or incest, which is carried out through the forced or deceptive copulation. In this event also two participants are involved: rapist and victim, generating two ontological relations shown as follow.

```
ObjectProperty: hasRapist
  Domain: Violation
  Range: Rapist

ObjectProperty: hasVictim
  Domain: Violation
  Range: Victim
```

An **Assault** event involves robbery or assault, which is the squatting, robbery of property, theft, robbery of vehicles, animals, among others. It considers two participants: attacker and assaulted person, which produces two ontological relations shown.

```
ObjectProperty: hasAttacker
  Domain: Assault
  Range: Attacker

ObjectProperty: hasAssaulted
  Domain: Assault
  Range: Assaulted
```

Each class of criminal event also has a data type property with multiple values called hasSynonym defined as follow.

```
DataTypeProperty: hasSynonym
  Domain: CriminalEvent
  Range: xsd:string

CriminalEvent hasSynonym some xsd:string
```

This property has been defined to express several synonyms of each criminal event. Spanish synonyms for each criminal event are extracted from EuroWordNet [28, 29] and they will support pattern acquisition process. All synonyms are lemmatized, i.e. infinite form is determined and it is represented in criminal event ontology. For example, some synonyms for the assault-type criminal event are: rob / robar, assault / asaltar, attack / atacar.

The ontological model obtained represents the starting point of the proposed approach for learning patterns based on verbal and nominal phrases.

IV. VERBAL AND NOMINAL PHRASES

Natural language texts are a way of communicating situations, therefore understand them involves to place a particular attention on the events contained in the interested texts. It is necessary to create a representation of what is expressed in terms of space, time, causes and agents related to events. In texts, the linguistic structure plays as a means in order to express how the real-world situations are constructed.

Consequently, several language structures used to express events should be considered and analysed from text. The events are present in most historical, journalistic and specialized texts, in which, they are characterized by either a verbal phrase [30, 31] or a nominalization [32].

Both theories L. Tesnière [30] and M. Halliday [31] affirm that the verb is the core, around which, all elements of the sentence rotate. Elements of the sentence are divided in actor roles (agent, object, and recipient) and the circumstantial (instrument, force, time and locative). These theories position the proposed approach in the idea that the verbal core into sentences, represents events, and therefore, verbal phrases must be characterized.

A nominalization is a word formation that uses a verb as a noun, i.e. nouns derived from verbs. They are known as action nominal [32]. Being derived from verbs, nominalizations also express events. In this paper nominalizations are further
characterized for the Spanish language.

For Spanish, the nominalization, according to Hernando [33], consists of the creation of nominal derivatives by suffixation. It corresponds to a suffix denominal, deadjectival and deverbal. We have a special interest in deverbal nominalization, because it is formed with a base or root form of a verb plus a suffix. In addition, we rely on the nominalizations are referring to a verb and consequently to an event.

In this paper, we adopt the following two formal definitions of events. In addition, we characterize these events in order to extract verbal and nominalization patterns from newspapers in Spanish with the purpose of extract the in the future.

We use JAPE rules to characterize and annotate three criminal events from Spanish text. JAPE [34] is a Java Annotation Patterns Engine that provides finite state transduction over annotations based on regular expressions based on CPSL – Common Pattern Specification Language.

**Definition 1 (Verbal event)** A verbal event is composed by a verbal phrase (VP) or a verbal periphrasis (VH), which is characterized as follows.

**Rule:** VerbalEvent

```java
{VP} | {VH}
```

A verbal phrase (VP) is combined with two or more verbal forms: an auxiliary verbal form and other forms with head verbs. VP is characterized as macros with JAPE rules as follows.

**MACRO:** AUXILIARVERB1

```java
{Token.lema == "can"} // puede
| {Token.lema == "could"} //podría
| {Token.lema == "has"} // haber
| {Token.lema == "be"} // ser, estar
| {Token.lema == "should") //debería
| {Token.lema == "must") // debe
```

**MACRO:** VP

```java
{AUXILIARVERB1} [1,2]
| {SpaceToken}
| {HEADVERB} [1,3]
```

A verbal periphrasis phrase (VH) is a verbal complex form that consists of: an auxiliary verbal form, in which verbs are different to auxiliary verbs previously defined to new verbs that are necessary to create some periphrasis phrase; and connection elements that link auxiliary verbs with the modified form, they are either prepositions or conjunction 'what'/"que"; and the modified verbal form that must be an infinitive or a gerund verb. Thus, we define a new macro called AUXILIARVERB2 and characterize a VH as a new macro.

**MACRO:** AUXILIARVERB2

```java
{Token.string == "open"} // abrir
| {Token.string == "finish"} // acabar
| {Token.string == "reach"} // alcanzar
| {Token.string == "come"} //venir
| {Token.string == "say"} //decir
| {Token.string == "go"} // ir
```

**MACRO:** CONNECCTION

```java
{Token.string == "to"} // a
| {Token.string == "for"} // por
| {Token.string == "of"} // de
| {Token.string == "what"} // que
```

**MACRO:** MODIFIEDVERB

```java
{Token.category=="VLger"} // gerund verb
| {Token.category=="VCLIinf"} //infinitive verb
```

**MACRO:** VH

```java
{AUXILIARVERB2}
| {SpaceToken}
| {SpaceToken}
| {MODIFIEDVERB}
```

**Definition 2 (Nominal event)** A nominal event (NE) is characterized by a deverbal nominalization, which follows the structure exposed in the following JAPE rule.

**Rule:** NominalEvent

```java
{NE}
```

In Spanish, a nominal event is defined as a concatenation of verbal root plus a corresponding suffix, as shown below.

\[ NE= V_r + A_f_n \]

Where \( V_r \) expresses the root or stem of a verb in Spanish, from which inflected words are removed, thus verbs are reduced, e.g. the stem of “violate/violar” verb is “viol”. \( A_f_n \) represents the suffix that completes the meaning of the
nominalization. Some suffixes are, for example, -(a)toria, -sición, -(a, i)m(i)ento, -(a, i)ción, -a, -eo, -o. They do not have English translation because of the nature of the Spanish language with the exception of -tion/-sión -(a, i)tion/-(a, i)ción and suffix.

In the following example, we show the two linguistic structures considered in this paper that characterize events. In example (1), a verbal event is included with an attack (assault) event expressed by “was assaulted/fue asaltado” verbal phrase. Then, in example (2), a nominal event is presented through Violation event by a deverbal nominalization, verb (violate/viola) plus -(a, i)tion/-(a, i)ción suffix.

(1) ... a man was assaulted in ... / ... un hombre fue asaltado en ...

(2) The violation of a woman in ... / La violación de una mujer en ...

V. ACQUISITION OF PATTERNS

In this section, we present the proposed approach for acquisition of patterns for types of criminal events that is illustrated in Fig. 1 as an architecture. The entire process involves the following issues: an ontological model for criminal events representation used to guide the patterns acquisition; a method to identify verbal and nominalization phrases from Spanish newspapers; and a clustering of phrases in each criminal events of interest. The output of our approach is a knowledge base of verbal and nominal patterns for characterization of three criminal events.

Fig. 1. Acquisition of verbal and nominalization patterns

A. Starting point

Pattern acquisition process has as a starting point the newspapers in Spanish. They are manually labelled in three criminal events: killing, violation and assault. This corpus is composed by 1200 news headlines, 400 for each criminal event.

Each news was provided to two humans in order to examine it and they identify events at a finer-grained level from Spanish news. In cases where the judges disagree, a third human reviewed the news and he/she place a label to break up. In this way, we collect 1200 news tagged with three criminal events at a finer-grained level.

B. Pre-processing news

In this step, three processes are applied to texts in order to normalize and to leave the necessary information for the automatic identification of phrases and nominalizations. Lemmatization, stemming and POS tagging are carried out using TreeTagger [35] and they are enforced to text in both cases, pattern recognition and event extraction phases.

Lemmatization of news consists on transforming each word to its corresponding lemma, without inflections, so words can be analysed as a normalized item. The lemmatization process allows grouping several inflected forms of the same word into a single item. In this particular case, lemmatization process is necessary to make possible the application of verbal phrase rules from Spanish news.

In this case, stemming is an essential process to obtain the root of verbs, which inflections are eliminated and the stem of each verb has resulted, e.g. the stem of “violate/viola” verb is “viol”.

Finally, POS tagging process is necessary to identify verbal periphrasis phrases, since it is able to identify categories of gerund and infinitive verbs.

C. Identification of verbal and nominal phrases

A semi-supervised learning method is applied for identifying verbal and nominalization patterns driven by criminal event ontology from newspapers in Spanish. The objective of this step is to obtain a set of candidate verbal phrases and nominalizations to be patterns.

Automatic extraction of verbal phrases and nominalizations is the first step to ontology-driven pattern recognition. In this manner, all verbal phrases that match with JAPE rules called VerbalEvent are identified from Spanish news and they are sent to a list of candidate verbal phrases. In the same way, the nominalizations are also considered as candidate nominalizations. Therefore, they are automatically extracted with NominalEvent JAPE rule. Both rules produce the list of candidate phrases to be compared with synonyms of criminal events and finally filtered.

From the list of candidate patterns, a comparison with hasSynonym relation of the criminal event ontology is carried out. With respect to verbal phrases, the head verb is compared with the synonyms of ontology for each criminal event. On the other hand, each lemma verb stored in ontology is nominalized by suffix, i.e. the verbal root plus a suffix are compared with the main noun of the nominalization phrase. All matches, both verbal and nominalization patterns are extracted as a relevant phrase belongs to criminal events.

D. Clustering patterns

Relevant patterns are grouped by criminal event; duplicated are eliminated. A useful list of patterns is acquired; thus a final set of patterns is formed.

After the clustering patterns, a final set of patterns are
extracted for each criminal event, such as: 27 for Killing event, 39 for Violation and 38 for Assault event. Tables I, II, and III show a list of the three most frequent patterns organized by criminal event. However, the full list of pattern for the three events can be downloaded from a github project.

TABLE I. PATTERNS FOR KILLING EVENT

<table>
<thead>
<tr>
<th>Id pattern</th>
<th>Killing event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(kill</td>
</tr>
<tr>
<td>2</td>
<td>(was</td>
</tr>
<tr>
<td>3</td>
<td>(a</td>
</tr>
</tbody>
</table>

It should be emphasized that there are very specific patterns that rarely appear in the corpus, it doesn’t mean that they are no longer considered in extraction phase. E.g. the following pattern appears, at most, three to five times at Killing event: (was | were) (calcined | lynched) // (fueron) (calcinados | linchados).

TABLE II. PATTERNS FOR ASSAULT EVENT

<table>
<thead>
<tr>
<th>Id pattern</th>
<th>Assault event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(a</td>
</tr>
<tr>
<td>2</td>
<td>(a</td>
</tr>
<tr>
<td>3</td>
<td>(assault</td>
</tr>
</tbody>
</table>

A less frequent pattern for the assault event is as follows: storm // tomar por asalto.

TABLE III. PATTERNS FOR VIOLATION EVENT

<table>
<thead>
<tr>
<th>Id pattern</th>
<th>Violation event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(a</td>
</tr>
<tr>
<td>2</td>
<td>(violate</td>
</tr>
<tr>
<td>3</td>
<td>(sexually) (abuse</td>
</tr>
</tbody>
</table>

A less frequent pattern for the violation event is illustrated as follows: (touch | touched | touches) intimate parts // (toca | tocó | tocán | tocaron | tocar) partes íntimas.

VI. PATTERN EVALUATION

In the acquisition phase, a total of 104 nominal and verbal patterns were obtained, which were applied on the 1200 newspapers that compose the three types of criminal events. In total, these patterns produced 1487 events of the three types: killing, assault and violation. The number of events is greater than the number of newspapers because it is possible that in the same newspaper talk about two or more criminal events. Table IV shows the distribution of events detected either by nominal or verbal patterns for each category.

TABLE IV. THE DISTRIBUTION OF DETECTED CRIMINAL EVENTS

<table>
<thead>
<tr>
<th>Criminal event</th>
<th>Nominal patterns</th>
<th>Verbal patterns</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Killing</td>
<td>201</td>
<td>277</td>
<td>478</td>
</tr>
<tr>
<td>Assault</td>
<td>266</td>
<td>397</td>
<td>663</td>
</tr>
<tr>
<td>Violation</td>
<td>127</td>
<td>219</td>
<td>346</td>
</tr>
<tr>
<td>Total</td>
<td>594</td>
<td>893</td>
<td>1487</td>
</tr>
</tbody>
</table>

The distribution shows that criminal events signalling by a verbal pattern predominate rather than by a nominal pattern. However, a notable contribution of this work is to consider the phenomenon of nominalization in criminal events for a future prediction or early detection of criminal events from newspapers in Spanish.

The 104 total patterns were evaluated with respect to the accuracy of the method used for the acquisition of patterns. This evaluation was carried out by humans who analysed the resulting pattern and decided whether or not it was relevant to the category of criminal event. Equation (1) has been used to measure the accuracy of patterns by category of criminal events.

\[
\text{Precision}(P) = \frac{|\text{Relevant patterns} \cap \{\text{Acquired patterns}\}|}{|\text{Acquired patterns}|} \tag{1}
\]

Table 5 reports the precision results for each category of criminal events.

TABLE V. RESULTS OF PATTERNS ACQUISITION

<table>
<thead>
<tr>
<th>Criminal event</th>
<th>Precision (P)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Killing</td>
<td>0.925</td>
</tr>
<tr>
<td>Assault</td>
<td>0.921</td>
</tr>
<tr>
<td>Violation</td>
<td>0.897</td>
</tr>
<tr>
<td>Overall</td>
<td>0.914</td>
</tr>
</tbody>
</table>

1https://github.com/alreyesor/patrones
The evaluation focused on the process of patterns acquisition from newspapers in Spanish text and it being supported by an ontology of criminal events. Our evaluating process has provided an overall precision of 0.914.

VII. CONCLUSIONS AND FUTURE WORK

This paper has presented a novel ontology-driven approach for acquisition of verbal and nominal patterns to characterize criminal events from Spanish newspapers. The proposed approach is basically focused on a set of news, manually labelled by humans, into three criminal events: killing, violation and assault. From such set of news, the approach pre-processes Spanish text in order to facilitate verbal and nominalization phrase extraction; then, a list of candidate phrases is obtained and they are filtered according to synonyms that exist in the criminal event ontology. Finally, the candidate patterns are clustering by criminal event. As a result of the presented approach, a knowledge base with verbal and nominalization patterns for criminal events is acquired. It can be very useful for applications to identify events at a finer-grained level from Spanish news, event data extraction or question answering systems.

It is important to note that this paper focuses on news in Spanish, a language in which limited works have been proposed. Therefore, the importance of presented verbal and nominalization patterns and the possibility to use them for extracting three criminal events from Spanish news.

The main contributions of this paper is the set of verbal and nominalization patterns focussed on three criminal events such as killing, violation and assault. In addition, the set of patterns has been evaluated achieving an overall accuracy of 0.914.

As future work, the set of patterns should be evaluated. A task-based evaluation, specifically, criminal events extraction task can be performed in order to evaluate the patterns for each criminal event. Such evaluation must be achieved with a different corpus of news, even, a corpus extracted from social networks like tweets would be an interesting evaluation. In addition, early detection of criminal events is highly important for preventing the crime in society. So, a system to predicting crimes using a set of patterns presented in this paper, it would be a research challenge to be addressed. Also, the obtained knowledge base can be enriched with information about time and space of events already extracted in this paper.

ACKNOWLEDGMENT

This work was partly supported by SEP-PRODEP at UAM-A-CA-142 project. The authors would like to thank the Autonomous Metropolitan University Azcapotzalco and SNI-CONACyT.

REFERENCES


Classification of Keyphrases using Random Forest

Mireya Tovar Vidal  
*Faculty of Computer Science*  
Benemérita Universidad Autónoma de Puebla  
Puebla, Mexico  
mtovar@cs.buap.mx

Gerardo Flores Petlacaalcño  
*Faculty of Computer Science*  
Benemérita Universidad Autónoma de Puebla  
Puebla, Mexico  
gerardo.florespe@alumno.buap.mx

Azucena Montes Rendón  
*TecNM*  
Instituto Tecnológico de Tlalpan  
Mexico City, Mexico  
amr@cenidet.edu.mx

Meliza Contreras González  
*Faculty of Computer Science*  
Benemérita Universidad Autónoma de Puebla  
Puebla, Mexico  
mcontreras@cs.buap.mx

Ana Patricia Cervantes Márquez  
*Faculty of Computer Science*  
Benemérita Universidad Autónoma de Puebla  
Puebla, Mexico  
patty@cs.buap.mx

Abstract—Keyphrases are words or phrases from a document that can describe its meaning. A keyphrase integrates the general idea of a document and implicitly contains the resources that the author used during the development of its research to achieve his goal. Therefore, there is a need to create classification models that allow the clustering of keyphrases according to their content for simplify reading. In this paper, keyphrases classification from scientific publications based on LSA and some classifying techniques is proposed and implemented. The aim is to create a classification model based on the extraction of features from the input corpus, without enriching it using external resources such as Wikipedia or online resources. Process, task, and material are the classes considered from Computer Science, Material Sciences, and Physics publications domains. Results show that Random Forest was found to be the best classification technique of keyphrases with 60% of measure-$F_1$.

Keywords—Keyphrases, Natural Language Processing, Machine Learning, Latent Semantic Analysis

I. INTRODUCTION

One task of the natural language processing is the extraction of important topical words or phrases from documents, usually known as terminology extraction or automatic keyphrase extraction.

Keywords allow readers to decide whether or not a scientific publication or article contains material important to their interest. Keywords provide readers with suitable terms to use in web-based searches to locate other materials on similar topics. Articles, prepositions, conjunctions and general words are too common to serve as keywords. A keyphrase is a phrase of relative importance in a text, like a keyword, with two or more words [1].

Recognizing the resources or keyphrases used within a text is a task of interest in the area of Natural Language Processing, that in recent years has grown in importance due to its complexity and usefulness in helping understand knowledge in a better way.

SemEval (Semantic Evaluation) is an in-progress series of evaluations of semantic analysis systems. It proposes a list of tasks that involved natural language processing and semantics. In the 2017 edition, it proposed twelve tasks. In particular, the task 10 at SemEval 2017 deals an extraction of keyphrases automatically given a scientific publication [2]. In that task, there are three subtasks:

1) Identification of keyphrases. The aim is to identify all the keyphrases in the document or in the text.
2) Classification of identified keyphrases. There are three types:
   a) Process. They are related to some scientific model, algorithm or process.
   b) Task. They denote the application, problem, task, etc.
   c) Material. They identify the resources used within the document.
3) Extraction of relationships between two identified keyphrases. Every pair of them is labelled by one of three types: hiponym, synonym or none.

In this paper, a method to classify keyphrases into three categories (defined in the SemEval-2017 task 10, subtask 2), using Random Forest and features selection is proposed. The proposed features are based on the context, properties of the keyphrase and semantic relationships using only information provided by the dataset without queries to external resources. The results obtained show a favorable performance when the approach is compared with other works of the state of the art that use external resources for keyphrase classification.

The rest of this paper is organized as follows. In section II, some related work on keyphrase classification is presented. Theoretical information on classification techniques is presented in Section III. The method and the features used are described in Section V, the obtained results are shown in Section VI, and finally, some conclusions are presented in Section VII.

II. RELATED WORK

Other authors in the Natural Language Processing area have made studies in the task of classification of keyphrases, thus some related work is presented in this field.
In [3] a system was proposed that extracts context features of keyphrases for a per-topic classification. Three classification models are created using Support Vector Machines (SVM) and the scikit-learn library.

In [4] Convolutional Neural Networks (CNN), MLP meta-classifier and Bi-LSTM for keyphrase classifying are used. The classifiers are trained with extracted features, these features are context independent of the keyphrase so that a simpler classification task is done.

[5] proposed a system named MayoNLP, this system is used for keyphrase classification in scientific texts. The features used are the semantic similarity and keyphrase PoS patterns. The system solves the classification task as recognition of entities, the system uses SVM for clustering. It gets good results in the evaluation. The research is extended to the recognition of semantic relations among keyphrases.

[6] presents a system for keyphrase classification using lexical features. Semantics networks UML are used, to create a relationship between keyphrases, as well as grouping them by similar features, this grouping is then converted to a dictionary, and the new keyphrases are classified according to a similarity measure between the keyphrase terms and dictionary terms. In [7] a system for text classification using keywords and automatic learning machines was created. For each text, keywords are extracted using tf-idf techniques in combination with the similarity obtained from WordNet1. Later, the keywords are classified using Naıve-Bayes (NB), K-Nearest Neighbor (KNN) and Decision Trees using cross-validation. Finally, the prediction is made and the system is evaluated, the results show that Decision Trees have the best performance.

In [8] a system for word classification using Latent Semantic Analysis (LSA) is proposed. First, a matrix of terms and documents is created. Later, a matrix reduction is made using Singular Value Decomposition (SVD). Finally, the reduced vector from the matrix are grouped using K-Means. The final results are similar vectors that represent words semantically similar between them.

[9] uses LSA for text categorizing. First, a text collection is converted to matrix representation, know as Bag-Of-Words (BoW), later a dimensionality reduction is applied using SVD, and finally, a grouping by topics using SVM polynomial is performed. The grouping is done in five categories and the predictions for the test data got a Precision of more than 70% of the recognized categories.

Other works use LSA for the evaluation of semantic relationships in domain ontologies [10], [11] and [12].

In this work, an approach is presented where features from a corpus are extracted. Later, a matrix is created and reduced using LSA techniques. Finally, several classification algorithms are used to make predictions for the keyphrases of the test dataset with the aim is create a classification model that use only features from the dataset without considered external resources.

### III. Learning Methods

In text classification, a text document may partially match one or more categories or classes [13]. It needs to find the best matching category for the text document. Then, a given object is sought in a set of classes and that way it is determined which class it belongs to. In this case, the object is the keyphrase and the classes or categories considered are Process, Task, and Material from Computer Science, Material Sciences, and Physics publications domains. Given a training set $D$ of labeled keyphrases $(kf, c)$, where $kf$ is a keyphrase $(kf \in KF)$, $c$ is a class $(c \in C)$, then $(kf, c) \in KF \times C$. For example: $(kf, c) = (Information\ Extraction,\ Task)$ for some document $d \in D$, the keyphrase “Information Extraction” is a Task. Using a learning method, it is intended that a classifier learns to map keyphrases to classes. The type of learning used is supervised learning. The learning method takes the training set $D$ and keyphrase $KF$ as input and returns the classification model.

Below, some techniques or learning methods for classification keyphrases to classes, that were employed in this research, are presented.

#### A. Support Vector Machines (SVMs)

Support vector machines (SVMs) have been applied successfully to text classification. An SVM is a vector space based machine learning method where the aim is to find a decision frontier between two classes that are maximally far from any point in the training data. This classifier finds the optimal hyperplane to separate two or $N$ values from a variable according to featured space.

#### B. kNN

$k$ nearest neighbor or $kNN$ classification determines the decision frontier locally. For $k \in N$ in $kNN$, the region in the space where the $k$ is the closest neighbors to the keyphrase is considered. A test keyphrase $kf$ is expected to have the same class as the training dataset located in the local region surrounding $kf$. The cosine measure of distance is used to determine the similarity between the $k$ nearest neighbors. $kNN$ is considered a nonlinear classifier [13].

#### C. Decision Tree

The decision trees describe a set of hierarchical rules for implementing a decision structure. A decision tree has leaves and nodes, each leaf contains a response and each node contains class conditions. The trees are created recursively using a training dataset [14], [15].

#### D. Random Forest

Random forest are an ensemble model, which means that they use the results from many different models to calculate a response. In the case of random forests, several decision trees are created, and the response is calculated based on the outcome of all of the decision trees [14], [16]. An important advantage of the random forest is its ability to handle a very large number of input attributes, as well as its speed.

### IV. Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a computational model used in natural language processing, considered in its beginnings as a method for representing knowledge [17].

1https://wordnet.princeton.edu
LSA is considered an unsupervised dimensionality reduction tool, such as principal component analysis (PCA) [18].

LSA has its origin in an information retrieval technique called Latent Semantic Indexing (LSI) whose purpose is to reduce the size of an array of document terms using a linear algebra technique called Singular Value Decomposition (SVD). The difference with LSA is that it uses a word-context matrix. The context can be a word, a sentence, a paragraph, a document, a test, etc.

This technique is based on the principle that the words in the same semantic field tend to appear together or in similar contexts [19], [20]. Consequently, indexing of documents with similar contexts should be included by the words that appear in similar contexts, even if the document does not contain those words. LSA does not consider the linguistic structure of contexts, but rather the frequency and co-occurrence of terms.

V. PROPOSED METHOD

In this study, a method has been proposed for classifying keyphrases. First, a set of features is extracted from the corpus (See Table I). Next, it is stored in a matrix and LSA is used to reduce its dimensionality. Later, some classification techniques are applied to predict the classes of the keyphrases of the test dataset. Figure 1 shows the behavior of the proposed method with reduced dimensionality.

However, we consider two variants one without reducing dimensionality and the other with reduced dimensionality.

The implementation carries out keyphrases classification by the following steps.

1) Reading of data files. In the training dataset, the paragraphs and keyphrases with their classes are extracted from the ANN files.

2) Preprocessing training dataset. For each text and keyphrase extracted in the previous step, a preprocessing is performed that consists of eliminating the stopwords2 and punctuation marks. The intention of the preprocessing is to leave in each paragraph and keyphrase alphanumeric symbols, as well as, Greek symbols because they were detected in the keyphrases training set and are considered of importance within the system.

3) Create query matrixes and dictionary terms. Using the TfidfVectorizer3 function of the scikit-learn4 package, the set of paragraph extracts is converted to an array with tf-idf weight. This matrix is reduced in dimensionality by means of LSA. In addition, a dictionary with all the terms included in the documents is obtained.

4) Extraction of features. Using the set of keyphrases and paragraphs obtained and preprocessed, the following features are extracted and stored in the matrix.

- Vector representation of the keyphrase. For each keyphrase, a vector representation is obtained through a query to the matrix produced in the previous step.

- Contains capital letters. If the sentence contains only capital letters; the value of the feature is 1, otherwise, it is 0.

- Contains Greek symbols. If the sentence contains some Greek symbols; the value of the feature is 1, otherwise, it is 0.

- Number of words. A numeric value that indicates the terms that make up the keyphrase.

- Start and end position. A numeric value indicating the relative position of the keyphrase within the preprocessed paragraph.

- Vector PoS. The keyphrase is labeled using the pos_tag function of the NLTK5 package and then it is converted to a numeric vector.

- Context vector of the keyphrase. The position of the keyphrase is located in the paragraph and a word is extracted to the left and another word to the right, in case there is no word to the left or to the right a null value is placed. This string is converted to a vector, with information of the reduced matrix of the previous step, the obtained vector is placed as a feature.

- Vector PoS of the context. The context is labeled with pos_tag and converted to a numeric vector.

- Identifier of the word to the left and to the right. From the dictionary terms the numerical identifiers of the words are obtained, in case some term is null, a value of -1 is set.

- Sum of IDF. It is the sum of the IDF of all the terms that make up the keyphrase (see Equation 1).

\[
\sum_{i=1}^{n} IDF(w_i)
\]  

Where \( w_i \) is a term that makes up the keyphrase.

- Keyphrase has a semantic relation of synonymy or hyperonymy. From the synonymy patterns of Table II and the hyperonymy patterns of Table III, it is determined whether the keyphrase matches any pattern shown, the value of the feature is expressed as 0 or 1.

5) Creation of a reduced matrix using LSA. The numeric array obtained in the previous step is reduced in dimensionality to 50 columns using the TruncatedSVD6 function of the scikit-learn package.

6) Creation of the model. Python’s scikit-learn classifiers: Random Forest7, Nearest Neighbors8, Decision Trees9, and Support Vector Machines10 are applied to the numerical matrix and the reduced matrix with

---

2stopwords list available in: http://www.ranks.nl/stopwords
4http://scikit-learn.org
5http://www.nltk.org/api/nltk.tag.html
LSA. The classifiers was used with default values from the scikit-learn library.

7) Evaluation. The steps are performed until the extraction of features on the test dataset is complete. Later, the models created in the previous step are applied to this dataset to obtain the class predictions. And finally, the coincidences with respect to a gold-standard are evaluated. The evaluation metrics considered are Precision ($P$), Recall ($R$), and Measure-$F_1$ ($F_1$), see Eqs. 2, 3 and 4.

$$P(S) = \frac{\text{Number of relevant terms retrieved}}{\text{Number of terms retrieved}}$$ (2)

$$R(S) = \frac{\text{Number of relevant terms retrieved}}{\text{Number of relevant terms}}$$ (3)

$$F_1(S) = \frac{2}{\frac{1}{P(S)} + \frac{1}{R(S)}}$$ (4)

In Tables II and III, some patterns are shown, some extracted from literature and others proposed by the authors. $S_1$ and $S_2$ are input keyphrases.

VI. EXPERIMENTS AND RESULTS

The proposed method is evaluated with the gold-standard using the metrics of Precision, Recall, and Measure-$F_1$ (see Eqs 2, 3 and 4).

The datasets used and the results obtained by the proposed method with the two variants are presented below.

### TABLE I. LIST OF FEATURES

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keyphrase</td>
<td>Vector representation of the keyphrase</td>
</tr>
<tr>
<td></td>
<td>Contains capital letters</td>
</tr>
<tr>
<td></td>
<td>Contains Greek symbols</td>
</tr>
<tr>
<td></td>
<td>Number of words</td>
</tr>
<tr>
<td></td>
<td>Start and end position</td>
</tr>
<tr>
<td></td>
<td>Vector PoS</td>
</tr>
<tr>
<td>Context</td>
<td>Context vector of the keyphrase</td>
</tr>
<tr>
<td></td>
<td>Vector PoS of the Context</td>
</tr>
<tr>
<td></td>
<td>Identifier of the word to the left and to the right of the keyphrase</td>
</tr>
<tr>
<td>Measure</td>
<td>Sum of $IDF$</td>
</tr>
<tr>
<td>Semantic relation</td>
<td>Hyperonymy</td>
</tr>
<tr>
<td></td>
<td>Synonymy</td>
</tr>
</tbody>
</table>

### TABLE II. SYNONYMY PATTERNS

$S_1$ and $S_2$ are identified keyphrases

<table>
<thead>
<tr>
<th>No.</th>
<th>Pattern</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$S_1$ often referred to as $S_2$</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$S_1$ as known as $S_2$</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$S_1$ aka $S_2$</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$S_1$ usually abbreviated as $S_2$</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>$S_1$ also called $S_2$</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>$S_1$ is often referred to as $S_2$</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>$S_1$ is referred to as $S_2$</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>$S_1$ alias $S_2$</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>$S_1$ aka $S_2$</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>$S_1$ as known as $S_2$</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>$S_1$ is referred to as $S_2$</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>$S_1$ is called $S_2$</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>$S_1$ called as $S_2$</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>$S_1$ is frequently abbreviated as $S_2$</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>$S_1$ also called $S_2$</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>$S_1$ is usually called $S_2$</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>$S_1$ generally known as $S_2$</td>
<td></td>
</tr>
</tbody>
</table>

### TABLE III. HYPERONYMY PATTERNS [25], [11]

$S_1$ and $S_2$ are identified keyphrases

<table>
<thead>
<tr>
<th>No.</th>
<th>Pattern</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$S_1$ such as $S_2$</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$S_1$ also such as $S_2$</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$S_1$ and other $S_2$</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$S_1$ including $S_2$</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>$S_1$, especially $S_2$</td>
<td></td>
</tr>
</tbody>
</table>

A. Dataset

The corpus was provided by the organizers of Task 10 at Semeval 2017 and it contains 500 articles, in English language, of the free collection of the ScienceDirect website\(^\text{11}\). It consists of 500 journal articles distributed among the domains Computer Science, Material Sciences and Physics. The training dataset consists of 350 documents, 50 are kept for development and 100 for testing. The total vocabulary is 10503 words.

Three types of documents are provided by the organizers: plain text documents with a paragraph, ANN documents with notes for those paragraphs and keyphrases with the class identifier, and XML documents with the original article.

Table IV shows information about the training and test dataset. Table V shows information about classes.

B. Results

In the proposed method four classifiers were used: Random Forest, kNN, Decision Tree, and SVM. The models were

\(^{11}\)https://scienceie.github.io/resources.html
created with the classifiers of the scikit-learn tool of Python using the default values and considered the best result of ten test. The best result of the ten test without LSA are presented in Table VI, and for the second variant, in Table VII, the bold values are the high scores in measure $F_1$.

In Table VI of the first variant, the results show that the classifiers based on Random Forest, SVM and Decision Tree achieve competitive results compared to kNN. The best result is reported with the Random Forest classifier that achieves an average of 0.60 with the measure $F_1$. For the material category, the SVM classifier obtained 0.70 of $F_1$, in the case of the task category the Random Forest and Decision Tree classifiers achieved better results.

In the case of the kNN classifier, for both variants, the values of $k$, $k = 3, 4, 5, 7, 9$ and 12 were tested, but the results were maintained with the average results reported in tables with $k = 5$ nearest neighbors.

In Table VII, which corresponds to the second variant with reduction of dimensionality, it can be observed that the Random Forest classifier yielded better results in the evaluation, on the contrary, kNN obtained a lower performance, only obtaining 0.44 of $F_1$. For the material category, the Random Forest classifier obtained 0.65 of Measure-$F_1$, and, in the case of the task category, it also achieved good results. In the case of the LSA variant, several tests were performed varying the number of dimensions, however, the average results of all classifiers were maintained in the variations.

The tables show that Random Forest performs the best in both variants. Finally, Table VIII presents a performance comparison between the proposed variants and other works reported in [2].

According to the results obtained with $F_1$, it was observed that the first variant using the Random Forest classifier achieved a good performance, very close to the first two systems with the highest keyphrase classification scores. The variant without LSA, with an $F_1$ value of 0.60, exceeds at least the random result, as well as other systems, including the variant with LSA (with an $F_1$ of 0.57).

The LSA approach shows low performance due to the application method, LSA is a technique that associates the words of a document to place them in the same context. However, in this investigation, LSA was applied to decrease the dimensionality of the features matrix and a reduced set of user-friendly data was obtained, but without sharing a context, so the benefits of LSA were ignored and affected the performance; unlike the without LSA approach that considered all the established features to determine the boundary between the classes.

Nevertheless, the two approaches show that using the features only from the dataset without external resources get good results in the keyphrase classification task.

VII. CONCLUSIONS

A method of keyphrase classification, selected from articles published in scientific journals has been proposed here. The proposed method uses only the dataset provided by the task’s organizers without to query external resources.

The results show that Random Forest and Support Vector Machines obtain the highest score in the classification tasks using the full feature vectors. However, by applying LSA the performance drops perceptibly even when failing to detect keyphrases in a category, as shown by Support Vector Machines that can not classify any keyphrase of task class after applying LSA.

The Random Forest classifier shows high performance in both tests, this is due to the nature of the classifier, because it bases its predictions in a calculation between the predictions of the Decision Trees that comprise it. While the kNN classifier shows poor performance in both tests, due to it basing its predictions on measuring distances between neighboring vectors. The above causes it to be sensitive to the number of samples because a majority of them cause false predictions by only calculating distances between neighboring vectors. In addition, it is considered that the behavior of the data is in a linear way and this favors the results of Random Forest, Decision Tree, and SVM classifiers.

LSA shows a slight performance improvement in the recall measure to detect keyphrases of the task class with these same classifiers. But the application method penalizes the LSA technique benefits and get low performance in keyphrase classification. Therefore, we study if a variation in the application of the method can improve the performance.

For future research, to extend the research in the Spanish language will be seeked, in addition to including more features, among whose the Pointwise Mutual Information (PMI) is considered. PMI is a statistical measure that calculates the probability that two or more words will appear together. As well as the extension of the base of patterns of synonymy and hypernymy. In addition to the use of other classifiers, such as Naive Bayes and Neural Networks.

ACKNOWLEDGMENT

This work is supported by the Sectorial Research Fund for Education of Mexico with the CONACyT project 257357.

REFERENCES


**TABLE VIII.** VARIANTS COMPARISON OF THE PROPOSED METHOD WITH OTHER SYSTEMS

<table>
<thead>
<tr>
<th>Team</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>MayoNLP</td>
<td>0.67</td>
</tr>
<tr>
<td>UKIP/ELECTION</td>
<td>0.66</td>
</tr>
<tr>
<td>Random Forest - without LSA</td>
<td>0.60</td>
</tr>
<tr>
<td>SVM - without LSA</td>
<td>0.57</td>
</tr>
<tr>
<td>Decision Tree - with LSA</td>
<td>0.57</td>
</tr>
<tr>
<td>Decision Tree - with LSA</td>
<td>0.52</td>
</tr>
<tr>
<td>LABDA</td>
<td>0.51</td>
</tr>
<tr>
<td>BUAP</td>
<td>0.45</td>
</tr>
<tr>
<td>Random</td>
<td>0.23</td>
</tr>
</tbody>
</table>


Efficient fine-grained road segmentation using superpixel-based CNN and CRF models

1st Farnoush Zohourian  
dep. Computer Science and Applied Cognitive Science  
University of Duisburg-Essen  
Duisburg, Germany  
farnoush.zohourian@stud.uni-due.de

2nd Jan Siegemund  
Delphi Electronics & Safety  
Delphi  
Wuppertal, Germany  
jan.siegemund@delphi.com

3rd Mirko Meuter  
Delphi Electronics & Safety  
Delphi  
Wuppertal, Germany  
Mirko.Meuter@delphi.com

4th Josef Pauli  
dep. Computer Science and Applied Cognitive Science  
University of Duisburg-Essen  
Duisburg, Germany  
josef.pauli@uni-due.de

Abstract—Towards a safe and comfortable driving, road scene segmentation is a rudimentary problem in camera-based advanced driver assistance systems (ADAS). Despite of the great achievement of Convolutional Neural Networks (CNN) for semantic segmentation task, the high computational efforts of CNN-based methods is still a challenging area. In recent work, we proposed a novel approach to utilize the advantages of CNN's for the task of road segmentation at reasonable computational effort. The runtime benefits from using irregular superpixels as basis for the input for the CNN rather than the image grid, which tremendously reduces the input size. Although, this method achieved remarkable low computational time in both training and testing phases, the lower resolution of the superpixel domain yields naturally lower accuracy compared to high cost state of the art methods. In this work, we focus on an refinement of the road segmentation utilizing a Conditional Random Field (CRF). The refinement procedure is limited to the superpixels touching the predicted road boundary to keep the additional computational effort low. Reducing the input to the super-pixel domain allows the CNN's structure to stay small and efficient to compute while keeping the advantage of convolutional layers and makes them eligible for ADAS. Applying CRF compensates the trade-off between accuracy and computational efficiency. The proposed system obtains comparable performance among the top-performing algorithms on the KITTI road benchmark and it's fast inference makes it particularly suitable for real-time applications.

Index Terms—Super-pixel, Semantic Segmentation, CNN, Deep learning, CRF, Road Segmentation

I. INTRODUCTION

Development of intelligent vehicle and advanced driver assistance systems (ADAS) is one of the most active research areas and has attracted much attention recently. Camera-based perception of the drivable road environment is an important problem in this context. The variations in the illumination and appearance, apart from occlusions are challenging issues that make accurate road segmentation a difficult task. One solution to this is semantic segmentation, that assigns each pixel a category label. The breakthrough techniques such as deep convolutional neural network (CNN) models [Krizhevsky et al., 2012], [Simonyan and Zisserman, 2014] significantly improved pixel-wise semantic segmentation tasks by extracting rich hierarchical features [Long et al., 2015], [Lin et al., 2016]. However, fast and accurate estimation of the pixel labels in a way compatible for embedding into real-time application is not an easy and straightforward task. While accuracy of recent approaches is increased by creating deeper networks with as many layers as possible [Simonyan and Zisserman, 2014], [He et al., 2016], in practice, most of them are fairly limited in computational power and memory. In our previous work [Zohourian et al., ], we proposed a novel approach to utilize the advantages of CNN’s for the task of road segmentation at suitable computational effort. This method mainly differs from usual semantic segmentation methods in two aspects: first the input data model provided for the CNN network and second the simple CNN- network layering. The state of the art convolutional networks for image segmentation are based on two different input data model: They are based on either patch-wise or taking full image resolution. Most recent improvement in CNN are accomplished by using above input data mode and increasing the network size which require powerful GPUs. As deeper networks provoke large computational costs they are generally not suitable for embedded devices in self-driving cars and ADAS. In our work, the runtime benefits from using few layers and irregular superpixels as basis for the input for the CNN rather than regular Patch or full image, which tremendously reduces the input size. This strategy disassembles the pixel grid into superpixels forming the basic units for a pre-classification via a CNN. Reducing the input to the super-pixel domain allows the CNN’s structure to stay small and efficient to compute while keeping the advantage of convolutional layers. Although, this method achieved remarkable low computational time in both training and testing phases, the lower resolution of the superpixel domain yields naturally lower accuracy compared to high cost state of the art methods.
The current work proposes a model strategy to refine the classification result from superpixel grid to pixel grid using Conditional Random Fields (CRFs) [Lafferty et al., 2001]. CRFs can model global properties like object connectivity, geometric properties and spatial relationship between objects (See Fig. 1). This work comprises two aspects for coupling local and global evidences. We combine the local image classification information extracted from CNN part with global information of neighboring pixel relations to decide for an accurate pixel label. The key idea of CRF inference for semantic labeling is to formulate the label assignment problem as a probabilistic inference problem that incorporates assumptions such as the label agreement between similar pixels or image regions. This idea follows largely previous work by applying CRF technique on CNN as a Post-processing step [Chen et al., 2016]. Our segmentation algorithm follows a number of steps which we briefly summarize here:

(a) segmenting the image into super-pixels, wherein the super-pixels are coherent image regions comprising a plurality of pixels having similar image features.
(b) determining image descriptors for the super-pixels, wherein each image descriptor comprises a plurality of image features.
(c) The super-pixels are assigned to corresponding positions of a regular grid structure extending across the image in order to create neighborhood relations for convolutional purpose.
(d) This lattice together with the image descriptors are fed to the convolutional network based on the assignment to classify the super-pixels of the image according to semantic categories.
(e) An optimization strategy based on CRF refines the superpixel-based results to a pixel-based result to increase the precision of road segmentation.

Steps a to d are done in our previous work and the current work discusses the step e (See Fig. 2). To keep the advantage in computation time, we limit the refinement scope to the superpixels bordered to street boundary estimated in the first step, i.e. neighboring superpixels assigned to a different class by the CNN result. Using CRF compensate the trade-off between accuracy and computational cost. The proposed system obtained comparable performance among the top-performing algorithms on the KITTI [Fritsch et al., 2013] road benchmark and its fast inference makes it particularly suitable for deployment in ADAS.

II. METHODOLOGY

In this paper, we combine both CRF and CNN into a framework to get a fine-grained road segmentation from urban scene images. Figure 2 displays the architecture of our method. First, we briefly explain our previously proposed method that provided a rational feature model fed into designed CNN network to segment road regions. The input data model is a combination of a higher dimensional feature space with irregular superpixel segmentations projected on a regular lattice structure, for convolutional purpose [Zohourian et al., ] Afterwards, we explain how we improve the segmentation results by applying a CRF technique. The refinement procedure is limited to the superpixels touching the predicted road boundary. Restricting to this area helps to enhance the segmentation accuracy while keeping the additional computational effort low. To obtain the highest performance, we evaluated three different CRF techniques. The individual steps are presented in the next discussions.

A. SUPERPIXEL-BASED CONVOLUTIONAL NEURAL NETWORK

Superpixel segmentation is local grouping of pixels on the pixel grid, based on features like color, brightness, texture, etc [Ren and Malik, 2003]. Compared to pixel units, superpixel units store more compact information that can greatly reduce the model complexity and computation cost especially for real time systems. Well-segmented superpixels preserve the object structures and adjust well to the object contours, that causes the accuracy improvement of subsequent tasks like semantic segmentation. To convolve the input data with kernels in convolutional layers of CNN, we need a regular structure (grid format). Irregular superpixels with different sizes or disordered shaped boundaries are not directly convolable, due to arbitrary neighborhood relations. On the other hand, enforcing a distinct topological structure of superpixel segmentation mostly prevent the maximum pattern homogeneity inside each superpixel. In [Zohourian et al., ], we proposed a novel approach to have
both maximum homogeneity and convolutional ability. First, we segment the image into coherent superpixels, comprising a plurality of pixels having similar image features. Then, for each irregular superpixel, an image descriptor is defined, which comprises a plurality of image features. The necessity of having a regular topology to be able to convolve the input data with kernels, motivated us to propose a superpixel lattice projection. The superpixels are projected into corresponding regular grid structure extending across the image. Finally, this lattice together with the image descriptors are fed to a convolutional neural network for pixel-wise classification purpose.

1) Input Data Model:
For superpixel segmentation, we used a modified version of SLIC algorithm [Achanta et al., 2010]. SLIC is adopted k-means clustering for grouping of pixels in the 5-D space defined by 3-D spectral space and 2-D spatial space. To enforce all superpixels to be connected and to avoid single isolated superpixels, SLIC applies an “Enforce-Connectivity” procedure which leads to non constant numbers of created superpixels in each iteration of adopted K-mean clustering making them unsuitable as direct CNN input model. To prevent this problem, in our modified version, we do not remove any small region with a single connected component. First, all label-connected components and adjacent superpixels in 2-D for each superpixel should be computed. Then for each superpixel with more than one segment and same label, we keep the larger one and merge the rest into the nearest superpixel. The nearest neighborhood is selected based on the euclidean distance between the center of sub-segment to the center of each adjacent segment. Now, we define our lattice centered in the rectangular structure extracted from the first iteration of SLIC method projected to the corresponding irregular superpixel from final step.
For having model relevant object characteristics in the image that are non-redundant and informative enough and facilitate the subsequent learning and generalization steps, we define for each superpixel a high dimensional feature descriptor. Each of the image descriptors comprises 69 image features consisting of 9 different color channels, 1 position and 59 Local Binary patterns(LBP). This provides high accuracy and reliability. The provided data model is fed to a simple convolutional network presented in the following.

2) CNN Network Architecture:
Due to our input data model with informative structure coming from superpixels and feature descriptor, our proposed method does not require a complex network architecture to handle large image context which leads to a considerable reduction of computational time. The network has two convolutional layers, two fully connected layers and one drop-out layer with non-linear activation function after each convolutional and fully connected layer. The input of our method is defined by the superpixel lattice on each image with size of $H/S$ and $W/S$, where $S$ is the initial superpixel size and $W,H$ are image width and height. The output is a set of three numbers to indicate which of the three classes of the road, non-road or un-labeled they belong to [Zohourian et al., ].

B. Segmentation Refinement with CRF
Our superpixel-based convolutional network already has the capability of modeling global relationships within a scene and to adaptively coarse segmentation by capturing the local properties such as object shape and contextual information. However, CNNs have their shortcomings to model the interactions and correlation between the output variables directly, which is important for a smooth semantic segmentation. To this end, graphical models like Conditional Random Fields (CRFs) have been used to impose consistency and coherency between labels. CRFs can model global properties like object connectivity, geometric properties and spatial relationship between objects. By combining CNN and CRF models, we are able to fine-tune the CNN segmentation results specially along the road border. CRFs are formulated in a probabilistic framework, modeling the joint probability of the image and its corresponding labels. We represented CRF in terms of energy minimization, where the energy function has two terms: one term penalizes solutions that are inconsistent with the observed data, called Unary term ($\varphi$), while the other term enforces some kind of spatial coherence, called pairwise term ($\psi$) (See Eq.1).

$$E(l, I) = \alpha \sum_i \varphi(l_i|x_i) + \beta \sum_{ij} \psi(l_i, l_j|x_i, x_j)$$

where $l_i$ is corresponding class label, $I$ is $\{x_1, ..., x_N\}$, and $x_i$ is the pixel intensity. The unary potential is the inverse likelihood of each pixel getting a particular label. The pairwise part encodes the neighboring information and computes the difference between a pixel label from its neighbor.
In this work, we evaluated three inference methods for pixel-wise image segmentation, considering the labels of the image as hidden states and solving the label prediction as a solution of the maximum posterior probability (MAP) [Nowozin et al., 2011]. We investigate quality of all three methods for classification and study the influence of computational gain and overall accuracy, allowing for robust and accurate statistical analysis on road segmentation. Image data is used to build the CRF that potential functions which are applied directly on image pixels belong to the super-pixels touching the road boundary. In all three inference methods, the respective unaries are supplied by CNN. The three inference methods are:

1) Iterated Conditional Modes (ICM):
Iterated Conditional Modes (ICM) [Besag, 1986] is an iterative algorithm that employs a deterministic greedy strategy to find a local minimum by minimizing an energy function (See Algorithm (1)). We initiated ICM by projecting the CNN prediction from superpixel to pixel grid. We smooth out the initial segmentation by assigning the new label to each pixel. This process is repeated until convergence. To estimate appropriate values for the variances of the pixel intensity levels, forming the unary term, we used Gaussian Mixture Model [Williams
and Rasmussen, 1996]. GMM has 3 components belonging to either road and non-road classes and is modeled as two separate Gaussian mixture models. Unary term in formula 1 is extended to \( \sum_i \varphi(l_i, k_i, \theta|x_i) \) Where \( k_i \in \{1, ..., K\} \) are the Gaussian mixture components, \( k \) is number of classes and \( \theta \) is GMM Model parameters. GMM parameters are learned by K-mean method as an iterative Expectation-Maximization (EM) algorithm.

In binary term, the neighborhood is based on 4-neighbors in pixel grid. For each class, we compute the sum of the labels which are not equal to the current pixel label and penalize the total energy of each class with some threshold value.

**Algorithm 1: Updated ICM**

<table>
<thead>
<tr>
<th>ICM ((I, L, T, MaxItr))</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong>: pixels contained in superpixels surrounding the road border ((I)), corresponding labels from CNN ((L)), Potential threshold ((T)), maximum number of iterations ((MaxItr))</td>
</tr>
<tr>
<td><strong>Output</strong>: smooth road segmentation</td>
</tr>
<tr>
<td>Initialize Mixture Models with CNN outputs // GMM</td>
</tr>
<tr>
<td>Assign GMM components</td>
</tr>
<tr>
<td>( K_i = \arg \min_k \varphi(l_i, k_i, \theta</td>
</tr>
<tr>
<td>Learn GMM parameters</td>
</tr>
<tr>
<td>( \theta = \arg \min \sum_i \varphi(l_i, k_i, \theta</td>
</tr>
<tr>
<td>Calculate unary energy // Unary part</td>
</tr>
<tr>
<td>( \varphi(l_i, k_i, \theta</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>while not MaxItr do</td>
</tr>
<tr>
<td>Calculate pairwise energy // Pairwise part</td>
</tr>
<tr>
<td>( \psi(l_i, l_j</td>
</tr>
<tr>
<td>Predict labels</td>
</tr>
<tr>
<td>( l = \arg \min \mathbf{E}(l, k, \theta</td>
</tr>
<tr>
<td>end</td>
</tr>
</tbody>
</table>

2) **Loopy Belief Propagation (LBP):**

For a more accurate message passing, we apply Loopy Belief Propagation (LBP) [Murphy et al., 1999] on the same energy term. We solved LBP by min-Sum algorithm [Kschischang, 1999]. As for ICM method, the input to our LBP method are all pixels that belong to the super pixels which touch the road border with corresponding label estimated from CNN output. The efficient approximate inference requires the Gaussian kernels computed over elements. The Gaussian Mixture Model with 3 components for each road and non-road classes is used. LBP is an iterative algorithm and will terminate if the changes in energy drops below a threshold or after a fixed number of iterations. At each iteration messages are passed around the MRF model. Our choice of the message passing scheme is right, left, up and down. Once LBP iteration completes, the label with highest energy is assigned to each pixel.

**Algorithm 2: Modified LBP (Min-Sum)**

<table>
<thead>
<tr>
<th>LBP ((I, L, T, MaxItr))</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong>: pixels surrounded the road border ((I)), corresponding labels from CNN ((L)), convergence tolerance ((T)), maximum number of iterations ((MaxItr))</td>
</tr>
<tr>
<td><strong>Output</strong>: approximate MAP labeling</td>
</tr>
<tr>
<td>Initialize Mixture Models with CNN outputs; // GMM</td>
</tr>
<tr>
<td>Assign GMM components (see Algorithm 1)</td>
</tr>
<tr>
<td>while not MaxItr do</td>
</tr>
<tr>
<td>Update Message: // Min-Sum method</td>
</tr>
<tr>
<td>( msg_{i \rightarrow j}(l) = \min_{l' \in \text{all labels}} \left[ \text{DataCost}(x_i, l') + \text{smoothnessCost}(l, l') + \sum_{k=\text{neighbors of } i \text{ expect } j} \text{msg}_{k \rightarrow i}(l') \right] )</td>
</tr>
<tr>
<td>where ( msg_{i \rightarrow j}(l) ) is the message from node ( i ) to node ( j ) for label ( l ). Calculate Belief</td>
</tr>
<tr>
<td>( \text{Belief}(x_i = l) = \text{DataCost}(x_i, l) + \sum_{k=\text{neighbors of } i} \text{msg}_{k \rightarrow i}(l) )</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>MAP assignment and calculating the energy:</td>
</tr>
</tbody>
</table>

3) **Dense CRF with Gaussian edge potentials:**

Contrary to the previous inference methods, we used different energy term here. The unary potential is obtained by the class conditional probability map obtained from soft-max function of CNN network for those superpixels that touched the road border. This is given to the fully-connected CRF proposed by [Krähenbühl and Koltun, 2011] for the pixel-wise labeling. The pairwise potentials are usually modeled based on the relationship among neighboring pixels and weighted by color similarity, whereas dense CRF model considers long range interactions among pixels instead of just neighboring information by using a fully-connected graph, where all pairs of image pixels, \( i, j \) are connected together.

The pairwise potential is defined based on formula 2 where the first term depends on both pixel positions \((P_i \text{ and } P_j)\) and pixel color intensities \((I_i \text{ and } I_j)\) and the second term only depends on pixel positions. \( \omega_m (m = \{1, 2\}) \) are linear combination weights. The terms are defined as Gaussian kernels whereby, first represents the color-similarity between nearby pixels and the latter removes all small isolated regions. The degrees of nearness and similarity are controlled by parameters \( \sigma_\alpha \) and \( \sigma_\beta \) and the size of small areas is thresholded by \( \sigma_\gamma \). The algorithm uses the mean-field approximation and a message passing scheme in a fully-connected graph to efficiently infer the latent variables that approximately minimize the Gibbs energy of a labeling [Krähenbühl and Koltun, 2011].

\[
\psi_{i,j} = \omega_1 \exp(-\frac{\|P_i - P_j\|^2}{2\sigma^2_\alpha}) - \frac{\|I_i - I_j\|^2}{2\sigma^2_\beta} + \omega_2 \exp(-\frac{\|P_i - P_j\|^2}{2\sigma^2_\gamma})
\]
III. EXPERIMENTS AND RESULTS

We evaluate our method on public KITTI [Fritsch et al., 2013] dataset comprising urban scenarios. KITTI comprises 502 8-bits RGB images splits in train, validation and test sets with ground truth label for three semantic classes. The training set has 289 images (95 images with urban markings (UM), 96 images with multiple urban markings (UMM) and 98 images where the street has no urban markings (UU)). The test set has 290 images including (96 UM, 94 UMM and 100 UU) images. The image dimensions are different in the width lying in [1226, 1238, 1241, 1242] and the height in [370, 374, 375, 376]. We selected 20% of training set images from 3 different categories UM, UMM, UU for the validation set. These images completely originate from different video sequences which are not part of the training set.

For having a fair comparison between our current approach and our previous work [Zohourian et al., ], we use the same conditions and parameter values. We used the hardware specifications, CPU: Intel(R) Core(TM) i7-4790K @ 4GHz for both training and testing. In [Zohourian et al., ] we used SLIC parameters $K = 400, m = 35$ resulting in 396 super pixels in each image projected to a $11 \times 36$ lattice for CNN input. Our original task was to segment road from non-road (background). We evaluated the accuracy once on pixel and once on superpixel level. For evaluation in the superpixel domain, the ground truth for each Super-pixel is defined based on the majority pixel-labels inside the superpixel. In current work, We evaluate the performance of the proposed approach implemented in three different CRF techniques with the accuracy of the pixel grid obtained from superpixel-based convolutional network [Zohourian et al., ] based on the image perspective and a birds eye projection provided by KITTI dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>ACC</th>
<th>F1</th>
<th>PRE</th>
<th>REC</th>
<th>FPR</th>
<th>FNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN classifier</td>
<td>94.41%</td>
<td>95.18%</td>
<td>91.41%</td>
<td>97.14%</td>
<td>8.59%</td>
<td>2.60%</td>
</tr>
<tr>
<td>CRF-ICM</td>
<td>84.31%</td>
<td>87.55%</td>
<td>80.49%</td>
<td>86.78%</td>
<td>9.51%</td>
<td>4.92%</td>
</tr>
<tr>
<td>CRF-Sequential</td>
<td>90.19%</td>
<td>91.94%</td>
<td>83.05%</td>
<td>93.62%</td>
<td>16.95%</td>
<td>8.09%</td>
</tr>
<tr>
<td>CRF-Meanfield</td>
<td>90.85%</td>
<td>90.94%</td>
<td>92.30%</td>
<td>90.05%</td>
<td>7.60%</td>
<td>2.10%</td>
</tr>
</tbody>
</table>

**TABLE I**

EVALUATION RESULTS ON KITTI VALIDATION SET BEFORE AND AFTER APPLYING CRF TECHNIQUES

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>MaxP</th>
<th>AP</th>
<th>PRE</th>
<th>REC</th>
<th>FPR</th>
<th>FNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>UU:ROAD</td>
<td>83.22%</td>
<td>72.94%</td>
<td>77.11%</td>
<td>90.39%</td>
<td>12.23%</td>
<td>9.61%</td>
</tr>
<tr>
<td>UMM:ROAD</td>
<td>90.99%</td>
<td>84.62%</td>
<td>87.88%</td>
<td>94.29%</td>
<td>15.22%</td>
<td>3.71%</td>
</tr>
<tr>
<td>UU:ROAD</td>
<td>80.02%</td>
<td>67.95%</td>
<td>77.56%</td>
<td>82.64%</td>
<td>17.79%</td>
<td>17.56%</td>
</tr>
<tr>
<td>URBAN:ROAD</td>
<td>85.37%</td>
<td>77.81%</td>
<td>82.08%</td>
<td>90.11%</td>
<td>10.89%</td>
<td>5.09%</td>
</tr>
</tbody>
</table>

**TABLE II**

EVALUATION RESULTS ON KITTI TEST SET BASED ON MEAN-FIELD CRF

For evaluation in birds-eye perspective in the KITTI benchmark the images are projected on the ground plane via the known camera geometry. We had an improvement of approximately 5% on official KITTI Test set compared to our CNN classifier without CRF refinement (See Table III). We could fix the weaker accuracy in the birds eye view projection compared to the image perspective evaluation, which appeared due to spreads of the error induced by inaccurate superpixels on the road border over a much larger region [Zohourian et al., ]. Table II shows the results which are split into the different road types (UM, UMM, UU, URBAN). Two samples in BEV are shown in Figure 4. Whilst the street is nicely segmented, there are a few false detections that mostly happened when the segmented area was fooled by a shadow covering the street. Moreover, compared to the obtained results from validation set we still have lower average accuracy on KITTI test set, which contains more challenging and diverse scenarios.
Future work will initially focus to improve the discrimination of the road pattern in challenging conditions; such as shadow on road surface, illumination changes or similarity with neighboring patterns like sidewalks. It is also promising to evaluate this approach for more than 2 classes and extend the pixel-wise classification to different objects such as sidewalks, lane, traffic sign, vehicles, buildings, sky, etc.

REFERENCES


Fig. 4. Road segmentation result from official Kitti test set in baseline and bird eye view perspectives. Here, red denotes false negatives, blue is false positives and green represents true positives.
Image Classification using Collaborative Mean Attraction with Sparse Optimization

Hiroki Oghihara
Graduate School of Engineering,
University of Miyazaki
Miyazaki, Japan
hm13015@student.miyazaki-u.ac.jp

Masayuki Mukunoki
Department of Engineering,
University of Miyazaki
Miyazaki, Japan
mukunoki@cs.miyazaki-u.ac.jp

Abstract—In this paper, we propose a new method (SCMA method) which incorporates sparse optimization into Collaborative Mean Attraction (CMA) method. Since the CMA method uses all training images for classification, it has an advantage that it can accurately classify the test images even when the number of training data in each class is small. However, the CMA method has a risk of overfitting the training data. By introducing sparse optimization to the CMA method, we expect that only appropriate training images will be selected and used for the classification. We implement the SCMA method and conduct comparative experiments with several other methods using the Caltech256 dataset. The experimental results show that the SCMA method can achieve better image classification results even when the variation of test image in each class is large.

Index Terms—General Object Recognition, ImageNet, Caffe, Caltech Dataset, Sparse Optimization, Small number of training images

I. INTRODUCTION

Image classification technique shows great progress in recent years. The progress is mainly owing to Deep Learning. In Deep Learning, convolutional neural network is trained by huge number of training images and it can obtain the ability to extract excellent general purpose image features. Using the extracted features, we can achieve good performance for many kinds of image understanding tasks, such as object recognition, object extraction and image classification.

In image classification, there are some cases where a relatively small number of training images for each class can be used or where it is preferable that the number of training images should be small. For example, there is a case that a user may index his/her personal photos based on his/her personal viewpoint. In this case, it is preferable that the user can automatically obtain the classification result by giving only a small number of sample images for each class, since the cost for the indexing is very low. In such image classification, we need a classifier that can learn the classes from a relatively small number of training images.

The Collaborative Mean Attraction (CMA) method [1] [2] is one of effective methods for classifying images with a small number of training images. This method has been proposed for person re-identification at first. In person re-identification, a person taken by a video camera (camera A) is identified among the persons taken by another video camera (camera B). In this case, the person in camera A is test data and the persons in camera B is a known categories (classes), and we can use only a few frames (images) observed while the person passes through the camera view area. The CMA method has shown high performance for the person re-identification [1]. We have also applied the CMA method to general image categorization [2]. The experimental results have shown that the CMA method achieves high performance even in general image categorization, especially when the number of training images is small.

The CMA method classifies test data using the entire training images. This method consists of two stages, that is, an optimization stage and a classification stage. In the optimization stage, it generates a representative point from the test data. It also generates an approximate point of the representative point using all training data in all classes. In the classification stage, it evaluates the contribution of training data in each class for generating the approximate point, and the test data is classified into the class with the highest contribution. Since the CMA method uses the entire training data for generating the approximate point, it shows high performance when the number of images in each class is small. However, it need to evaluate the contribution of each class for classification. In some cases, the contribution of each class does not differ significantly among classes. It is because we try to use all training data for generating the approximate point.

In this paper, we propose the Sparse Collaborative Mean Attraction (SCMA) method. The SCMA method uses sparse optimization at the optimization stage of the CMA method. Since sparse optimization tends to select only a few important data and to generate concentrated information, we expect that the SCMA method can evaluate the contribution of each class more clearly. As the result, we can obtain better performance according to Occam’s razor.

This study is funded by KAKENHI(C) 17K00240, Japan.
We implement the SCMA method and compare it with CMA, Support Vector Machine (SVM) and some other methods by applying to the Caltech256 data set. From the results, we investigate the characteristics of the SCMA method.

II. CMA METHOD

For better understanding of the SCMA method, we firstly introduce the original CMA method [2] in this section.

The CMA method is an identification method that classifies unknown test data into one of known categories. Known categories are given as the feature sets of one or more training images belonging to that category. The test data is also given as the feature of the test image, but it is possible to give a set of test images as one group and classify the group into one of the known categories at once.

The CMA method consists of two stages, that is, an optimization stage and a classification stage (Fig.1). In the optimization stage, we generate a representative of the test data and approximate it with all the training images in all classes. In the classification stage, we select the category that most contributes to this approximation.

A. Optimization stage

A representative point and its approximate point are expressed by linear sums of test data and training data, respectively. In the optimization stage, the coefficients of the linear sum constituting the representative point and the approximate point are obtained.

The coefficients are determined so that the distance between the representative point and the approximate point becomes short. At the same time, constraints are imposed that the representative point and the approximate point should be close to the average of each data so that the properties of each data are retained.

A test data matrix \( Q \in \mathbb{R}^{m \times N_q} \) and a training data matrix \( X_i \in \mathbb{R}^{m \times N_i} (i \in \{ 1, \ldots, n \}) \) are given, where \( m \) is the number of dimensions of the feature vector, \( N_q \) is the number of the test images, \( n \) is the number of known categories, \( N_i \) is the number of training images in the \( i \)-th category. We denote the matrix that concatenates all the training images as \( X = (X_1, X_2, \ldots, X_n) \in \mathbb{R}^{m \times N_x} \), where \( N_x = \sum_i N_i^x \) is the total number of training images.

Here, we obtain the coefficient vector \( \alpha, \beta \in \mathbb{R}^m \) by minimizing the following expression:

\[
\arg \min_{\alpha, \beta} f(\alpha, \beta)
\]

\[
f(\alpha, \beta) = \| Q \alpha - X \beta \|_2^2 + \lambda_1 \| \alpha - \frac{1}{N_q} \|_2^2 + \lambda_2 \| \beta - \frac{1}{N_x} \|_2^2
\]

where \( 1_k \) is the \( k \)-dimensional vector with each element 1, \( \| \cdot \|_2 \) is the \( L_2 \) norm of the vector. \( Q \alpha \) and \( X \beta \) in (1) are the representative point and the approximate point, respectively.

The first term is a term for minimizing the distance between the representative point and the approximate point. The second term and the third term are constraint terms (regularization terms) for bringing the representative point and the approximate point closer to their respective average points. \( \lambda_1 \) and \( \lambda_2 \) are the weight parameters of the regularization terms.

B. Classification stage

Using the coefficients \( \alpha \) and \( \beta \) obtained in the optimization stage, we find the category that most contributes to constructing the approximation point, and classify the test data into that category. Each dimension of the coefficient \( \beta \) corresponds to a training image. \( \beta \) can be decomposed into \( \beta_i \) as coefficients of the training images in the \( i \)-th category. Thus, we calculate the following formula for each category \( i \) and we classify the test data into the category with the minimum \( d^i \).

\[
d^i = (\| Q \|_* + \| X_i \|_*) \cdot \| Q \alpha - X_i \beta_i \|_2^2 / \| \beta_i \|_2
\]

The smaller \( \| Q \alpha - X_i \beta_i \|_2^2 \) is, the better the representative point can be approximated with only the \( i \)-th category. Similarly, when the value of \( \| \beta \|_2 / \| \beta_i \|_2 \) is larger, the coefficients of the \( i \)-th category play a large role in the all known categories. \( \| \cdot \|_* \) is the nuclear norm of the matrix (sum of singular values of the matrix). The larger the variation of the data, the larger the nuclear norm becomes. It is used for weighting the variance of categories.

III. SCMA METHOD

The SCMA method follows the same idea as the CMA method. It also consists of an optimization stage and a classification stage. In the optimization stage, it generates a representative point by linear combination of test data. It also generates an approximation point of the representative point by linear combination of entire training data. The main difference between the SCMA and CMA method is that the SCMA method uses sparse optimization for calculating the coefficients of the linear combinations. In the classification stage, the SCMA method uses a simpler formula than that of the CMA method, since we expect that the contribution of each class will be clearer with the SCMA method.
A. Optimization stage

In the SCMA method, we use sparse optimization for generating a representative point and its approximate point. Sparse optimization is achieved by introducing the following minimization expression:

\[
\arg \min_{\alpha, \beta} f(\alpha, \beta)
\]

\[
f(\alpha, \beta) = \| (Q - \bar{Q}) \alpha + \bar{q} - (X - \bar{X}) \beta + \bar{x} \|_2^2 + \lambda \| \alpha \|_1 + \lambda \| \beta \|_1, \tag{3}
\]

where \( q \) and \( x \) are mean vector of training image and test image, respectively. \( Q \) and \( X \) are matrix obtained by arranging \( q \) and \( x \), \( N_q \) times. \( (Q - \bar{Q}) \alpha + \bar{q} \) and \( (X - \bar{X}) \beta + \bar{x} \) represent the representative point and the approximate point, respectively.

The first term is a term for making the representative point close to the approximate point and keeping them close to the average point of each data, in order to maintain the property of data. For sparse optimization, the zero elements in the coefficients \( \alpha \) and \( \beta \) do not affect to express the points, i.e. when \( \alpha \) and \( \beta \) are zero vectors, the points must be the average of test data and entire training data, respectively. For that purpose, the data array \( Q \) and \( X \) are normalized by \( Q, \bar{X}, \bar{q} \) and \( \bar{x} \) so that the average of the data will be zero vector.

The second and third terms are the \( L_1 \) norm regularization term. \( \lambda \) is the coefficient of the regularization term. FISTA [3] [4] is used to solve this optimization problem.

B. Classification stage

In the classification stage, we evaluate the contribution of each class for generating the approximate point. Since we expect that the SCMA method shows the contribution shaper than the CMA method, we employ simpler formulation in this stage.

Let \( \beta_i = (\beta_{i1}, \ldots, \beta_{iN_i}) \) be the coefficient of the training image of class \( i \). We calculate the following expression for each class \( i \) and classify the test data into a class with the minimum value.

\[
d^i = -\sum_{j=1}^{N_i} |\beta_{ij}| \tag{4}
\]

Larger \( |\beta_{ij}| \) means larger contribution. In the CMA and other comparison methods, \( d^i \) means distance, that is, larger value means less contribution. Therefore, we make the sum of \( |\beta_{ij}| \) negative for making the evaluation uniform.

IV. IMAGE CLASSIFICATION

A. Data set

We used the Caltech256 dataset [5] to evaluate our method. The Caltech dataset is a public dataset created at the California Institute of Technology. Caltech256 contains 30,607 images in total. They belong to one of 257 classes. The number of images in each class is more than 80. Caltech256 is a relatively large data set, which is difficult for classification. In the experiment of this paper, we use all 257 categories of Caltech256 dataset.

B. Feature extraction

For feature extraction, we use the model that trained by ImageNet images [6] using Caffe Deep Learning framework [7]. This model is constructed using 1.2 million training images in 1,000 categories of ILSVRC2012. The pre-trained model is publically available as bvlc_reference_caffenet [8]. We give an image to this model and use the output of the hidden layer (fc7) nodes as the feature of the image. The dimension of the feature vector is 4096.

C. Comparison method

We employ SVM, CPD, MPD and CMA for comparison methods. In SVM(Support Vector Machine), a linear kernel is used, and multi-class classification is carried out by the one-to-one classification method. For CPD (Center Point Distance), the distance between the center point of a category and the center point of test data is taken as the distance between the category and the test data. For MPD (Minimum Point Distance), the minimum distance between all training data of a category and all test data is taken as the distance. For CPD and MPD, test data is classified into a category that has the minimum distance with the test data.

V. EXPERIMENT

A. Single test image experiment

In this section, we conduct an experiment in the case where one test data consists of one image. We investigate how the classification results change when the number \( k \) of training images has been changed. We randomly extract \( k \) images from each 257 categories of Caltech256 as training images, extract features and make them as training data. All remaining images are used as evaluation data. A test data is extracted one by one from the evaluation data and classified. After determining the accuracy of classification against the evaluation data for each category, we calculate the overall accuracy by averaging them for all categories. The number of training images is \( k = 1, 2, 4, 8, 16, 30 \). In order to decrease the influence of the training image selection, we conduct the experiment for 8 kinds of training image selection for each \( k \) and the average of the overall accuracy is calculated as the experimental result for each value of \( k \). Between the SCMA method and the comparison methods, the same 8 kinds of training image selection are used so that the conditions will be the same. The results are shown in Fig.2 and Table I.

Experimental results show that the SCMA method shows higher accuracy than any other comparison methods in any \( k \). When the number of training images is small (\( k = 1, 2, 4 \)), the SCMA and CMA methods show better performance than SVM, CPD and MPD methods. Difference between the SCMA and CMA methods is small in this case. In the range of training images \( k = 8, 16, 30 \), the SCMA method shows higher accuracy than CMA and SVM methods. However, the difference is about 1%. So, the superiority of the SCMA and CMA methods can be shown around \( k = 1 \sim 4 \).

In order to confirm the difference of accuracies between the SCMA and CMA methods, we apply 2-sided Welch’s t-Test on
the experimental results. The null hypothesis “The accuracy is same” is rejected in the case of \( k = 4 \) at the 1% significance level and in the cases of \( k = 8, 16, 30 \) at the 0.1% significance level. However, the null hypothesis is not rejected in the case of \( k = 1, 2 \).

### B. Difference between SCMA and CMA methods

Fig. 3 demonstrates the difference between the SCMA and CMA methods. Fig. 3 is an example when the SCMA result is correct and the CMA result is wrong. The left side of Fig. 3 shows the training images of the *baseball_glove* class, the right side shows the test images. The test images are pictures of “a person with glove”. In the CMA method, these two test images are classified into categories which involve training images of several “persons”. On the other hand, in the SCMA method, the test images are classified into the *baseball_glove* class correctly. In the training images of Fig. 3, only the image C includes “a persons with glove”. The CMA method seems to use the several “person” images to approximate the test images, while the SCMA method primarily use the image C to approximate them. Fig. 4 is another example of the similar case. Looking at other examples, we find that there are several cases where the classification by the SCMA method succeeds when the test data is similar to only one or a few images in that category. This may imply that the SCMA method can approximate by selecting only the optimal training image at the sparse optimization stage.

### C. Processing time

Table II shows the processing time for classifying one test data by each method in milli-seconds. We apply each method for 1,000 test data and measure the execution time and averaging the result by the number of test data. It includes the preprocessing time, that is, reading the data, calculating the mean of data and so on. We use a Linux (fedora26) workstation, Intel Core i7 7700K 4.2GHz CPU with 16GBytes memory for the measurement. Experimental result shows that processing speed of the SCMA method is much slower than the other methods. The SCMA method uses sparse optimization, which requires iterative convergence calculation, and it takes longer time as the number of training data \( k \) increases. We use FISTA [3] for sparse optimization. Some sparse optimization methods can converge faster than FISTA. One of the future works is to use such a faster method and to improve the processing time of the SCMA method.

### D. Parameter tuning

The SCMA method contains one regularization parameter \( \lambda \). In the above experiment, we use \( \lambda = 0.1 \). However, this value may affect the classification results. In this section, we evaluate the effect of parameter \( \lambda \).
TABLE II
PROCESSING TIME FOR CLASSIFYING ONE TEST DATA (MSEC)

<table>
<thead>
<tr>
<th></th>
<th>k=1</th>
<th>k=2</th>
<th>k=4</th>
<th>k=8</th>
<th>k=16</th>
<th>k=30</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCMA</td>
<td>71.0</td>
<td>225.7</td>
<td>603.1</td>
<td>1613.7</td>
<td>4440.8</td>
<td>9347.0</td>
</tr>
<tr>
<td>CMA</td>
<td>19.7</td>
<td>21.9</td>
<td>26.6</td>
<td>36.9</td>
<td>65.7</td>
<td>139.1</td>
</tr>
<tr>
<td>SVM</td>
<td>4.4</td>
<td>7.7</td>
<td>17.3</td>
<td>44.7</td>
<td>120.3</td>
<td>293.2</td>
</tr>
<tr>
<td>CPD</td>
<td>5.1</td>
<td>5.5</td>
<td>5.9</td>
<td>7.0</td>
<td>9.4</td>
<td>13.1</td>
</tr>
</tbody>
</table>

We use the same dataset and settings as in the experiment above. The number of training data is \( k = 1 \). We evaluate the accuracy of classification by changing \( \lambda \) from 0.1 to 1.0 stepping by 0.1. We also evaluate the accuracy by changing \( \lambda \) from 0.01 to 0.1 stepping by 0.01. The result is shown in Fig. 5.

The vertical axis denotes the accuracy and the horizontal axis denotes the value of parameter \( \lambda \). The maximum accuracy of 31.2% appears at \( \lambda = 0.07 \). However, the accuracy is not so different in the range of \( \lambda = 0.01 \sim 0.5 \).

E. Group test images experiment

The SCMA method can use a group of images as one test data. In this section, we evaluate the performance of the SCMA method for group test images.

We randomly extract \( k \) training images from each categories of Caltech256 in the same way as the single test image case. The remaining images are used as evaluation data. In the evaluation, a group of \( l \) images is randomly extracted as a test data and classified. We repeat this test until the remaining evaluation data is less than \( l \). Then the accuracy of classification is calculated over these test data.

For comparison, we also evaluate the CMA and CPD methods using the same dataset. The result shows that the accuracy becomes higher than that of the single test data case. The SCMA and CMA methods showed better performance for \( k = 1, 2, 4 \), but almost the same as that of CPD for \( k = 8, 16, 30 \), which is the same property as for the single test data experiment.

VI. CONCLUSION

In this paper, we proposed the SCMA method that introduced sparse optimization in the CMA method. We showed that the SCMA method tended to classify correctly even when the training images contained wide variety of images in a class. This means that we can evaluate the contribution of each class shaper by sparse optimization. As the result, we could obtain higher accuracy in image classification. In our experiment, the SCMA method shows about the same accuracy as the CMA method when the number of training images is small, and when the number of training images increases, the SCMA method shows higher accuracy than the CMA method.

The SCMA method is slower than the comparison methods. Improving the processing speed is one of the future works. Another future work is to compare our method with state-of-the-art Deep Learning methods.

REFERENCES


Copyright © 2018 by CENPARMI, Concordia University
Saliency and Object Detection

Phutphalla Kong
Department of Information and Communication Engineering
Institute of Technology of Cambodia
Phnom Penh, Cambodia
phutphalla@itc.edu.kh

Matei Mancas
Numediart Institute for Creative Technologies
Faculty of Engineering (FPMs), University of Mons (UMONS)
Mons, Belgium
matei.mancas@umons.ac.be

Seng Kheang
Department of Information and Communication Engineering
Institute of Technology of Cambodia
Phnom Penh, Cambodia
sengk@itc.edu.kh

Bernard Gosselin
Circuit Theory and Signal Processing Lab
Faculty of Engineering (FPMs), University of Mons (UMONS)
Mons, Belgium
bernard.gosselin@umons.ac.be

Abstract— Visual attention allows the human visual system to effectively deal with the huge flow of visual information acquired by the retina. Since the years 2000, the human visual system began to be modelled in computer vision and it became part of artificial intelligence: while learning focuses on repetitive data which can easily be modeled, computational attention focuses on abnormal, rare and surprising data which can hardly be learnt.

Attention is a product of the continuous interaction between bottom-up and top-down information. While the bottom-up information has been extensively investigated through saliency models, top-down influence on visual attention has been less investigated. This paper intends to study the influence of object-based (faces and text) top-down information on bottom-up saliency maps. It proposes a simple yet effective fusion scheme that can be applied on any bottom-up saliency model depending on the object detector effectiveness and the object size. The evaluation results show that it is possible to highly improve classical bottom-up saliency models with the arrival of better object detectors. In the future, such attention models can become as effective as deep-learning based attention models while keeping them more generic and avoiding underestimating bottom-up features.

Keywords— saliency, top-down attention, bottom-up attention, visual attention, data fusion, eye-tracking, eye fixations.

I. INTRODUCTION

Computational visual attention tends to mimic human visual attention and focuses on the more informative and important parts of images. It has been the subject of various studies in a wide range of research fields such as psychology, neuroscience or computer vision. In computer vision, the main approach to the implementation of visual attention includes bottom-up and top-down information1, however, while bottom-up attention was investigated a lot [4][5][6][7][8][9][11], there were only a few experiments using top-down information in the literature. This is probably because before the arrival of deep learning in attention in 2014, the top-down detectors were not good enough, and afterwards most researchers focused on obtaining an end-to-end deep learning saliency model which naturally integrates top-down information.

There are various kinds of top-down information which can be used in addition with bottom-up saliency [1] such as location-based, contextual-based or object-based models. In this paper, we focused specifically on object-based top-down attention and especially faces and text.

The combination of face detection and low-level saliency provides already results improvements in [2]. The linear combination was weighted to give to faces the same weight that each one of the three bottom-up conspicuity maps (orientation, colour, intensity) which means that the face map global weight was quite low. This helped the authors to deal with false positives from the face detector used at that time which was not optimal. In [10], the authors showed that the high-level features such as faces and people can enhance the model performance, but there was no any precise information related to the relative importance of those features. The author also stated that using a bad object detector could clearly decrease the model performance if it produces too many false positives. In [3], the authors dealt with the importance of people and cars for saliency detection. In [16], the authors introduced the idea of the use of object symmetry as top-down attention in images. In [12], target object features from the Pascal VOC object database are learned using a CRF-modulated dictionary. The saliency maps were really focused on the objects with a very high weight.

In computer vision, Deep Neural Networks (DNNs) have changed the saliency paradigm since 2014. The deep features were first used in eDN model [17]. DeepGaze1 model [18] then showed that DNN features trained on object recognition are very useful for saliency detection. This finding seems logical as objects are most of the time regions of interest. Since then a variety of models used fine-tuned mixes of features from several deep learning models. These DNN-based models naturally incorporate top-down information during learning (such as faces and text for example). It seemed that DNNs were the perfect solution to improve the performance of classical bottom-up models.

However, in [19], the authors showed that the importance of bottom-up attention was underestimated by DNN-based models. Indeed, a simple bottom-up model can outperform a state-of-the-art DNN model on images containing less top-down information. This demonstrated that DNNs too much neglect the bottom-up aspect of visual attention and are mostly trained to detect the very attractive top-down objects than to really detect saliency. Moreover, they cannot easily adapt to images which are very different from the ones they were trained on and finally DNNs have the structural issue to provide a result that cannot really be explained in an explicit way.

1 Bottom-up information is also known as reflex exogenous reaction, while top-down information is known as reflective endogenous information.
The authors of [13] provide cues about relative importance of features based on a manual segmentation of the image dataset and is the basis of the work on this paper. In [13], the authors build a saliency model mixing their own bottom-up approach and several higher-level features. While in [13] weights of different features are computed within a specific model and cannot be used in other contexts, the purpose of this paper is to quantify the relative importance of two features proven in [13] to be very influential (faces and text) so that they can be used and integrated very easily to any general bottom-up saliency model, not necessarily in ours. In addition, we also study the size of faces and text assuming that the size is not an independent variable. The question we address here is: how to add in a simple yet effective way an object detector result to any bottom-up saliency model?

The remainder of this paper is organized as follows. In Section II, we describe the high-level features and detectors that are used. In Section III, we deal with the methods and experiments used mix bottom-up and top-down information based on several kinds of detectors. The results and discussion are finally presented in Section IV.

II. HIGH-LEVEL FEATURES AND DETECTORS

Low- and high-level information are both important to predict human gaze accurately [19][13]. In [13], the relative importance of different features was used to evaluate the model performance, which was computed by a linear SVM classifier. In terms of importance, it shows that face, text, and gaze direction are the three main features. In addition, colour, orientation, or intensity still have an interesting influence especially when human faces and text were absent [19]. However, in [13], the result cannot be easily applied to any bottom-up attention model, while here we intend to be able to use them in a generic way with any model. In sectionsII.A and II.B text and faces features are further described.

For the object detectors, we focus on 1) faces and 2) text. Inside those two important features, we try to understand the importance of the size (big faces and text versus small faces and text). The OSIE dataset (Object and Semantic Images and Eye-tracking) [13] provides us with a set of images and the manually segmented masks for text (Fig. 1b) or faces (Fig. 1c) along with the eye tracking fixation map (Fig. 1d). The different detectors that we used are detailed in Section II.C.

![Fig. 1. Extracting text and face features and eye-tracking fixation map from OSIE dataset. (a) Input image, (b) Text features, (c) Face features, and (d) Eye-tracking fixation map.](image)

A. Text features

Based on the OSIE dataset, we only select the images containing text (Fig. 2). When checking the corresponding eye-tracking maps, big text seems much more interesting than small text (around twice more interesting). This consideration pushed us to check the difference between big (Fig. 2b) and small (Fig. 2c) text. We used the OSIE masks to separate text regions between big text (more than 29 pixels in height) and small text (less than 29 pixels in height). This threshold depends of course on the image size, but all the images in the database are of the same size here.

![Fig. 2. Defining big and small text features and eye-tracking fixation map from the OSIE dataset. (a) Input image, (b) Big text features, (c) Small text features, and (d) Eye-tracking fixation map.](image)

B. Face features

In the same way, we use the manually segmented masks for the images containing faces (Fig. 3). Even if the difference in terms of eye-tracking maximum is less obvious between big faces and small faces than between big text and small text, we separate big faces (we used a threshold of 76 pixels in height) from small faces (less than 76 pixels in height) the same way as text. Again, the size of the images is always the same here, but for other datasets this threshold should be computed relatively to the mage size to be used on other datasets. Here we only take into account the frontal faces as heads viewed from rear or from the side have less chances to be correctly detected by an automatic face detector.

![Fig. 3. Defining big and small face features and eye-tracking fixation map from OSIE dataset. Big and small face features are defined by us as the text features. (a) Input image, (b) Big face features, (c) Small face features, and (d) Eye-tracking fixation map.](image)

C. Object detectors

We first define a perfect detector which is simply the human-based masks already segmented in [13].

For faces, we use a state-of-the-art face detector based on the DLIB library [22]. We used the classical Histograms of Oriented Gradients (HOG) feature followed by a SVM classifier which has a good face detection rate [21]. On this detection, we added the face template approach based on a cascade of classifiers from [15] which exhibited good results for frontal faces, with few false positives.

![Fig. 4. Applying face detection method from [15]. (a) Input images, (b) Result of face detection. The results contain either big faces (brighter), either small faces, and either both big and small faces.](image)
For text detection, we used an older approach which is integrated into the OpenCV library [14]. This detector used Extremal Regions (ERs) which are robust to several image transformations. A second step is used in the algorithm: OCR helps to improve overall results. However, in this paper we did not use any OCR results. For real-life images, this detector results are poor with both misdetections and false detections.

Thus, we can compare the results between a perfect detector (for text and face features), a good detector (face detection), and a poor detector (text detection).

### III. EXPERIMENT

The experiment intends to provide us with a clear view on how top-down information effects bottom-up information by adding text and face detection. It can be divided into two questions: 1) how to extract weights for different kind of top-down information and 2) how to mix the top-down information to bottom-up in a simple way.

To do so, we choose the top-down information as described in the previous section (Fig. 6, bottom-right), while the bottom-up saliency comes from the RARE model [11] (Fig. 6, bottom-left). This approach was purely bottom-up (no additional centred gaussian or learning-based information), and it considered both local information and global information through a rarity approach. By considering the MIT saliency benchmark [23], this model is bellow most of the DNN-based models.

![Fig. 5. Applying text detection method from [14]. (a) Input images, (b) Result of text detection inside white bounding boxes, and (c) Converting text detection areas into white to indicate text features. The results contain both big (brighter) and small texts.](image)

![Fig. 6. Components of bottom-up and top-down attention used in our experiment.](image)

![Fig. 7. Object binary mask is used along with the eye-tracking map to extract the maximum eye-tracking value for the object. This value, averaged on all the images will provide a weight for a given object.](image)

![Fig. 8. For face and text, the binary masks are split between big and small masks and then low-pass filtered to provide a smoothed result before being fused with the bottom-up saliency map.](image)

#### A. Top-down features weight

While in [13], the features weight is computed by the means of a classifier, we choose here to use the experimental data that we have in the OSIE dataset to extract a meaningful individual weight for each feature of interest. For that purpose, we decided to measure the average maximum eye gaze attractivity on all the OSIE images for big and small text and face masks. As it can be seen in the schema in Fig. 7, the eye-tracking map is multiplied by the binary mask which will provide the eye-tracking intensity on the object of interest. Then the maximum of these values is averaged over all the images in the dataset providing a weight for the given feature. For weight of big text, after 75 images we are stabilized between 0.75 and 0.78, and for weight of small text, after 75 images we are stabilized between 0.31 and 0.34. For weight of big face and small face, after 75 image we stabilized between 0.81 and 0.84 and between 0.64 and 0.67, respectively. As a result, between 75 images and 100 images are enough to get stable weights which do not depend a lot on the images we add.

#### B. Top-down and bottom-up fusion

Once the weight was computed for one of the objects among small text (ST), big text (BT), small faces (SF), and big faces (BF), the question is how to make a fusion between this information and the bottom-up saliency map.

First, as described in Fig. 8, for each feature, we split the small and big masks and then smooth them in order to obtain and image close to the fuzzy bottom-up saliency map.

We made linear combinations between bottom-up information (saliency maps) and top-down information (text and face detection). We generated results of saliency maps using RARE [11] and 1) text detection using both [14] and the masks as described in section II.A and 2) face detection using both [15] and the masks as described in section II.B.

To make a simple fusion between bottom-up saliency maps (SM) and top-down information (faces alone, text alone or both), we used linear combinations which are easy to implement. The weights were either the same for text (big and
small) and faces (big and small), either different by using the results that we obtained in section III.A which are given by the following formulas:

\[
(SM + ATF) / 2
\]
\[
(SM + BTF) / 2
\]
\[
(SM + STF) / 2
\]
\[
SM + (wSTF / wBTF) * STF + BTF
\]

where SM is bottom-up saliency maps computed from RARE [11], ATF is either all text (big and small), either all face (big and small), either all text and face depending on the experiment, BTF is either big text, either big faces, either all big text and big faces depending on the experiment, STF is either small text, either small faces, either all small text and small faces depending on the experiment. wSTF and wBTF use the weights found in section III.A for either small text or small faces (wSTF) and for either big text or big faces (wBTF).

To also test the impact of the detector accuracy, we divided our experiment into three different parts: perfect detector, good detector, and bad detector. For good and bad detector, we use the state-of-the-art detector [15] and [14], respectively while for the perfect detectors we used the masks from [13] as shown in Fig. 8 for both face and text.

IV. RESULTS AND DISCUSSION

A. Weights for face and text

To get ideal weights for big and small text and face, we used the method described in Fig. 7. As a result, we obtained a weight of big text (wBT) = 0.7871, of small text (wST) = 0.3221, of big face (wBF) = 0.8159, and of small face (wSF) = 0.6457. We can see that the difference between big text and small text is more important than big face and small face. Big face is a little more important than big text, but the difference is not very significant.

B. Perfect detector

For the perfect detector, we did three experiments. In the first one, we combined the bottom-up saliency map (SM) with text alone, then with face alone, and finally with both text and face.

We used several metrics to evaluate the bottom-up attention model object-based top-down attention by making correlation between some different results (from fusion algorithms) and the eye-tracking Fixation Maps (FM). For those metrics, we used Correlation Coefficient (CC), Kullback-Leibler Divergence (KLDiv), Normalized Scanpath Saliency (NSS), Similarity, and Area Under the ROC curve from Judd (AUC_J). Those metrics provide some complementarity and are well described in [20]. For CC, NSS, Similarity, and AUC_J, the higher value is the best, for KLDiv, the lower value is the best.

The results are summarized given the three different experiments in Tables I (SM with text alone), II (SM with face alone), and III (SM with both text and face). The first line corresponds to the comparison between the bottom-up saliency map (SM) and the eye tracking fixations map (FM). The second line corresponds with the comparison of all features (all text (AT) in Table I, all faces (AF) in Table II and all text and face (ATF) in Table III). The third line is a comparison between FM and big text (BT) in Table I, FM and big face (BF) in Table II, and between FM and big text and face (BTF) in Table III. The fourth line in Tables I and II represent the comparison between FM and small text (ST) and FM and small faces (SF), respectively. The final line is the comparison of the weighted fusion for big and small text (wBST) in Table I, for big and small face (wBSF) in Table II and all big and small text and face (wTF) in Table III.

A first global remark is that all metrics are very coherent, and they provide almost the same relative rank for all measures.

### TABLE I. RESULT BETWEEN BOTTOM-UP AND TEXT DETECTION

<table>
<thead>
<tr>
<th>Correlation</th>
<th>Metrics</th>
<th>CC</th>
<th>KLDiv</th>
<th>NSS</th>
<th>Similarity</th>
<th>AUC_J</th>
</tr>
</thead>
<tbody>
<tr>
<td>SM, FM</td>
<td></td>
<td>0.4683</td>
<td>1.0591</td>
<td>1.5364</td>
<td>0.4364</td>
<td>0.8365</td>
</tr>
<tr>
<td>AT, FM</td>
<td></td>
<td>0.5042</td>
<td>1.0140</td>
<td>1.7013</td>
<td>0.4514</td>
<td>0.8452</td>
</tr>
<tr>
<td>BT, FM</td>
<td></td>
<td>0.5058</td>
<td>1.0151</td>
<td>1.7008</td>
<td>0.4504</td>
<td>0.8444</td>
</tr>
<tr>
<td>ST, FM</td>
<td></td>
<td>0.4666</td>
<td>1.0587</td>
<td>1.5420</td>
<td>0.4378</td>
<td>0.8372</td>
</tr>
<tr>
<td>wBST, FM</td>
<td></td>
<td>0.5061</td>
<td>1.0127</td>
<td>1.7081</td>
<td>0.4517</td>
<td>0.8454</td>
</tr>
</tbody>
</table>

In Table I, it indicates that adding information about small text brings nothing to the result because result of small text is a little less good than the bottom-up saliency map alone in some metrics. On the other hand, adding big text provides an important and improvement result in the CC metric compared to result when adding both small and big text (AT). For all metrics, the use of the weights provides the best results of all.

### TABLE II. RESULT BETWEEN BOTTOM-UP AND FACE DETECTION

<table>
<thead>
<tr>
<th>Correlation</th>
<th>Metrics</th>
<th>CC</th>
<th>KLDiv</th>
<th>NSS</th>
<th>Similarity</th>
<th>AUC_J</th>
</tr>
</thead>
<tbody>
<tr>
<td>SM, FM</td>
<td></td>
<td>0.4683</td>
<td>1.0591</td>
<td>1.5364</td>
<td>0.4364</td>
<td>0.8365</td>
</tr>
<tr>
<td>AT, FM</td>
<td></td>
<td>0.5352</td>
<td>0.9790</td>
<td>1.8334</td>
<td>0.4581</td>
<td>0.8494</td>
</tr>
<tr>
<td>BF, FM</td>
<td></td>
<td>0.5277</td>
<td>0.9876</td>
<td>1.7890</td>
<td>0.4553</td>
<td>0.8478</td>
</tr>
<tr>
<td>SF, FM</td>
<td></td>
<td>0.4762</td>
<td>1.0514</td>
<td>1.5851</td>
<td>0.4396</td>
<td>0.8379</td>
</tr>
<tr>
<td>wBSF, FM</td>
<td></td>
<td>0.5354</td>
<td>0.9786</td>
<td>1.8348</td>
<td>0.4582</td>
<td>0.8494</td>
</tr>
</tbody>
</table>

In Table II, it indicates that adding information about small face brings this time a small improvement to the bottom-up saliency map alone and is never negative. However, adding big face provides an important result improvement. Moreover, adding both big and small face also brings improvement. The best case, for all metrics, the use of the weights provides the better results of all.

In Table III, we did not compute the result for small text and face since it was always smaller than big text and face. We just kept here the best combinations: ATF for all text and
face, BTF for only big face and text, and the weighted text and face (wTF). While ATF is always a little better than BTF, the weighted version is even better.

### C. Imperfect detectors

Here we decide to use two state-of-the-art detectors which imply some misdetections or false detections (especially for the text detector). We don’t combine result for both text and face detection since the text-related results are very bad (see Table IV). However, the results from face detection are good because facial landmarks approach [15] can detect the frontal face well although it misses some faces in a scene.

Table IV shows that the misdetections and even more the false detections of the text detector seriously decrease the results compared to the bottom-up saliency map alone. This text detector is not good enough to be used to add top-down information. For all metrics, results of the saliency map alone are better than all text, big text, and weighted.

Table V shows that the face detector, which has a better quality, can provide good improvement of the bottom-up saliency map. However, the difference between a simple average fusion (AF) and the weighted version (wBSF) is not significative. For some metrics such as CC and KLDiv it is even better to just make the global average instead of using the face weights.

### D. Discussion and conclusion

In this paper, we show how to simply add top-down information to any bottom-up saliency models in a generic way. Our work focused on both text and face features.

We tested several object detectors (bad, good, and perfect), and we demonstrated that if the detector is not good enough, it is better not to use it at all and only use the bottom-up information (Table IV). If the detector is good, a simple average can be almost as good as a more complex weighted average (Table V). When the detector becomes very good, then the weighted average really makes sense (Tables I to III). This is even more the case when several top-down features are mixed to bottom-up and some might be more important than others.

The size of the top-down object is very important. This was more the case with text where the difference in terms of eye fixations between big text (titles) and small text (description) is very important. Indeed, people provide attention to text because of its cognitive content. While for titles, the cognitive load needed is very small, for blocks of smaller text, they will less attract attention, especially if the beginning of the text (the most attended) has no important information. There is still a difference between big face and small face, but this difference is smaller. If we just consider big text and big face, the weights values of them are almost the same. It is an interesting result as previous results do not consider the difference between big and small text or big and small face [13]. The result was polluted by small text which really decrease a lot the overall text importance.

An important result improvement can be obtained by using classical bottom-up attention models to which we can add easily the higher level detected objects. Resulting models will approach novel DNN-based attention approaches while they keep generality. They are also well responding to bottom-up features which are underestimated by DNNs [19]. In addition to that, classical models can have a behaviour which can be explained while DNNs provide results without letting any chance to the programmers to explain why exactly their model works well or not. Being able to explain the reaction of an algorithm might be critical especially for security applications. That is why, for our future work, we will go deeper in the object-based top-down features which can be extracted and the optimal mix with bottom-up saliency maps.

### Acknowledgment

This research study is supported by ARES-CCD (program AI 2014–2019) under the funding of Belgian university cooperation.

### References


Comparing Presentation Attack Detection Methods using Convolutional Neural Networks and Local Binary Patterns

Justin Spencer  
Computer Science Department  
North Carolina Agricultural and Technical State University  
Greensboro, North Carolina, United States of America  
bjspencer@aggies.ncat.edu

Deborah Lawrence  
Computer Science Department  
University of North Carolina—Asheville  
Asheville, North Carolina, United States of America  
dlawrenc@unca.edu

Prosenjit Chatterjee  
Computer Science Department  
North Carolina Agricultural and Technical State University  
Greensboro, North Carolina, United States of America  
pchatterjee@aggies.ncat.edu

Kaushik Roy  
Computer Science Department  
North Carolina Agricultural and Technical State University  
Greensboro, North Carolina, United States of America  
kroy@ncat.edu

Albert Esterline  
Computer Science Department  
North Carolina Agricultural and Technical State University  
Greensboro, North Carolina, United States of America  
esterlin@ncat.edu

Jung Hee Kim  
Computer Science Department  
North Carolina Agricultural and Technical State University  
Greensboro, North Carolina, United States of America  
jungkim@ncat.edu

Abstract—The use of biometrics to authenticate users and control access to secure areas has become extremely popular in recent years, and biometric access control systems are frequently used by both governments and private corporations. However, these systems may represent risks to security when deployed without considering the possibility of biometric presentation attacks (also known as spoofing). Presentation attacks are a serious threat because they do not require significant time, expense, or skill to carry out while remaining effective against many biometric systems in use today. This research compares two deep learning-based methods and one texture-based method for facial and iris presentation attack detection on baseline datasets. The first deep learning method uses Inception-v3, a pre-trained deep Convolutional Neural Network (CNN) made by Google for the ImageNet challenge, which is retrained for this problem. The second deep learning method uses a shallow CNN based on a modified Spoofnet architecture, which is trained normally. These CNN-based approaches are compared with a traditional texture-based method using Local Binary Patterns (LBP). The datasets used are the ATVS-FIr dataset, which contains real and fake iris images, and the CASIA Face Anti-Spoofing Dataset, which contains real images as well as warped photo, cut photo, and video replay presentation attacks. We also present a third set of results, based on cropped versions of the CASIA images.

Keywords—biometrics, presentation attack, deep learning, CNN, LBP.

I. INTRODUCTION

Researchers have studied ways to measure and differentiate between different palm prints, gaits, voices, fingerprints, irises, faces, and other biometric identifiers. All are interesting and effective ways to verify an identity, although the research reported here examines only iris and facial biometrics. Through a variety of methods, it has become possible for a computer system to scan and analyze a face or iris and then grant access based on whether that biometric is recognized.

Biometric-based presentation attacks involve gaining access to a biometric sample from databases or external resources, then reusing that biometric to gain unauthorized access to confidential data or secure facilities. Though the use of biometric authentication strengthens security through unique features, the cloning of the biometric sample and the associated unique features to access a biometric system illegally is feasible.

Though face and iris recognition are more reliable biometrics, spoofing has still become a common threat for these biometrics. There are multiple ways an attacker can spoof a biometric system. High-resolution copies of biometric samples have been used to spoof systems. Photorealistic face masks and synthetic images have also been used successfully in presentation attacks. Digital retouching of images is also a common spoofing threat [1]. Face and iris spoofing can be categorized as texture-based spoofing, motion-based spoofing, 3D shape-based spoofing, and multi-spectral reflectance-based spoofing [2].

II. RELATED WORK

We investigate several forms of presentation attack detection that use machine-learning methods, including Convolutional Neural Networks (CNNs), deep belief networks, and micro texture analysis [3, 4, 5, 6]. We also examine techniques for the more general problem of facial recognition and classification that are useful for our research [7, 8].
Deep learning techniques to mitigate presentation attacks have shown promising results [1, 2, 5]. Menotti et. al propose building an anti-spoofing system using a CNN with a combination of two approaches [5]. The first approach focuses on learning an appropriate CNN architecture. The second approach consists of learning filter weights via the standard backpropagation algorithm [9, 10]. Yang et. al. [2] and Zhang et. al. [11] propose diverse spoofing attack prevention techniques including face localization, spatial augmentation, and temporal augmentation in combination with canonical CNN filtering techniques for feature training and classification. CNNs are the primary focus of this research.

Silva et. al. use a deep belief network in order to detect whether a user is wearing contact lenses [12]. They define a three-class detection problem by dividing the images based on the presence of soft (uncolored) contact lenses, colored contact lenses, and no contact lenses. They use a combination of a CNN for deep image representations and a fully connected three-layer network for classification. Instead of using a specific search algorithm, the researchers analyzed a set of parameters to build the final network topology and to learn the filter weights by backpropagation. They have also suggested future work with random weights. On certain datasets, their methods outperform state-of-the-art approaches. However, their current approach does not segment the iris, and this becomes a problem in datasets where the iris region is not pre-identified.

Farfade, Saberian, and Li investigate facial recognition for partially obscured faces and faces at an angle [13]. Using a specific CNN called Deep Dense Face Detector (DDFD), the authors create a system that can recognize facial features without the entire face being visible. They specifically were interested in creating a single model that could recognize faces despite a variety of obstructions, such as rotated faces, skewed faces or faces in profile, partially obscured faces, and others. DDFD does not require landmark annotation, meaning it does not specifically pick out images of eyes, noses, and other significant facial features. Instead, the authors used a dataset of 21,000 unaltered and altered images featuring partially obscured human faces in order to train the CNN. Altered images involved sampling the dataset and randomly cropping certain images in order to represent an obscured face. By training the CNN to recognize cropped images, DDFD could recognize faces in a variety of situations. DDFD used five convolutional and three fully connected layers and produced classification results comparable with state-of-the-art results in other published work.

Garcia and Delakis focus primarily on facial recognition with images that have been captured in non-controlled environments [14]. These images are of variable size, quality, and rotation. The authors used a dataset created from images from the internet and scanned newspapers to create a CNN called the Convolutonal Face Finder (CFF) specifically for their work. When processing images, CFF classifies various local features such as end-points or edges, combines them in later layers to identify larger features such as noses or eyes, and then measures the feature’s distance relative to other features to recognize a face.

Yang, Lei, and Li are perhaps the first to suggest using a CNN to differentiate between real and spoofed faces [2]. They crop the CASIA and Replay-Attack dataset images to five different sizes in order to determine the influence of the background in presentation attack detection. Their best result is with the second-largest size, possibly because the largest size results in overfitting. Their research shows that the background can be useful in certain image identification problems, and that CNNs are an effective solution for presentation attack detection.

Li et. al. also propose the use of a CNN in presentation attack detection [15]. They retrained the VGG-face model created by the Oxford Visual Geometry Group (VGG) on the CASIA and Replay-Attack datasets. Their approach has reasonable performance in comparison with previous work on CASIA and Replay-Attack.

Microtexture analysis has also proven to be effective not only with identifying faces but also with detecting a presentation attack. Multiple microtexture analysis techniques in order to identify texture differences between a real face and a spoofing attempt, with Local Binary Patterns (LBP) performing most effectively on a dataset of real and printed photos [16, 17]. Chingsokva et. al. [18] extend this work and test LBP variants on a custom dataset (Replay-Attack) that includes multiple presentation attack methods. The methods involved include a print photo attack, a mobile attack (a photo taken with a mobile phone is displayed to the sensor), and a high definition attack (a photo taken with a high-quality tablet is displayed to the sensor.) It proves that a basic LBP technique produces the most accurate results, achieving stronger results, possibly due to the differences in datasets.

Farfade et. al. [13] as well as Garcia et. al [14], do not directly investigate presentation attack detection. However, their respective work on detecting obscured faces and faces in non-ideal environments is a necessary condition for presentation attack detection. These attacks may be carried out in poor lighting or with partially obscured faces, so any presentation attack detection technique designed for the real world use must be able to operate effectively under these conditions. The research by Silva et. al. on contact lens detection is similarly applicable to detecting iris presentation attacks that use textured contact lenses. Finally, the research by Menotti et. al. [5] and Yang et. al. [2] helped inform our decision to use CNNs and to test cropped versions of the CASIA images.

III. THEORETICAL BACKGROUND

Presentation attack detection algorithms are classified as either hardware-based or software-based methods. We investigate software-based methods, which are cheaper since they do not require specific hardware and tend to be more user-friendly since they do not require a challenge-response [16]. Software-based methods can be divided into additional subclasses, such as dynamic methods that use temporal information (e.g., videos) or static methods that do not (e.g., still images). We use only software-based static methods for still images. Techniques examined include a texture-based approach using LBP's and a machine learning-based approach using CNNs.

Texture-based approaches in image analysis use microtextural data in order to determine characteristics of the images provided. LBP is a texture classification technique that
compares a single pixel with its eight neighboring pixels [17]. Moving in a circle around the center pixel, LBP compares the brightness of the center pixel with each neighbor, determining whether the neighbor has a larger (brighter) or smaller (darker) value than the center pixel. If the neighbor is brighter than the center cell, it is assigned a value of one. If it is equally bright or darker than the center pixel, it is assigned a value of zero. These values are then converted into a histogram. After performing this process for a series of pixels, the histograms are combined to produce a feature vector. Finally, this feature vector is used to classify the image. LBP has proven to be effective in image classification and facial recognition, and it can be applied to presentation attack detection due to microtextural differences between real human faces and printed photos or iPad screens used for spoofing [18].

CNNs are a heavily modified form of traditional neural networks designed for image recognition. Traditional neural networks are made out of a number of different layers. Each neuron in a layer receives input signals from all the previous layer’s neurons and transmits an output signal to all the next layer’s neurons if and only if the input signal strengths exceed some threshold. Because each neuron in a layer is fully connected to every neuron in the previous layer, the number of connections grows rapidly with the number of layers in the network as well as the number of neurons in each layer.

In practice, traditional neural networks support a very limited number of layers, require large amounts of training data even for small networks, and have high hardware requirements and performance costs [4]. Since they do not scale effectively, it is difficult to use them for image recognition and other problems with complex input domains.

CNNs differ from traditional neural networks in that neurons in each layer are not fully connected to all the previous layer’s neurons. This is because not every pixel in an image is related to every other – for example, background pixels are not related to pixels of a subject’s face [4]. By removing unnecessary connections between neurons, the size and complexity of the network is reduced, allowing CNNs to have more layers and achieve better results on image recognition problems. They also become desensitized to minor variations in the input image. Because of these factors, they are especially efficient tools for facial recognition and classification.

Finally, we explore a technique called transfer learning. Transfer learning involves the reuse of a neural network that has already been trained by other researchers on a different problem. Normal training involves initializing the neuron weights to some value, and then updating them to achieve better accuracies using backpropagation. In transfer learning, a neural net trained on one problem (e.g., facial recognition) is retrained on a new, similar problem (e.g., facial presentation attack detection). Since the two problems are similar, the neural network does not need to be fully trained on the new problem and can reuse what it learned on the original problem. To achieve this reuse of parameters, most of the layers have their weights held constant, and only a few layers have their weights updated. This greatly reduces the time and amount of data required to train the neural network, although the accuracy suffers slightly since the two problems are not exactly the same. For larger CNNs, transfer learning can reduce the training time from several weeks to several hours.

IV. RESEARCH OVERVIEW

As there is no baseline dataset or analysis technique for presentation attack detection methods, it can be difficult to properly determine the most effective methods. We have chosen to compare the strength of multiple techniques. The techniques to compare have been chosen based on their effectiveness reported in the papers that have proposed and tested them. Our work will measure the effectiveness of LBP, a shallow CNN, and a deep CNN on cropped and uncropped images. We design and implement a shallow CNN that successfully classifies real and spoofed faces and irises. We then compare the accuracy of our fully trained CNN against a pre-trained Inception-v3 instance that is retrained for this problem. Inception-v3 is a deep CNN designed by Google and trained on the ImageNet dataset [19].

Our goal is to examine the accuracy of a shallow CNN architecture versus a deep CNN architecture and determine the ratio between accuracy, performance, and complexity. Our CNN is a modified version of the Spoofnet architecture with six layers whereas the Inception-v3 model has 48. We also compare the effectiveness of ordinary training versus transfer learning for presentation attack detection.

Next, we use LBP to detect presentation attacks, and compare this texture-based method with the above deep-learning methods. LBP was chosen as a comparison technique because it is generally an effective method for classifying multiple kinds of presentation attacks [16]. Since LBP is both highly accurate and based on a different methodology than deep learning [20], it serves as a useful benchmark for our comparison.

By using both cropped and uncropped images, we may compare the accuracy of the different tools when background information is both present and absent [21, 22]. Cropping theoretically should not affect the texture-based methods but will likely affect the CNNs due to the loss of background information. In previous work, it has been shown that CNNs struggle with over-cropped images but perform best when some cropping is done to prevent overfitting [2, 23, 24].

We use multiple datasets with multiple forms of image recognition in order to compare the relative performance of these methods. The results of this work can inform future researchers about the effectiveness of different techniques for presentation attack detection.

V. CNN ARCHITECTURE

Generally, a CNN architecture is formed by a stack of discrete layers [4]. It transforms the input volume into an output volume through a differentiable function [4]. Inspired by the Spoofnet architecture used in [5], we designed our own CNN architecture. The first layer in our CNN is a convolutional layer with a 5x5 kernel, 16 filters, and ReLU activation. Next, there is a max-pooling layer with a 3x3 pooling window and a stride of three. The third layer is convolutional and is identical to the first except for having 32 filters. This is followed by a second max-
pooling layer that is identical to the first. We flatten the output of this layer before feeding it to a dense layer with 128 neurons and ReLU activation. Finally, there is a dense layer with one neuron and sigmoid activation. We use binary cross-entropy as the loss function, a learning rate of 0.001, and the Adam optimizer. Fig. 1 shows the major features of our CNN architecture.

VI. DATASETS

We report results for three datasets: ATVS, CASIA, and CASIA-cropped. ATVS is an iris image dataset that contains fifty subjects [21, 22]. Subjects had each eye photographed four times in each of two different sessions. Each image was then printed and rescaled at a reduced quality to create the spoofed iris images. There are a total of 1600 images, with 32 per user and 800 per class (real and fake). Each image is a grayscale 640x480 BMP file that we converted to JPG.

CASIA is a face video dataset containing fifty subjects [11]. Within each class, a subject has one low-resolution landscape-style video, one low-resolution portrait-style video, and one high-resolution portrait-style video that are approximately ten seconds long. The four classes are real subjects, “warped photo” presentation attacks (printed photo of the subject held up to the camera with the photo moved back and forth to fool liveness detection systems), “cut photo” attacks (printed photo of the subject with eyeholes cut out and the real user positioned behind the photo to fool blinking detection systems), and video replay attacks (tablet or screen held up to the camera while playing a video of subject).

We converted each MP4 video to a series of still JPGs, with one image per frame of video. Not all videos are the same length, and therefore not every class has the same number of images. Some (but not all) high-resolution videos had an initial black frame, which was discarded. The low-resolution images are 640x480 (landscape style) and 480x640 (portrait style). The high-resolution images are 720x1280 (portrait style). In total, there are approximately 111,000 color images after these transformations, with 20,000 to 30,000 per class.

CASIA-cropped is a custom dataset that we created based on modified versions of the CASIA images. We used pretrained OpenCV Haar cascades [25] to detect and crop the face region for every subject image in the CASIA dataset. Images without a detected face region, or with more than one detected face region, were discarded. We then cropped the resulting images a second time to ensure all images were the same size (140x140). There are approximately 94,000 color images in this dataset, with 22,000 to 26,000 per class. Cropping the images creates a more challenging problem for the classifier since a CNN can no longer learn information about the background region to detect spoofing. In the standard CASIA dataset, for example, a CNN may detect presentation attacks by locating the edges of a photo or tablet held up to the camera. In CASIA-cropped, this is no longer possible, and the neural net must learn other, less obvious features to detect spoofing. This helps to create a more robust spoofing detection system that will generalize better to new types of attacks. For example, a photorealistic mask would not have well-defined edges and might fool systems that use edge detection to classify attacks.

VII. METHODOLOGY

Inception-v3, originally trained by Google on the ImageNet dataset, is retrained on these datasets for 4000 epochs and uses an 80%/10%/10% training/testing/validation split. It does not require that input images be the same size, and it loads the full contents of every dataset in batches.

Our custom CNN is trained normally for 30 epochs, uses a 50%/50% training/testing split, and uses the first 30 testing images as a validation set. Unlike Inception-v3, it requires that all input images be the same size. Note that our CNN code can load the full ATVS dataset due to its smaller size but cannot load the larger CASIA and CASIA-cropped datasets. This is because it must load the entire dataset at once instead of in batches like Inception-v3. We therefore load 800 640x480 images per class for CASIA and 800 140x140 images per class for CASIA-cropped. The number of images per class, and the image size used for CASIA, were chosen to match ATVS. Although our CNN obtains reasonable accuracy when it converges, it fails to find a gradient about one-third of the time.

In this research, we generate ROC curves for our custom CNN and for binary classification problems. We place all warped photo attack, cut photo attack, and video replay attack images into a single “fake” class, which is compared with the existing “real” class.

Our LBP code also requires that all input images be the same size but is able to load all images of that size instead of only 800 per class. We load all 640x480 images for CASIA and all 140x140 images for CASIA-cropped. We report the smallest number of patches that achieves the highest accuracy.

VIII. RESULTS AND DISCUSSION

Our results are shown in Table 1. The spoofed ATVS images are quite simple and are easily detected by all methods. The CASIA images are more complex, and although our CNN performs well on the regular images, the cropped images force our CNN to guess randomly for the full four-class problem. Binary classification on CASIA-cropped is simpler, and our CNN does much better on that problem due to its reduced dimensionality. LBP and Inception-v3 both perform extremely well regardless of the image size or dataset used.
The ROC curves (Figs. 2-4) show the True Positive Rate (TPR, solid line) versus the False Positive Rate (FPR, dashed line) for our CNN on the ATVS, CASIA, and CASIA-cropped datasets. The TPR should be as close to 1.0 as possible, meaning that all legitimate users were accepted and none were rejected by mistake. The FPR should be as close to 0.0 as possible, meaning that all presentation attacks were rejected and none were mistakenly accepted as real users. In all cases, the area under the curve remains at 0.98, which is extremely close to the ideal value of 1.0.

Our custom CNN performs best on ATVS, which is a simple binary classification problem. Our CNN also achieves good results on CASIA. However, it is forced to guess randomly when classifying CASIA-cropped (25% accuracy with four classes). We hypothesize that this is due to its small number of layers. It can learn simple features (such as the edge of a photo or tablet) very easily and does well on CASIA, where those features are present. On CASIA-cropped, the accuracy suffers because these features are not present. Compensating by learning more complex features is not possible for this very simple CNN.

On the simpler binary CASIA problem, our custom CNN is able to achieve much better accuracies. We believe that the differences between real images and spoofed images are very simple and therefore easy for a simple CNN to learn. Distinguishing between different types of attacks, however, is more complex since the different attack images are very similar (e.g., both warped photo and cut photo attacks involve a printed photo held up to a camera). Learning the difference between these very similar classes requires a more complex CNN with more layers, such as Inception-v3.

We believe Inception-v3’s reduced performance on CASIA-cropped is also due to the loss of obvious features such as photo edges. Inception-v3 is nonetheless able to compensate for the loss of information due to its large number of layers. The extra complexity allows it to learn more abstract features. Possible candidates include texture-based features such as albedo (AKA reflectivity).

Finally, we believe that LBP’s excellent performance is due to the same textural features. All presentation attack images have the common characteristic of being printed on paper or displayed on a screen. The microtextural differences between these surfaces and human faces have been previously reported, and LBP is known to perform well on presentation attack detection [18]. We would be interested to see whether LBP

---

**Table 1: Results Matrix for Individual Tools and Datasets.**

<table>
<thead>
<tr>
<th>Methods Used</th>
<th>ATVS</th>
<th>CASIA</th>
<th>CASIA-cropped</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inception-v3</td>
<td>100% R/F</td>
<td>98.7% R/F</td>
<td>90.2% R/F</td>
</tr>
<tr>
<td></td>
<td></td>
<td>90.5% R/F</td>
<td>25% W/C/V/R</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modified Spoofnet</td>
<td>97% R/F</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LBP</td>
<td>100% R/F (1x1 patch)</td>
<td>100% W/C/V/R (1x1 patch)</td>
<td>100% R/F (1x1 patch)</td>
</tr>
</tbody>
</table>


*b* R: Real images. F: Fake images. For CASIA and CASIA-cropped, “Fake” includes all warped photo, cut photo, and video replay attack images.

---

---

---

---
performs similarly well with photorealistic masks or textured contact lenses used in spoofing.

IX. CONCLUSION

Through this work, we provide a baseline dataset and analysis technique that provides a more effective comparison between presentation attack techniques. We compare the use of a deep CNN with the use of a shallow CNN for biometric presentation attack detection in still images. We also compare the use of a texture-based method versus machine-learning methods and the effectiveness of transfer learning versus normal training. We demonstrate a modified Spoofnet architecture that can effectively distinguish between presentation attacks and real users and can classify presentation attacks by type under certain conditions. Finally, we discuss whether a machine learning-based approach or a texture-based approach is most effective, and the relative strengths and weaknesses of each.

For future work, we would like to enhance our CNN code to function more like Inception-v3. These enhancements would include loading data in batches (to make it possible to load all images in theCASIA datasets), loading input images of different sizes, and using the same proportions of training, testing, and validation data as Inception-v3. We would also like to implement some form of data augmentation and multi-class ROC curves.

ACKNOWLEDGMENT

This research is based upon work supported by the Science & Technology Center: Bio/Computational Evolution in Action Consortium (BEACON), NSF REU grant (#1460864), NSF IUSE grant (#1504917) and the Army Research Office (Contract No. W911NF-15-1-0524).

We would like to acknowledge our graduate student mentor, John Jenkins, as well as Dr. Joseph Shelton, who provided code to analyze images using LBP, and Ryan Dellana, who provided support for our custom CNN code.

REFERENCES


SDRN: Scalable Deep Rectifier Network for opinion spam detection

Zeinab Sedighi  
Department of Computer Engineering,  
Faculty of Computer and Electrical Engineering, University of Kashan  
Kashan, I.R.Iran  
sedighi.63@gmail.com, sedighi.63@grad.kashanu.ac.ir

Hossein Ebrahimpour-Komleh  
Department of Computer Engineering,  
Faculty of Computer and Electrical Engineering, University of Kashan  
Kashan, I.R.Iran  
ebrahimpour@kashanu.ac.ir

Ayoub Bagheri  
Department of Computer Engineering,  
Faculty of Computer and Electrical Engineering, University of Kashan  
Kashan, I.R.Iran  
a.bagheri@kashanu.ac.ir

Abstract—Today, comments on the web effect on decision making, significantly. Unfavorable impacts of deceptive spam attacks can promote or demote people’s attraction deliberately. So these reviews are able to follow heavy damages for customers and organizations. Thus, detecting deceptive reviews has attracted more attention these days. The majority of methods so far classify reviews to spam and non-spam groups. Therefore, most researches are done on feature learning techniques to enhance the classification performance. From another point of view, presence of huge amount of features makes text classification overwhelming. Existing approaches mainly use handcraft techniques to extract proper features for better discrimination. However, since these methods select context free features, they fail to extract relevant ones. To this end, we propose a parallel deep structure to combat both mentioned challenges simultaneously, which obviously demonstrated on two different datasets. In this way, reasonable features are derived as well as reducing the model complexity. Our proposed model accuracy and scalability is honorable compared other common models.

Keywords—opinion spam detection, deep learning, text mining.

I. INTRODUCTION

Due to vast use of internet, more data is generated every day. Maintaining compatibility with this trend, companies and organizations employ their users comments as feedback to improve their business. Apart from this, most of users and customers use others reviews to make decision about a product or an issue. Thus, more attention are paid to opinion mining Nowadays[1, 2]. Since customers are able to easily comment and share their experiences on specific topics, it likely increase fake review attacks, called spam, through individual or groups. Hence, reviews may contain deceptive comments and blind trust in them leads to unfavorable consequences [3]. Consequently, opinion mining on customer reviews has become a hot topic in recent years. Since pros comments on a product encourage other people to conform, it can be advantageous for any organization or individual. This property of customer reviews lead growth in production of deceptive review spam. Deceptive review spam is a kind of spurious review which is written in such a way that looks reliable. Discerning a deceptive review from truthful ones for human is difficult and inaccurate [4], therefore more attention is given to detect review spam spontaneously. Deceptive review spam detection aims to establish security on the web and maintain its reliability. Although the issue has attracted more attention so far, but it’s an open problem yet and one-third of opinion reviews on web are approximated as fake [3]. In general, detecting fake reviews is a classification problem which separate deceptive reviews from truthful ones. To avoid deceptive reviews effects on decision making, opinion spam detection techniques benefit machine learning methods to filter out these harmful reviews, automatically. From the primitive work on detecting deceptive reviews [5] to now, most researches apply machine learning classifiers to distinct spam reviews from others. Consequently, more attentions are paid to learn appropriate features which leads the classifiers tend to better performance.

According to Dixit et. Al. [6] spam reviews are divided in to:
1) untruthful reviews which deliberately affect user decisions,
2) reviews which only advertise on the brands and
3) non-reviews which are advertisements or irrelevant.

In this article we focus on reviews type 1, which try to mislead users by deceptive comments on a special issue. From now on, we call this type of review as spam. To diminish the number of model inputs, we present a deep learning model to learn representation and features automatically. In order to reduce the complexity and run time of model to deal with large amount of features in text processing techniques, a deep classifier is applied. To diminish the number of model inputs, we present a preprocessing level which perform basic feature engineering methods. Then we utilize a supervised learning model to discriminate truthful reviews from spam ones and employ it in a public dataset. The proposed model obtains better result compared to the current popular models. This article is organized as follows. Section 2 surveys related works of deep learning and opinion spam review detection. The methodology we apply in this article is explained in
section 3. Section 4 demonstrates the discussion and after all, section 5 concludes the paper.

II. Related works

Discovering deceptive reviews has been challenging nowadays since reviews are wrote by unknown people for any purpose [7, 8]. Detecting spam reviews has been accomplished by supervised learning methods for the first time [5]. These techniques employ text classification models which trained by labeled data sets to distinguish between deceptive and truthful reviews. Opting convenient features is the most principal act which enhances the classification quality. Since a deep network model is used to select representations of data in this article, we review the related works into two following subsections opinion spam detection and deep learning for this specific subject.

A. Opinion spam detection

Some significant techniques of detecting spam in various social media is discussed in [9, 10]. Zhang et al. [11] evaluated machine learning techniques to filter out spam text. Most of machine learning techniques are applied to detect spam reviews such as rule based techniques [12], other classifiers like decision tree [13] and ensemble [14]. Many researches survey on reviews which seek to find methods dealing with detecting spam reviews [15]. All these techniques aim to separate spam from truthful reviews which are divided into unsupervised, semi-supervised and supervised methods. Since reviews on different topics are unlabeled all around the Internet, some studies perform on unsupervised based models. Reymond et al. [16] propose an unsupervised model which use semantic language techniques to spam detection. In order to improve the detection performance, semi-supervised methods can be used to profit few amount of labeled data to cluster all the data. Mentes et al. [17] applied PU-learning [18] as a semi-supervised model to distinguish spam reviews.

To overcome the lack of labeled data, Ott et al. [19] produced an artificial data set and applied supervised learning technique for text classification. Spam detection is employed for text summarization in [20]. Jindal et al. [20] apply Naïve Bayes, logistic regression and SVM methods after feature extraction using POS tags and LIWC. To dominate in cross domain challenge, Li et al. [21] use a data set consist of three reviews domains to avoid the dependency to a specific domain. They examine SVM and SAGE for classification phase.

All the above methods suffer from hand crafted feature extraction procedures which can be effective in the classifiers performance.

B. Deep learning analysis

Deep learning is a hierarchical model which is inspired by neurons in the human brain. It became popular by [22] which models a face recognition system. Bengio in [23] details these representation learning approaches and ways of training them.

Deep learning is an executable impressive technique in various areas such as computer vision, speech recognition and natural language processing. Since feature engineering in processing documents is demanding and inaccurate, using automatic and more accurate model is preferred. Deep learning methods automated procedures with no need to get prior knowledge. By migrating from text to real value features, deep learning models is widely used in natural language processing and text mining [24] such as sentiment analysis [25], co-reference resolution [26], POS tagging [27] and parsing [28].

In sentiment analysis and opinion mining deep learning approaches are notable for generating textual sequences [29], sentence modeling [30], word embedding [31] and etc. Yafeng and Donghong [32] applied a hybrid model based on convolutional networks to build a model which learns document-level representation for detecting deceptive review spam. Jain et al. [33] overviews RNN and CNN techniques for spam detection in social media.

To process a document using deep models, a word embedding step is performed. Word embedding is a feature learning step which maps vocabulary space to real number vectors. Therefore, in a mathematical view, a word space is embedded in a real value space which contains lower dimensions. Hence, words distributions and distance can be calculated by mathematical approaches. To facilitate use of deep learning structure we apply H2O [34] for implementation which will explain further. Thus, using a deep hierarchical model to extract relevant features fully automates detecting deceptive reviews. On the other hand, parallelizing a linguistic model deals with the processing numerous features produced in mining documents.

III. SDRN model

Here, we detail how we employ deep learning to detect fake reviews. Then explain the model characteristics.

A. Deep structure to detect review spam

Today, deep learning techniques overcome the traditional neural network shortcoming for generalization and scalability. There exist various frameworks for deep learning models, here, in order to overcome the problem of vanishing gradient in deep structure we apply a deep rectifier feedforward network.

Deep learning superiority over shallow machine learning techniques is using efficient feature learning algorithms instead of manual approaches. Utilizing hierarchical feature extraction structure develops features in different resolutions [35]. Output features from each layer in these networks are transferred to the next layer. Consequently, higher level features induced by lower layers construct a hierarchy from low level to high level features [36].

Each layer composed of nonlinear processing unit. In this study we employ rectifiers as processing units. No need to normalization and exponential time complexity lead rectifiers fast units [37]. Moreover, it doesn’t suffer from vanishing
gradient due to its function form. Fig 1 shows rectifier function. Rectifier is an activation function can be defined by:

\[ f(x) = \max(x, 0) \]  

(1)

where \( x \) is the input to a neuron. As a smoothed form of it, we can use:

\[ f(x) = \ln(1 + e^x) \]  

(2)

Despite other activation functions, gradient doesn’t vanish by increasing \( x \) in a rectifier function.

A deep neural network is formed of various hidden layer in addition to input and output layers by which non-linear functions can be modeled. Features in each layer are composed of features from previous layer that enable the network to make complex structures. A neural network output is specified by the weights connecting neurons together. Thus, weigh adaptation occurs by minimizing the training error. The objective function which should be minimized is

\[ J(\theta) = L(f(j, \theta), y) \]  

(3)

where for training sample \( j \), \( \theta \) is a symbol of the model parameters and \( f(j, \theta) \) is the predicted output [34].

Fig 2 details our deep model structure. We present a hierarchical model providing sentence level features. The network consists of three layers of rectifier units. At lookup layer, it maps the words to a lookup table by applying word embedding process. Hence, this layer results real value vectors which are embedded as input words. At the next layer, using local information of each words in the lookup table, words selected as features. These features are not reasonable enough since each word affects differently in various issues and conditions. Accordingly, a higher level is essential to determine precisely the exact features. To this reason, we suggest to add a higher level for feature engineering by the third layer. This layer selects features considering sentences. Therefore, words situations and roles in the sentences are considered in this hierarchy of the model.

At the output layer, we apply an activation function which assign maximum value to the more probable class. Softmax is a normalized differentiable form of exponential function.

Besides, the probability of all output units will add up to one. If an N-dimensional vector is given, softmax transfer it to another N-dimensional space such that:

\[ s(\alpha) = \frac{e^{a_j}}{\sum_{k=1}^{N} e^{a_k}} \]  

(4)

where

\[ s_j = \frac{e^{a_j}}{\sum_{k=1}^{N} e^{a_k}} \]  

(5)

It can be written as a probabilistic function as below:

\[ s_j = p(y = j | \alpha) \]  

(6)

where \( y \) is the output class \( j, j=1, 2, ..., N \) and \( \alpha \) can be any vector.

From another point of view, softmax acts much like the max function and increases the probability of maximum value of outputs more than the others.

B. Parallelizing deep structure

Due to the computational load that deep learning based methods are faced, this study proposes to employ a parallelized structure. Parallel algorithms are divided and executed on several processing devices where combined together at the end of whole process [39].

The parallelizing procedure used in this article is a gradient descent approach minimizing the objective function. Since parallelizing gradient descent is time consuming, HOGWILD scheme, introduced by Niu et. al. [40], is used by H2O package. Using a shared memory, a system containing multiple cores or nodes manages subsets of data. Procedures are accomplished in parallel in each node or core and finally parameters \( W \) and \( B \) are calculated by averaging. These procedures operate on top of Hadoop and Spark which the data are compressed in-memory [34].
IV. Experiment and results

After processing reviews in a hierarchy of intelligent feature engineering, we compare the results with other common methods. In addition to obtain improvement in the performance of fake review detection, the results were achieved in a significant short time in comparison with other popular and usual methods. Here, we apply our model on two different datasets to illustrate the scalability of the model.

In this section we introduce the data set and due to the accuracy comparison in text classification, our model is compared to logistic regression, Naïve Bayes and Support Vector Machine methods.

A. Yelp dataset

Yelp's website presents a source of trade reviews which has web pages assigned to some places such as restaurants or schools, where its users can give commands or reviews on their products and services and rate them from one to five stars. It has surveyed that more than 20% of Yelp's reviews are deceptive [41] like other reviews on the Internet. Yelp website does not delete suspicious reviews but locates them in a list, which is public and available, however, avoids presenting them on the businesses' pages.

The challenge of detecting deceptive reviews is still an open problem. In this study, we use a supervised approach to utilize Yelp's filtered reviews. These reviews are restaurant reviews which gathered by [42].

We examine the proposed model for text classification on Yelp dataset to examine its performance, feature representation and decision making techniques.

Table 1 surveys the comparison of the proposed model on Yelp dataset. It demonstrates that our proposed model excels the others.

<table>
<thead>
<tr>
<th>Models</th>
<th>precision</th>
<th>recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>71.72</td>
<td>73.43</td>
<td>72.56</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>64.32</td>
<td>65.83</td>
<td>65.06</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>63.74</td>
<td>65.56</td>
<td>64.64</td>
</tr>
<tr>
<td>SDRN</td>
<td>90.13</td>
<td>85.70</td>
<td>87.60</td>
</tr>
</tbody>
</table>

Table 2. the execution process of SDRN log table

<table>
<thead>
<tr>
<th>Epoch</th>
<th>Training MSE</th>
<th>Training AUC</th>
<th>Training classification error</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>48.36</td>
<td>56.44</td>
<td>46.47</td>
</tr>
<tr>
<td>50</td>
<td>34.52</td>
<td>76.58</td>
<td>23.83</td>
</tr>
<tr>
<td>100</td>
<td>22.43</td>
<td>92.36</td>
<td>11.11</td>
</tr>
</tbody>
</table>

As mentioned before, precision is the fraction of retrieved terms which are relevant while recall is the fraction on relevant terms which are retrieved. High precision and high recall show our model effectiveness.

SDRN processing log is depicted in table 2. It demonstrates that in each iteration, the model evolves and probable features can be extracted to improve review spam detection performance.

B. Three-domain dataset

Li et al. [43] introduced a review spam data set which is a collection of truthful and deceptive reviews in three domains of Hotel, Restaurant and Doctor. They separated truthful reviews from deceptive ones in each domain.

Table 3 depicts the dataset. For each experiment, we measure the accuracy, precision, recall, and the F-measure among different classes. To evaluate various kinds of classifier in detecting deceptive reviews, we carry out a set of experiment on a combination of all three domains.

<table>
<thead>
<tr>
<th>Domains</th>
<th>Turker(positive/negative)</th>
<th>Customer (positive/negative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hotel</td>
<td>400/400</td>
<td>400/400</td>
</tr>
<tr>
<td>Restaurant</td>
<td>200/0</td>
<td>200/200</td>
</tr>
<tr>
<td>Doctor</td>
<td>356/0</td>
<td>200/0</td>
</tr>
</tbody>
</table>

Table 4 compares these results. Obviously, deep learning methods have a higher accuracy compared to other machine learning methods. The reason is because deep neural networks extract better features from the data that help discerning them.

<table>
<thead>
<tr>
<th>models</th>
<th>precision</th>
<th>recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>72.33</td>
<td>68.50</td>
<td>70.36</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>61.69</td>
<td>63.32</td>
<td>62.49</td>
</tr>
<tr>
<td>Log Regression</td>
<td>55.70</td>
<td>57.34</td>
<td>56.50</td>
</tr>
<tr>
<td>SDRN</td>
<td>77.69</td>
<td>72.92</td>
<td>75.23</td>
</tr>
</tbody>
</table>

V. Conclusion

A novel parallelized rectifier neural network is introduced in this article to learn document representation automatically and quickly for opinion spam detection. Using a hierarchy of feature extraction in a deep structure neural network, enables us to construct a semantic model in different resolution to detect fake reviews. Engineering features in sentence level in addition to the word level features produces features with higher quality. Utilizing a model with no need to manually feature extraction from documents with high accuracy, is
effective to improve the detection of spam comments. Experiment results illustrate our model superiority over others.

REFERENCES


[40] B. Recht, C. Re, S. Wright, and F. Niu, "Hogwild: A lock-free approach to parallelizing stochastic gradient descent,"


Learning classifier predictions: is this advantageous?

Vitaliy Tayanov  
Department of Computer Science and Software Engineering, Concordia university, Montreal, QC, Canada  
vttayanov@yahoo.com

Adam Krzyżak  
Department of Computer Science and Software Engineering, Concordia university, Montreal, QC, Canada  
krzyzak@cs.concordia.ca

Ching Suen  
Department of Computer Science and Software Engineering, Concordia university, Montreal, QC, Canada  
suen@cse.concordia.ca

Abstract—This paper presents a new machine learning approach for classification by ensemble of experts. Instead of learning pattern features or attributes which may represent some physical properties of an object or some other data we aim to learn the behaviour of the ensemble of classifiers which map feature space to decision space. To obtain good structure of the mapped data we need a set of independent experts. Such experts can be generated by Random Forests. In this case we do not need the set of strongly discriminatory features. The only requirement is to have a subset of features that help to generate less correlated classifiers. The basic belief supporting this idea is that it is better to learn the substance (black box) based on experience of the experts than to learn it directly from data. The experiments presented in the paper validate the stated hypothesis.

Index Terms—ensemble of classifiers, classifier independence, learning classifier predictions, feature mapping, feature strength

I. INTRODUCTION

Feature based classification and prediction is standard approach in pattern recognition and then machine learning. This concerns image/video recognition, stock predictions, behaviour classification, action recognition, etc [1]. All these problems require some descriptors/features which help to distinguish different categories. To this end features used should have strong discriminatory properties to help categorise given probes. However in many situations such descriptors may be contaminated by noise and noise influence may have non-liner nature. Most of real-world problems are complex enough not to be linearly separated and have very large overlapping regions. This makes it difficult to separate data even by means of powerful non-linear classifiers. Another problem which often occurs in practice is overfitting, which is one of the main problem in machine learning [2], [3]. To avoid overfitting different types of regularizers have been proposed: $L_2$ and $L_1$ regularisations for individual classifiers [4], bootstrap aggregation (bagging) for ensemble of classifiers [5], early stopping and dropout for deep neural networks [6], [7]. Notwithstanding all these methods cannot eliminate the effect of overfitting completely. Generally there is no always strong correlation between the loss to be minimised and the effect of overfitting completely. Generally there is no always strong correlation between the loss to be minimised and the accuracy of classification or prediction.

One general approach to deal with such kinds of problems is to use rejection-based algorithms [8]–[10]. Generally outliers should be removed from training data because they are not typical representatives and using them for training leads to unnecessary shifting of an algorithm parameters which is usually harmful for the learning algorithm. This shows that we may deal with rather complex data (with outliers or ambiguity examples) which demands more capacity from our algorithm. It is known that more complex algorithms overfit more. It is also better to have as little correlated classifiers as possible. In [11] it was proposed to use the agreement of classifiers to decompose the data into some functional groups. The principal idea behind it is that less complex data require less complex algorithms which should make it more robust to overfitting. According to this approach some part of data is classified by classifier agreement. When consensus is achieved and is correct then this part of data is the easiest part of the entire dataset. Some other part of data where consensus is not reached can be considered as ambiguous or difficult part of data. Some subsets of data classified by consensus agreement when agreement is wrong is called misclassification by classifier agreement and basically consists of examples that are close to outliers. in this paper we carried out a number of experiments confirming the effectiveness of such approach.

In this paper we propose a fundamentally different approach to supervised machine learning whereas instead of learning only from features we propose to learn from classifier predictions also. So after preliminary classification by an ensemble of classifiers predictions from each of classifiers of the ensemble play the role of features. After such mapping feature discriminatory strength has no direct influence on classification of new samples. To have more effective mapping it is good to have as little correlated classifiers as possible. There is a relationship between the number of samples, the number of original features and the number of estimators in ensemble. However in general case it might be hard to evaluate it. Classification pipelines using such approach are more complex in terms of computations that those using original feature sets. However if learning of new attributes is much faster than original ones then that compensates the extra effort needed to obtain predictions from the ensemble of classifiers. Simulation results presented in the paper validate our new proposed approach. The parameters of the classifiers have not been optimized so further improvements are possible.
II. LEARNING DATA MANIFOLDS

As it was shown in [11] there are three categories of patterns produced by a consensus of classifiers: easy patterns (those which are classified by consensus of classifiers), ambiguous patterns (those for which consensus has not been reached) and outliers (relatively small number of patterns where consensus was reached and it was incorrect). In the present paper we are not going to learn outliers because in such a case the structure of learned manifold becomes too complicated. Another reason is that we do not need to learn patterns that are not typical representatives of a corresponding class. To the best of our knowledge to date no research was carried out on the structure and learning capabilities of the manifold of rejected patterns. For learning manifolds some of eight geometries (spherical geometry $S^3$, Euclidean geometry $E^3$, hyperbolic geometry $H^3$, the geometry of $S^2 \times R$, the geometry of $H^2 \times R$, the geometry of the universal cover of SL(2, R), Nil geometry and Solv geometry) might be used [12]. The experiments carried out in [11] using AdaBoost [13] to learn a manifold of rejected patterns produced very promising results about possibility to learn such manifolds. The total error consists of the error produced by outliers and some part of ambiguous patterns classified incorrectly by AdaBoost. Thus the total error measured on five datasets from UCI repository is less than all those produced by AdaBoost, Random Forest, Rotation Forest, Random Subspace method and Random Oracle for every dataset [11], [14].

A. Mapping features to classifier outputs

One way to learn the optimal parameters of some classification algorithm is to use metalearning [15]. However for ensemble of classifiers instead of learning parameters of an algorithm we can learn the behaviour of the entire ensemble using classifier outputs. To make it more effective we can apply this scheme only for learning the manifold of rejected patterns. As it was mentioned before this allows to simplify the manifold complexity which should lead to better, faster and more effective learning. The only limitation in using this approach to datasets in UCI repository is that the available datasets are of a low scale or size.

In [14] it was suggested to use classifier in ensemble which are as less correlated as possible to get better accuracy. Having better diversity allows better learning of mutual behavior and interaction between classifiers. In [16] interaction between classifiers has been presented using pairwise classifications. Algebraically it can be represented by the appropriate matrices. To have independent classifiers means that their outputs or errors they produce are independent. At this point we do not need to have strong classifiers (i.e., classifiers yielding low error probabilities) anymore. The most important is to have independent classifiers to have “good” data for learning. This also might mean that having strong discriminatory features is not important anymore. Good features are those which lead to less dependent classifiers in the ensemble.

It is well known that each classifier performs mapping $X \mapsto Y_i$. In case of ensemble a feature vector for each pattern is mapped as follows $X \mapsto \bigcup Y_i$, where $Y_i$ is mapping $X \mapsto Y_i$. Here $i$ is the index of a classifier in the ensemble.

If classifiers in an ensemble are independent then we can map features into some quasi-orthogonal space of classifier outputs. The classifier selection problem is equivalent to determining the size of this quasi-orthogonal space. Continuing the optimization of classifier selection leads to dimensionality reduction problem [17], [18]. On the other hand we are going to test the prediction-based learning on binary classification problems due to the simplest and faster implementations. Classifier prediction space can be considered as a coordinate space where each coordinate correspond to a particular classifier. If all the classifiers are independent or almost independent this may be considered as orthogonal or quasi-orthogonal space. The example of metric to be used in that space is Hamming distance.

It is interesting to find out if discriminatory strength of feature vector has any influence on learning classifier outputs in an ensemble. Also it is important to determine the influence of classifier independence on learning ability of outputs from the classifier ensemble. If classifier independence is a dominating factor we do not need to worry about feature strength and focus only on generating independent classifiers in the ensemble.

III. EXPERIMENTAL RESULTS

We are going to compare feature-based learning with prediction-based learning obtained by Random Forests. To this end we use MLP with not too deep architecture to avoid overfitting. Here our objective is to find out if learning is faster or (and) better for prediction-based case.

![Blobs visualisation](image.png)

Fig. 1: 2D visualisation of two classes generated via blobs generator in python
Fig. 2: Losses for prediction-based and feature-based learning tested on different sizes of simulation data

Fig. 3: Accuracies for prediction-based and feature-based learning tested on different sizes of simulation data
The experimental protocol is as follows. To obtain our first results we do not use $q$-fold cross-validation to validate and compare two approaches. Instead we generated independent training and testing sets. For future more detailed experiments we are going to use 10-fold cross-validation if a dataset is rather small, 5-fold cross-validation for bigger sets and do not use cross-validation for large datasets as it is usually done in deep learning applications. Thus to obtain simulation data we used blobs generator for two labeled classes with multivariate normal distribution. The number of dimension is equal to 60 in all the experiments. The numbers of samples and the number of estimators change from experiment to experiment. The batch size is adopted for each data size. This allows to see the behaviour of both approaches for different scales of data.

The neural network which should be learnt from features and from ensemble predictions consists of five hidden layers with the number of units halved when passing to the next layer. The largest layer consists of 32 neurons. As loss function one uses binary cross-entropy. As activation function ‘RELU’ was applied [19] and optimization was performed with ‘rmsprop’ algorithm [20]. We used ‘BatchChNormalisation’ [21] to speed up the learning process. We do not use any regularization ($L_1$, $L_2$, dropout or early stopping) for this neural network architecture.

We used three sizes of data: 60, 600 and 6000 samples for training and 20, 60 and 600 for testing. For larger data sizes we took 10% of samples for testing. For the smallest size of data (60 samples) we used $\frac{1}{3}$ of corresponding samples for validation.

As seen from Fig. 2 generally prediction-based learning requires less time to learn the neural network model for all data sizes. Also it can be concluded from Fig. 3 that accuracy achieved using prediction-based learning is higher and more stable with training. Additional experiments aiming to measure the mean and variance of accuracy and loss as well as the number of epochs during training for both types of learning will be the subject of our future research.

IV. CONCLUSIONS AND FUTURE WORK

This preliminary research shows the advantage of using prediction-based learning. We can use less estimators than the size of feature set especially for data with smaller sizes. Faster learning and lower dimensional prediction space can be used to compensate some additional computations needed to generate the predictions. Some further optimization will be carried out in our future research as well as some theoretical analysis will be carried out concerning the relation between the number of predictors in ensemble, the number of original features and the number of samples. Also we are going to test our approach on difficult real-world data sets as well as explore more research directions to develop new approaches on how to use predictions to build tensors that might contain more information about patterns.

V. ACKNOWLEDGEMENTS

We appreciate the support from the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

Abstract—This article considers the cytological malignancy grading problem for fine needle aspiration biopsies of breast cancer. Previously, we proposed six different automatized cytological grading systems based on six published cytological grading schemes used by pathologists, each system designed to follow the cytological characteristics as defined by each of these schemes. In this paper, we seek to improve the performance of these cytological grading systems when used with an imbalanced data, resulting in an imbalanced classification problem. To improve the performance of the cytological grading systems when dealing with imbalanced data, we study the use of two data sampling techniques, oversampling (minority class) and undersampling (majority class), as well as the Hybrid RUSBoost ensemble-learning algorithm (which combines random undersampling and boosting techniques) to adjust the data distribution. The best overall accuracy result was 98.86% for case classification obtained by the adaptive boosting decision tree (DT+AdaBoost) with the oversampling technique for the computer-aided version of the Robinson’s, while the best overall accuracy result was 99.61% for patient classification obtained by the Support Vector Machine classifier with the oversampling technique for the computer-aided version of the Fisher’s cytological grading systems.

Index Terms—Malignancy Grading, Breast Cancer, Cytological Images, Fine Needle Biopsy, Imbalanced Data Classification.

I. INTRODUCTION

Breast cancer is the world’s most commonly diagnosed and deadliest form of cancer for women. According to American Cancer Society [1], in 2017, an estimated 252,710 new cases of invasive breast cancer are expected to be diagnosed among women in the United States along with about 2,470 new cases expected in men. An important and often used diagnostic method for breast cancer is the so-called triple-test, based on 3 medical examinations, that is used to achieve high confidence in the diagnosis. It includes self-examination (palpation), mammography, and Fine Needle Aspiration (FNA) Biopsy. FNA is an invasive examination that consists of obtaining material directly from the tumor with a fine needle (typically 18- to 22-gauge), sometimes with the aid of ultrasound. The collected material of the FNA is then examined under a microscope to determine the prevalence of cancer cells. Typically, two major obstacles in effectively treating breast cancer are its late detection and inaccuracy of the malignancy degree determination. Therefore, an important step is to accurately grade tumors so that the most appropriate medical regimen is selected, which requires extensive knowledge and experience of the pathologist responsible for the diagnosis. The most widely used systems for breast tumor grading are based on Bloom-Richardson (BR) histological grading [2]. The BR grading is defined particularly for histological images of thin slices of excised tumor biopsies where tissue structure is mostly preserved allowing for determination of tubule formation and cell nest structures.

On the other hand, for FNA the material is extracted by a needle and smeared, usually non-uniformly, on a glass to create a slide for cytological imaging. This may result in partial destruction of the tissue structure, and sometimes even of the nuclei of cells. Since this loss of tissue structure complicates the determination of tubule formation, combined with difficulties in scoring mitoses, cytological grading has been proposed instead of histological BR grading, based on cytological characteristics such as architecture, dissociation of cells, cell nuclear features, etc. Examples of cytological grading systems are Mouriquand’s grading [3], Robinson’s grading [4], Fisher’s modification of Black’s nuclear grading [5].
The need for accurate malignancy grading has led to increasing interest in computer-aided cytological image analysis. An automatic, objective malignancy grading can assist inexperienced or overworked and fatigued pathologists to avoid grading errors by providing an instant second expert opinion, particularly for uncertain cases that would require further manual examination by the specialist. To achieve this objective, in our previous study [9], six computer-aided cytologic malignancy grading systems were proposed for FNA biopsies based on the six published cytological grading schemas. Each system tailored to follow the cytological characteristics as defined by each of these published schemes. However, the difficulty we face relates to the imbalanced class data distribution in the raw dataset, where we have few samples belonging to the highest malignancy grade G3 (minority class) compared to the intermediate malignancy grade G2 (majority class). This high disparity of the number of patients between the two classes results in an imbalanced classification problem. In class imbalanced classification, traditional classifiers may be biased towards the majority class which might lead to poor predictive accuracy over the minority class. This poor performance regarding the minority class is undesirable, as they are often the classes we are more interested in (positive classes). Although class imbalance is a very serious problem in machine learning and data mining, a comprehensive study of how this problem affects the accuracy performance of most traditional classifiers is not completely clarified [10].

Our main objective of this study is to enhance the performance accuracy of the proposed cytological grading systems in [9], where they face difficulty related to the nature of the data used (imbalanced dataset). Thus, with the purpose of relieving the influence of the imbalanced classes distribution problem on classification accuracy, two data sampling and a hybrid ensemble-boosting approaches are performed in this study. The data sampling techniques include the use of oversampling or undersampling to achieve a balanced (50:50) data distribution. The oversampling technique overcomes imbalanced class data distribution by adding samples to the minority class by either duplicate the samples or add new samples. Whereas, the undersampling technique handles this problem by eliminating samples from the majority class. On the other hand, the RUSBoost boosting-based ensemble learning approach combines boosting (using AdaBoost algorithm) with random undersampling. This approach is designed basically to improve the performance accuracy of models trained on skewed data [11]. Thus, using this algorithm we can randomly remove samples from the majority class to balance the class data distribution as well as enhance the performance of the weak learners by the AdaBoost algorithm and using the new balanced datasets.

II. RELATED WORK

Imbalanced datasets present a significant challenge to the machine learning community. As we stated earlier, traditional classifiers may be biased towards the majority class which might lead to poor predictive accuracy over the minority class. Many attempts have been done to deal with unbalanced data in classification problem using different techniques such as data sampling, algorithmic level and ensemble learning. In the reviewed literature, we find several solutions proposed by the authors to investigate this problem using some of these mentioned methods. Batista et al. [12] did a comprehensive investigation and evaluation of different existing methods that deal with the problem of class imbalance. The study provided evidence that class imbalance problem does not systematically hinder the performance of learning systems. But it is a critical problem for the classification task if they share overlapping characteristics between classes. Krawczyk et al. [13] proposed an approach to describe and compare several state-of-the-art methods that are based on the oversampling approach to eliminate the disproportion among classes. Mazurowski et al. [14] examined the influence of imbalance data in simulated training dataset with the purpose of developing neural network (NN) classifiers for automated medical diagnosis system. The authors constructed the NN with two techniques involving classical backpropagation (BP) and particle swarm optimization (PSO). Based on the study results, even the low ratio of class imbalance in a training set could deteriorate the classifier performance. Guo et al. [15] conducted a study which included the combination of boosting and an ensemble-based learning algorithm. This algorithm generated the data to re-balance their original imbalance dataset which included two classes. With their method the samples with high correlation criteria (hard samples) from both the classes were identified during the boosting algorithm execution. Subsequently, the hard samples were used to generate independently synthetic samples for both classes.

In terms of breast cancer malignancy grading research, the following studies have given serious attention to the imbalanced class data distribution problem. Krawczyk et al. [16] presented an application of Adaptive Splitting and Selection (AdaSS) ensemble classifier to design an efficient clinical decision support system for breast cancer malignancy grading. A dedicated ensemble model was used to exploit local areas of competence in the decision space to combat imbalanced classes in the dataset, resulting in better accuracy. In another study of Krawczyk et al. [17], the authors proposed an automatic and comprehensive malignancy grading system for breast cancer to support the clinical decisions. They then utilized EUSBoost ensemble classifier which combines a boosting scheme with evolutionary undersampling technique to handle the problem of imbalanced class data distribution in the dataset. According to the study, the most significant samples for the classifier learning step can be selected using this evolutionary undersampling technique. Using a similar approach, in order to enhance the accuracy performance of the proposed six cytological malignancy grading systems for breast cancer in [9] where the systems were examined on an imbalanced dataset, in this study, we propose the use of the oversampling (minority classes), undersampling (majority classes) versus a Hybrid RUSBoost approach to adjust the
classes data distributions.

III. BREAST CANCER DATASET

To test the proposed classification schemes, we use a dataset, labeled as JELEN16, that was taken from 63 patients and includes 48 patients with G2 grade (comprising 133 cases of pairs of 100x and 400x images) and 15 patients with G3 grade (comprising 33 cases of pairs of 100x and 400x images). The dataset contains FNA biopsy images collected during examinations from the Department of Pathology and Oncological Cytology of the Medical University of Wrocław, Poland. Preparation of the slides includes staining with Haematoxylin and Eosin. Regions of interest on the slides were digitalized with the resolution of 96 dots per inch (dpi) and a size of 764x572 pixels. Each dataset consists of images obtained with two different magnifications (100x and 400x) of the same tissue region for each patient (see Fig.1). Each pair of images (100x and 400x) will be referred to as a single case for that patient. The malignancy grading for all the patients was histopathologically validated using surgical biopsies graded using BR grading [2].

IV. METHOD

We previously proposed computer-aided cytological grading systems for FNA slides of breast cancer based on the six cytological schemes [9]. In order to estimate the malignancy factors of these cytological grading schemes (CGS), sets of features that represent the cellular (from low magnification images) and nuclear (from high magnification images) characteristics were calculated for classification purpose. By converting the criteria of the CGS into classification problems, the proposed frameworks are able to evaluate and assign a malignancy grade (G2 or G3) to an FNA slide. As mentioned before, the main objective of this study is to enhance the performance accuracy of the proposed cytological grading system in [9]. Thus, before classification stage, an adjustment of the class data distribution is necessary. To archive this target we consider the use of the oversampling and undersampling techniques, as well as the RUSBoost algorithm.

A. Image Preprocessing

Originally, the FNA slides are saved as three channel images, red-green-blue (RGB). This means the channel components are highly correlated with respect to different medical stains meaning it is difficult to obtain accurate image segmentation. The FNA in our datasets have Hematoxylin-Eosin staining where Hematoxylin (blue) mainly stains the cell nuclei and Eosin (magenta-red) acts as a cytoplasmic stain. To aid image segmentation, as a pre-processing step, we performed color deconvolution [13] to convert the RGB channels into Hematoxylin and Eosin channels to, in effect, extract nuclei and cytoplasm layers. To improve the quality of the extracted Hematoxylin layer, we adjusted the intensity values using contrast enhancement. We then computed multilevel thresholds for the adjusted image using Otsu’s method and applied the quantization process using the estimated threshold values to segment the image into three regions represented by distinct colors. We then converted the resulting image into a new RGB image. Finally, since each active contour method (geometric or parametric) requires an initial curve, we used the boundary between the regions in the green channel of the extracted image as initial contours.

B. Image Segmentation

After the pre-processing step, we perform image segmentation. Three phases are involved at this stage: Nuclei segmentation, nuclei filtration and cytoplasm segmentation as follows:

Nuclei segmentation: To perform nuclei segmentation, as a first stage, we use a combination of Gradient Vector Flow (GVF) snake algorithm and morphological operations (MO) (which we refer to as GVF-MO) to segment the nuclei regions. As in the work of Malek et al. [19], Active Contour or GVF-Snake has been used. As a second stage of the nuclei segmentation, to separate individual nuclei from clusters of connected nuclei that are indistinguishable to GVF-MO, we utilized a marker-controlled watershed segmentation (WS) algorithm based on mathematical morphology; see Yang et al. [20]. WS algorithm effectively segments the clusters of connected nuclei regions based on the nuclear size with less over-segmentation results. Specifically, we examined the nuclei sizes criterion and supposed that if the nuclei sizes are bigger than a specific, experimentally determined Nuclear Size (NS), we assumed those sizes belong to clusters of nuclei. So, as a stage two, we applied the WS algorithm to re-segment those nuclei clusters into individual nuclei.

Nuclei filtration: In the cytological images, some of the nuclei clusters did not re-segment completely by applying the WS algorithm due to the poor quality of some of the images, presence of red blood cells, etc. Further, the presence of red blood cells and false positives results caused by geometric arrangements in the background being incorrectly identified as the boundary of a nucleus. To avoid this problem, we separated the resultant segmented nuclei as well-segmented (used for feature extraction and classification purpose) or poorly-segmented (ignored results) nuclei. To do this, we adapted the nuclei filtration procedure as used by Filipczuk et al. [21], applying the procedure to the results of the watershed method rather than the Circle Hough Transform First, for all the segmented nuclei regions seven nuclear features were calculated: Euler number, size, entropy, homogeneity, energy, correlation and standard deviation (uniformity of nuclei). A support vector machine (SVM) classifier, with a Gaussian radial basis kernel function, was trained using these features to classify the segmented regions as well or poorly segmented nuclei. Using JELEN16 the obtained classification accuracy was 80.24% and has been evaluated using 5-fold cross-validation technique. The manually prepared dataset contained 2211 nuclei regions including 1273 and 938 nuclei regions, that represented the well and poorly segmented nuclei, respectively.

Cytoplasm Segmentation: In the second phase of segmentation, we used GVF-MO combination to segment the cytoplasm regions using the Hematoxylin layer image. When Eosin
channel has been used to extract the cytoplasm layer the results were not good. Thus, only Hematoxylin channel was used in this study for both nuclei and cytoplasm layers extraction. We introduced this stage of cytoplasm segmentation as in some of the cytological grading schemes (Fisher’s [5] and Taniguchi et al. [8]) some of the cytological characteristics have been included (namely, nuclear chromasia and nuclear-cytoplasmic ratio). The well-segmented nuclei, as well as their cytoplasm regions, have been used in the next stage of feature extraction to estimate sets of different cellular and nuclear features.

C. Feature Extraction

The feature extraction involves two phases. In the first phase, to reflect the cell dissociation and cellular characteristics criteria that is present in Robinson’s and Mouriquand’s schemes respectively, we extracted a set of three structural features (average area of groups, number of groups and dispersion) from low magnification images (100x) for each case. These features were able to characterize the cells’ ability to form clusters or to disperse around the image as proposed by Jeleń et al. [22].

In the second phase, we focused on evaluating different nuclei characteristic features that are required in all the six developed grading frameworks. To achieve this target, we divided this phase into three steps to estimate three different sets of nuclear features. These features are derived from high magnification images (400x) only and represent the nuclear polymorphic, textural and morphologic characteristics. They are able to provide accurate information about the shape, size and staining of cell nuclei. The 5 nuclear polymorphic features [22, 23] are estimated to reflect the nuclear size, anisonucleosis (morphological manifestation of nuclear injury), cellular pleomorphism, nuclear pleomorphism and nuclear features malignancy factors of the discussed grading schemes. On the other hand, in order to calculate nuclear textural features, we used Gray-Level Co-occurrence Matrix (GLCM) and gray-level run-length matrix (GLRLM) as applied by Filipczuk et al. [21]. These features were able to reflect the characteristics of nucleoli, nuclear chromatin, nuclear chromasia (measures the dark staining of nucleus due to increased DNA content), chromatin pattern and chromatin granularity malignancy factors of these grading schemes. We estimated 10 textural features from first matrix and 11 from the second (21 in total). The last estimated group of features was a set of 16 nuclear morphologic features that reflected the nuclear margin (measures the irregularity of nuclear envelope due to its distortion), cell uniformity, nuclear membrane, cellular size and cell uniformity. The mean and variance were then calculated for each of the mentioned features (polymorphic, textural and morphologic) giving a total of 84 different nuclear features. All these 84 features have been used with all the cytological grading schemes except for Howell’s scheme [6] that used 24 features including 12 polymorphic and 8 morphological nuclear features out of 84 nuclear features plus the mitotic count and 3 cellular features obtained from the low magnification images. The other major challenge was related to some of the cellular characteristics that measure the variety of change in size and shape of the cancerous cells. We estimated the cellular criteria in terms of their nuclei characteristics variability, i.e., we evaluated the cellular criteria based on their nuclei change that occurs during the gradual transformation of normal cells to cancerous cells in the breast.

The third challenge was estimation of the mitotic count malignancy factor. To estimate the mitotic count malignancy factor we used the methodology of Irshad [24]. Thus, we calculated 25 morphological and textural features from the five mentioned channels in Irshad study [24]: 18 textural features were estimated using the GLCM and GLRLM matrices and 7 morphological features (perimeter, uniformity of nuclear size, uniformity of nuclear shape, nuclear irregularity, nuclear smoothness and nuclear roundness) were added based on our experimental results. We then used these features to classify the candidates as mitosis, non-mitosis and ignored candidates. Using an SVM with the Gaussian radial basis kernel function, the classifier has been trained on a manually prepared dataset of 1853 candidates that consisted of 129, 203 and 1521 mitosis, non-mitosis and ignored candidates, respectively. The accuracy of the mitosis classification has been evaluated using 5-fold cross-validation and Receiver Operating Characteristic (ROC) curve. The obtained accuracies were 65.22% and 78.26% for image and patient mitosis count, respectively.

Finally, our last challenge was naked tumor cell nuclei (NTCN) estimation which required some information that could be provided by direct observation (ground truth) such as the standard nuclei size which is difficult to determine using FNA images. Even though some of the estimated polymorphic and morphological features estimate the naked characteristics which have been defined implicitly, we estimated the highly atypical nuclear changes in size using the ratio of the nuclei number with very large and very small sizes to the total number of nuclei.

D. Feature Selection

Feature selection plays a central role in helping to reduce the high-dimensionality and noise of the data by removing redundant and irrelevant features. After experimenting with several feature selection methods we chose Fisher method as reported by Roffo [25] because it found more relevant and non-redundant subset of features in addition to giving better classification results.

E. Classification

To determine the malignancy level of FNA biopsies, we used nine different classifiers that take a feature vector as an input and return one of the two malignancy grades as an output. The classifiers used were Linear Discriminant Analysis (LDA), Feedforward Neural Network (FFNN), SVM, MLP, Decision Tree (DT), Adaptive Boosting (AdaBoost) of Decision Trees (DT-AdaBoost), Random Forest of Decision Trees (RF), Naive Bayes (NB), and K-Nearest Neighbors (KNN) [26]. Two classification schemes were used, case classification and patient classification [23], as follows:
Case classification: In this scheme, a feature vector is independently computed for each pair of images with high and low magnifications that belongs to each patient, and each case is classified separately. The results of the case classification represent multiple classification results for the patient.

Patient classification: To classify a certain patient as G2 or G3 in this scheme, the final classification result is achieved by majority voting of the classification results of the individual cases for that patient.

Before the classification stage, adjust the class data distribution by re-balancing the two classes equally is necessary. Thus, we applied the oversampling and undersampling techniques as well as the RUSBoost algorithm in this study. And then we compared the newly obtained results of this study with the old results achieved previously in [9]. The comparison has been done based on the archived overall average accuracy and the other estimated measures (specificity, etc) which used in both studies to evaluate the performance of each classifier.

V. EXPERIMENTAL RESULTS

In this section, we present simulation results for the six computer-aided malignancy grading frameworks for FNA biopsies of breast cancer after adjusting the class data distribution of the data used as well as comparing these results with the results obtained in [9]. In both studies, for nuclei segmentation, the parameter NS was chosen experimentally on the test set of 400x magnification images and set to 100 pixels. For feature selection process, for each implemented cytological scheme, using Fisher method, the final feature vectors were reduced by 70%. Also, k-fold cross-validation with k = 5 was used to divide the raw dataset as well as the training subset. Further, the authors averaged 30 runs for each classifier to confirm the obtained results. During the simulations the images for any given patient were used for either training or testing, but not both. Moreover, the 95% confidence intervals were calculated for the obtained classification accuracies.

In [9], the best classification results were obtained for the Support Vector Machine classifier using the JELEN16 (imbalanced dataset) for computer-aided versions of the Robinson’s and Khan et al.’s cytological grading systems with accuracies of 97.57% and 96.98% for case classification (where a case is a pair of 100x and 400x magnification images for a patient) and 95.23% and 98.36% per patient classification, respectively. In this study, our main objective is to eliminate the imbalanced data distribution of the JELEN16 dataset to enhance the overall performance accuracy of the proposed cytological grading systems in [9] using the data re-sampling and RUSBoost approaches. The considered dataset consists of 266 samples, where 33 of the samples belong to the MG3 (positive class), while 133 samples belong to the MG2 (negative class). After applying the mentioned balancing data techniques to adjust the class data distribution to a (50:50) ratio among the two classes, the new training dataset using the oversampling technique was 266 samples, while the new training dataset was 66 samples using the undersampling technique. With the usage of the oversampling technique, there is no loss of information but the training time of the classifiers has increased due to the added samples. Also, it can lead to overfitting problem. Whereas the usage of the undersampling technique results in a loss of information due to the removed samples but the training time of the classifiers has decreased. According to obtained results, we found that oversampling technique has worked overall the best for the dataset used in this study. On the other hand, though the random undersampling technique in the RUSBoost approach leads to loss of information, the boosting algorithm implicitly handles this problem [11]. The only side effect that we noticed with this algorithm is that it needs long training time that required to train the boosting-ensemble. As a comparison between the two data sampling techniques and the RUSBoost approach, the experimental results show that the data sampling techniques outperformed the average accuracy of the RUSBoost approach by at least 8% for all the six cytological grading systems as shown in the figures [1-4] (with 95% confidence intervals indicated) that represent the best two systems, over all the classifiers considered, for the case and patient classification, respectively.

To evaluate the robustness of the proposed frameworks on the test subset, we used the confusion matrix to compare the two classification schemes (case and patient schemes). We computed the average of the accuracy as well as the three other measures, that are, the sensitivity, specificity and precision rates (see Tables [1, 2, 3, and 4]). The tables include the evaluation results of the best two systems for each of the case and patient classification.
In Tables [I] and [II] for case classification, we expand the comparison between the best two schemes of Robinson’s and Khan’s and the other computer-aided classification schemes based on the six six classification results using the JELEN16 dataset after adjusting the class distribution.

In Tables [III] and [V] for patient classification, we expand on the comparison between the best two schemes of Fisher’s and Khan et al.’s and the other computer-aided classification schemes based on the six classification results using the JELEN16 dataset after adjust class distribution.

According to the obtained results, the best two accuracies for case classification (where a case is a pair of 100x and 400x magnification images for a patient) were 98.86% and 98.67% obtained from the adaptive boosting decision tree (DT+AdaBoost) with the oversampling technique for computer-aided versions of the Robinson’s and Khan et al.’s, respectively. The best two accuracies of 99.61% and 99.00% for patient classification were obtained by the SVM classifier with the oversampling technique for the computer-aided version of the Fisher’s cytological grading system and for the computer-aided version of the Khan et al.’s cytological grading system, respectively.

VI. CONCLUSIONS AND DISCUSSION

In this paper, we compared the RUSBoost ensemble-learning algorithm and two data sampling techniques to overcome the difficulty that occurs with the imbalanced classification problem of breast cancer cytological malignancy. We found the data oversampling technique worked best and it was able to enhance the overall performance accuracy of all the six proposed cytological grading schemes in [9].

REFERENCES

[7] M. Khan, A. Haleem, H. Al Hassani, and H. Kfoury, “Cytopathological grading, as a predictor of histopatho-
Sensitivity

<table>
<thead>
<tr>
<th>Classification algorithms</th>
<th>Average accuracy</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM-OS</td>
<td>99.61%</td>
<td>99.33%</td>
<td>99.93%</td>
<td>99.93%</td>
</tr>
<tr>
<td>RF-OS</td>
<td>98.02%</td>
<td>98.24%</td>
<td>97.77%</td>
<td>98.06%</td>
</tr>
<tr>
<td>DT+AdaBoost-OS</td>
<td>90.88%</td>
<td>82.88%</td>
<td>98.88%</td>
<td>97.77%</td>
</tr>
<tr>
<td>DT-OS</td>
<td>96.18%</td>
<td>95.03%</td>
<td>97.50%</td>
<td>97.76%</td>
</tr>
<tr>
<td>ANN-OS</td>
<td>93.98%</td>
<td>95.13%</td>
<td>92.97%</td>
<td>92.30%</td>
</tr>
<tr>
<td>MLP-OS</td>
<td>93.81%</td>
<td>94.86%</td>
<td>92.90%</td>
<td>92.18%</td>
</tr>
</tbody>
</table>

TABLE III

Evaluation results of the first best patient classification for the best six classifiers using the JELEN16 dataset for Fisher’s cytological grading frameworks. OS - Oversampling dataset.

Specificity

<table>
<thead>
<tr>
<th>Classification algorithms</th>
<th>Average accuracy</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM-US</td>
<td>99.00%</td>
<td>98.88%</td>
<td>99.11%</td>
<td>99.92%</td>
</tr>
<tr>
<td>DT+AdaBoost-OS</td>
<td>98.70%</td>
<td>99.39%</td>
<td>97.91%</td>
<td>98.26%</td>
</tr>
<tr>
<td>RF-OS</td>
<td>98.64%</td>
<td>98.78%</td>
<td>98.47%</td>
<td>98.67%</td>
</tr>
<tr>
<td>ANN-OS</td>
<td>97.50%</td>
<td>97.29%</td>
<td>97.60%</td>
<td>97.39%</td>
</tr>
<tr>
<td>MLP-OS</td>
<td>96.79%</td>
<td>96.25%</td>
<td>97.27%</td>
<td>96.87%</td>
</tr>
<tr>
<td>DT-US</td>
<td>96.11%</td>
<td>92.66%</td>
<td>95.55%</td>
<td>99.58%</td>
</tr>
</tbody>
</table>

TABLE IV

Evaluation results of the second best patient classification results for the best six classifiers using the JELEN16 dataset for Khan’s cytological grading frameworks. US - Undersampling dataset; OS - Oversampling dataset.

Precision

<table>
<thead>
<tr>
<th>Classification algorithms</th>
<th>Average accuracy</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM-US</td>
<td>99.00%</td>
<td>98.88%</td>
<td>99.11%</td>
<td>99.92%</td>
</tr>
<tr>
<td>DT+AdaBoost-OS</td>
<td>98.70%</td>
<td>99.39%</td>
<td>97.91%</td>
<td>98.26%</td>
</tr>
<tr>
<td>RF-OS</td>
<td>98.64%</td>
<td>98.78%</td>
<td>98.47%</td>
<td>98.67%</td>
</tr>
<tr>
<td>ANN-OS</td>
<td>97.50%</td>
<td>97.29%</td>
<td>97.60%</td>
<td>97.39%</td>
</tr>
<tr>
<td>MLP-OS</td>
<td>96.79%</td>
<td>96.25%</td>
<td>97.27%</td>
<td>96.87%</td>
</tr>
<tr>
<td>DT-US</td>
<td>96.11%</td>
<td>92.66%</td>
<td>95.55%</td>
<td>99.58%</td>
</tr>
</tbody>
</table>

logical grade, in ductal carcinoma (NOS) of breast, on air-dried Diff-Quik smears,” *Diagnostic Cytopathology*, vol. 29, pp. 185–193, 2003.


A generalized unified discrete linear method for edge detection by antisymmetric FIR kernels

A. Raji
Laboratory of Images, Signals and Intelligent Systems
Paris Est Creteil Val de Marne University
F94010, Creteil, France
raji@u-pec.fr

Abstract—In this paper, we present an optimal linear edge detection method. Using an adequate discrete model for noiseless edges, a preliminary theoretical analysis that tries to analytically express the implementation of the detection steps leads to an antisymmetrical impulse response as generic solution. Then, considering for noisy edges the separability between the edge and the noise response components in the filter output, we derive a new parametric edge detection operator. Unlike the methods developed theoretically in continuous form then discretized or approximated by discrete solutions for application, which often results in a loss of performance between continuous and discrete models such as in Canny approach, the proposed method is entirely conceived in discrete form, which is more suitable and appropriate for digital signal and image processing applications. An optimization example is given, the obtained results illustrate the consistency of the proposed approach which could be declined in a number of solutions with different optimization criteria. The parametric form of the proposed method offers the opportunity to adapt the detection performance to the image characteristics such as the noise type and intensity.

Index Terms—edge detection, discrete linear filtering, FIR

I. INTRODUCTION

Transitions in signals and borders between regions or objects in images can constitute information of major importance in many applications. The theme of edge detection has known an increasing scientific interest during last decades in both research and applied engineering fields. However, it suffers from lack of models that explain theoretically the task in sufficiently clear and comprehensive manner and lead to straightforward and easy implementation in practice. Various edge detection approaches were proposed in literature over the five past decades. The most recent ones propose a learning-classifier planned technique [1]–[3]. Earlier methods use classical techniques such as order statistics [4], fuzzy logic [5], morphological gradient [6], deformable models [7], thresholding techniques [8], linear filtering [9], [10] and numerical computation [11].

The most commonly used linear methods can be classified in three categories. Earlier differential methods, such as Prewitt and Roberts operators, simply give numerical approximations of the image gradient. Since the derivative operation amplifies the high frequency components, including noise generally, a smoothing operation in the direction orthogonal to that of the detection is introduced in Sobel operator in order to reduce differential detectors sensitivity to noise. In the second category, the edge detection issue has been approached in terms of selective frequency filtering. This led to various Gaussian filtering based techniques [2], which unfortunately did not universally solve the problem because edges generally composed of different kinds of variations are not limited to a particular frequency band. Finally, an interesting approach posed the problem in terms of optimization of detection and localization objectives [6]. However, while the related theoretical study used an analogical model, the optimal operator have been determined by numerical computation before being approximated by the first derivative of a Gaussian filter as optimal solution, which does not preserve the same performance rates as the original numerical solution. In the present work we propose an approach where both of the edge and the detection method are entirely introduced within a discrete analytical scheme in an attempt to provide a unifying framework including the previously cited methods and allowing to find others by mean of additional specific criteria.

II. THEORETICAL ANALYSIS

A. Principle and modelization

For simplicity, we will present the method in one dimensional case. Edge models usually suggested in literature generally define a step edge by a two level function \( u(t) \) with an abrupt change occurring at a given point:

\[
    u(t) = \begin{cases} 
    0 & \text{if } t < 0 \\
    A & \text{if } t \geq 0 
    \end{cases}
\]  

(1)

Beyond the advantage that it gives an analytical description of the edge, we can point out some drawbacks of this model. On the one hand it does not represent a true edge because of its discontinuity, on the other hand the relevance of such a choice is not coherent with the final implementation generally achieved in discrete form, in particular for image processing applications. We propose instead of this the following discrete model defined by:

\[
    x(k) = \begin{cases} 
    0 & \text{if } k \leq -1 \\
    A & \text{if } k \geq 0 
    \end{cases}
\]  

(2)

where the signal transition from the low level 0 to the high level A is localized at two points \((-1, 0)\). The transition separates the signal \( x(k) \) into two regions, one to the right and one to the left of the edge position \((-1, 0)\). This implies
a symmetry in the perception of both regions which represent from edge detection point of view two symmetrical entities with regard to the edge position, equivalent by their semantic representation even if they have different algebraic values. From this consideration, an ideal edge detector would give a symmetrical binary response with magnitude 1 on the edge points \((-1, 0)\) and 0 everywhere else, as illustrated in Fig. 1.

The antisymmetry property (6) of these solutions is directly resulting from the proposed edge model and the detection objective formulation while it is an implicit property and even a symmetry constraint can be expressed by:

\[
y(0) = y(-1) \\
y(1) = y(-2) \\
\vdots \\
y(k) = y(-1 - k), \quad k \geq 0
\]

The response samples are given by:

\[
y(k) = \sum_{l=-\infty}^{+\infty} x(l) h(k - l) = A \sum_{l=0}^{+\infty} h(k - l)
\]

or by taking \(i = k - l\):

\[
y(k) = A \sum_{i=-\infty}^{k} h(i)
\]

The symmetry constraint (3) implies for \(k \geq 0\):

\[
y(k) - y(-1 - k) = 0 \\
\iff \sum_{i=-\infty}^{k} h(i) - \sum_{i=-\infty}^{1+k} h(i) = 0
\]

Applying (5) at different positions gives:

\[
k = 0 \longrightarrow h(0) = 0 \\
k = 1 \longrightarrow h(-1) + h(1) = 0 \implies h(-1) = -h(1) \\
\ldots \\
k \geq 1 \longrightarrow h(-k) = -h(k)
\]

which leads to an antisymmetrical linear filter \(h\) satisfying:

\[
\begin{cases}
  h(0) = 0 \\
  h(-k) = -h(k) \quad \text{for} \quad k \geq 1
\end{cases}
\]

We obtain thus a set of linear filtering solutions to detect the edge as a center of symmetry in the filter response. Notice the similarity of this approach with that used in the Canny study where the contour is identified as a local maximum of the filter response. The anti-symmetry property results here directly from the used model while it is an intrinsic property of differentiation-based detectors and a prior assumption in Canny’s approach.

Obviously, an edge detector can not be really useful if it is not robust in the presence of noise. We consider then a step edge disturbed by centered additive white noise \(n\) of variance \(\sigma^2\). Our goal is to select from the set (6) linear filters able to perform the edge detection according to the scheme proposed in Fig. 1 despite the presence of noise. We will try to achieve this by using the separability between the responses to the contour and noise considered separately. Indeed, the filter response to the noisy input \(e = x + n\) given by \(y = e \otimes h\) can also be computed by \(y = y_1 + y_2\) where \(y_1 = x \otimes h\) and \(y_2 = n \otimes h\). So, let us first characterize separately the two filter response components \(y_1\) and \(y_2\). The response of the filter (6) to the noise is given by:

\[
y_2(k) = \sum_{l=-\infty}^{+\infty} h(l) n(k - l)
\]

Since \(h(0) = 0\) and \(h(-l) = -h(l)\) for \(l \geq 1\),

\[
y_2(k) = \sum_{l=-\infty}^{l=-1} h(l) n(k - l) + \sum_{l=1}^{l=+\infty} h(l) n(k - l)
\]

\[
= \sum_{l=-\infty}^{l=-1} -h(-l) n(k - l) + \sum_{l=1}^{l=+\infty} h(l) n(k - l)
\]

By taking \(i = -l\), \(\sum_{l=-\infty}^{l=-1} -h(-l) n(k - l) = \sum_{i=1}^{i=+\infty} -h(i) n(k + i)\), then:

\[
y_2(k) = \sum_{i=1}^{i=+\infty} -h(i) n(k + i) + \sum_{i=1}^{i=+\infty} h(i) n(k - i)
\]

\[
y_2(k) = \sum_{i=1}^{i=+\infty} h(i) (n(k - i) - n(k + i))
\]
The filter response to the noise can then be characterized by its mean and variance given respectively by:

\[
\mu_2 = E\{y_2(k)\} = 0
\]

\[
\sigma_2^2 = \sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j) [r(k-i,k-j) - r(k-i,k+j) + r(k+i,k+j) - r(k+i,k-j)]
\]

\(r(i,j)\) representing the autocorrelation function of the noise \(n(k)\), which we suppose to be independent and identically distributed, with:

\[
r(i,j) = E\{n(i)n(j)\} = \begin{cases} 0 & \text{if } i \neq j \\ \sigma^2 & \text{if } i = j \end{cases}
\]

then:

\[
\sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j) r(k-i,k-j) = \sigma^2 \sum_{i=1}^{+\infty} h^2(i)
\]

\[
\sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j) r(k+i,k+j) = \sigma^2 \sum_{i=1}^{+\infty} h^2(i)
\]

\[
\sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j) r(k-i,k+j) = 0
\]

\[
\sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j) r(k+i,k-j) = 0
\]

which gives:

\[
\sigma_2^2 = 2\sigma^2 \sum_{i=1}^{+\infty} h^2(i)
\]

On the other hand, the filter response to the transition alone is given by (4):

\[
y_1(k) = A \sum_{i=-\infty}^{k} h(i)
\]

As the response \(y_1(k)\) is symmetrical with respect to the positions \((-1,0)\) as described in (3), we consider its right side, e.g. for \(k \geq 0\), where we can write:

\[
y_1(k) = A \left( \sum_{i=-\infty}^{-k-1} h(i) + \sum_{i=-k}^{k} h(i) \right)
\]

and using the antisymmetry property of the filter \(h\) we have \(\sum_{i=-k}^{k} h(i) = 0\), then:

\[
y_1(k) = A \sum_{i=-\infty}^{-k-1} h(i) = -A \sum_{i=-\infty}^{-k-1} h(-i)
\]

or with \(j = -i:\)

\[
y_1(k) = -A \sum_{j=k+1}^{+\infty} h(j)
\]

for \(k \geq 0\).

Since the ideal operating mode expected with the noiseless model is not realistic, we should choose among the solutions proposed in (6) one that is effective in the presence of noise. From the above results, the next phase should be the introduction of a criterion to minimize the operator sensitivity to noise. This criterion could be simple, expressed in terms of signal to noise ratio, or more complex, in terms of detection and localization performance for example. The present article does not solve definitely this question with multiple possible answers. Indeed, assuming multiple solutions might even be an opportunity to adapt the operator to different problems in terms of image and noise types. We present hereafter an approach example trying to reduce the sensitivity to noise.

B. Implementation as a parametric linear filter

The filtering solutions (6) were designed such that the response \(y_1(k)\) is symmetrical around the positions \((-1,0)\):

\[
y_1(k) = y_1(-1-k), \quad k \geq 0
\]

The expression (7) shows clearly that the noise component \(y_2\) of the filter output is not symmetrical because it is random. Consequently, the total response \(y = y_1+y_2\) is not symmetrical and the transition points \((-1,0)\) cannot be identified as the center of symmetry of the filter output. In order to limit this noise effect, we introduce two additional conditions. First, we search for a filter of type (6) that produces a response \(y_1\) which is maximal at the edge points \((-1,0)\) and symmetrically decreasing on both sides of the edge. Second, the response \(y_2\) due to the noise should not affect significantly the desired behaviour of \(y_1\) such as a subsequent thresholding of the total response \(y = y_1+y_2\) will allow to restore the transition points \((-1,0)\) as the center of symmetry of the thresholded filter output. Using the symmetry property, we consider the half right-side of the response and define for \(k \geq 0:\)

\[
\triangle(k) = y(k) - y(k+1) = \triangle_1(k) + \triangle_2(k)
\]

where \(\triangle_1(k) = y_1(k) - y_1(k+1), \triangle_2(k) = y_2(k) - y_2(k+1)\). Using the noise and signal response components expressed in (7) and (11) respectively, we have:

\[
\triangle_2(k) = \sum_{i=1}^{+\infty} h(i)[n(k-i) - n(k+i) + n(k+1+i) - n(k+1-i)]
\]

\[
\triangle_1(k) = -A \sum_{j=k+1}^{+\infty} h(j) + A \sum_{j=-\infty}^{-k-1} h(j)
\]

\[
\triangle_1(k) = -A h(k+1)
\]

The noise effect on the desired behaviour of the total response \(y(k)\) can be reduced by minimizing \(|\triangle_2(k)|\) and maximizing \(|\triangle_1(k)|\) at \(k = 0\) (and symmetrically at \(k = -1\) for the half left-side of the filter response). From (12) it follows that \(\triangle_2(k)\) is random with zero mean \(E\{\triangle_2(k)\} = 0\), so we can reduce it by minimizing its variance. The two aims can be reached simultaneously by maximizing the parameter:

\[
D(k) = \frac{\triangle_1^2(k)}{Var\{\triangle_2(k)\}}
\]
where:
\[
Var\{\Delta_2(k)\} = 2\sigma^2 - 2E\{y_2(k)y_2(k + 1)\}
\]
\[
E\{y_2(k)y_2(k + 1)\} = \sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j)\left[r(k - i, k + 1 - j) + r(k + i, k + 1 - j) - r(k + i, k + 1 + j) - r(k - i, k + 1 + j)\right]
\]
\[
\sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j)r(k - i, k + 1 - j) = \sigma^2 \sum_{i=1}^{+\infty} h(i)h(i + 1)
\]
\[
\sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} h(i)h(j)r(k + i, k + 1 - j) = \sigma^2 \sum_{i=1}^{+\infty} h(i)h(i + 1)
\]

Thus:
\[
E\{y_2(k)y_2(k + 1)\} = 2\sigma^2 \sum_{i=1}^{+\infty} h(i)h(i + 1)
\]

and, using (9):
\[
Var\{\Delta_2(k)\} = 4\sigma^2 \sum_{i=1}^{+\infty} h^2(i) - 4\sigma^2 \sum_{i=1}^{+\infty} h(i)h(i + 1)
\]

This implies:
\[
D(k) = \frac{A^2}{4\sigma^2} \frac{h^2(k + 1)}{\sum_{i=1}^{+\infty} h^2(i) - \sum_{i=1}^{+\infty} h(i)h(i + 1)}
\]  \hspace{1cm} (14)

At \(k = 0\):
\[
D(0) = \frac{A^2}{4\sigma^2} \frac{h^2(1)}{\sum_{i=1}^{+\infty} h^2(i) - \sum_{i=1}^{+\infty} h(i)h(i + 1)}
\]

Setting to zero the derivatives of \(D(0)\) with respect to the impulse response samples:
\[
\partial D(0)/\partial h(1) = 0 \text{ leads to } h(1) = \frac{2}{h(2)} \left( \sum_{i=2}^{+\infty} h^2(i) - \sum_{i=2}^{+\infty} h(i)h(i + 1) \right)
\]  \hspace{1cm} (15)

and \(\partial D(0)/\partial h(k) = 0\) for \(k > 1\) leads to:
\[
h(k) = 2h(k + 1) - h(k + 2)
\]  \hspace{1cm} (16)

which can also be expressed by:
\[
h(k) = \frac{1}{2} \left(h(k - 1) + h(k + 1)\right) \text{ for } k \geq 2
\]  \hspace{1cm} (17)

Again, and despite the additional constraints to make the filter robust in the presence of noise, we do not obtain a unique solution, but a family of solutions defined by (16) or equivalently by (17). The samples of \(h\) are defined relative to each other, the sample \(h(1)\) satisfies both of (15) and (16). The obtained filtering solutions are of infinite length. However, in practice operators of finite size are more suited for usual signal and image processing applications.

We can define filters of finite length \(M = 2m + 1\) by the following algorithm which consists to assign zero to the samples \(h(k > m)\), initialize the sample \(h(m)\) with an arbitrary value \(a\) and deduce the other samples of \(h\) using (16).

\[
h(k > m) = 0
\]
\[
h(m) = a
\]
\[
h(1 \leq k \leq m - 1) = 2h(k + 1) - h(k + 2)
\]
\[
h(0) = 0
\]
\[
h(k < 0) = -h(-k)
\]

We obtain thus a family of parametric linear filters where the parameters \(m\) and \(a\) represent respectively the size and the last sample value of the filter. Figures 2 and 3 show some kernel examples. By examining the shape of the obtained filter, we see that we can describe it by the following analytical expression:

\[
h(k) = \begin{cases} 
0 & \text{if } k = 0 \text{ or } k > m \\
-ak + (m + 1)a & \text{for } 1 \leq k \leq m + 1 \\
-h(-k) & \text{for } k < 0
\end{cases}
\]  \hspace{1cm} (18)
III. RESULTS AND DISCUSSION

The 1D step edge of figure 4 is superposed to centered gaussian noise in a signal to noise ratio of 2. The transition is clearly detected in the filter output with a more pronounced peak at the transition position for larger filter sizes. The images showed in figure 5 present homogeneous regions with fairly well defined borders. Edges produced by the proposed method are shown in figures 6 and 7 as well as the results obtained by classical differential methods. The purpose here is not to compare the performance of these methods but to show the validity of the proposed formal model in which the other methods simply represent particular situations, Sobel and Prewitt operators for example have the same form as our filter with \( m = 1 \). For larger sizes, the presented filter is visually very similar to discretized Canny operator. Furthermore, edges result generally in images from several physical phenomena (brightness variations, reflectance properties of objects, view orientation and distance to the observed scene). All of these effects are concentrated in the spatial grey level distribution. Hence, it seems more natural for computing the edge information at a given position to take into account the intensity of its neighbours. Of course the neighbours influence decreases as their distance to the considered position increases. This property is implicitly verified by the proposed filters which present symmetrically decreasing magnitude on either sides of the impulse response center. From this point of view, the differential operators can be seen as very narrow versions of the proposed filter. This limited visibility on neighbours causes the poor performance of such operators in the presence of noise. We can notice two directions to explore as perspective. On the one hand, it would be worth to study more deeply the ability of the proposed operator to offer through its parametrical shape and size optimal detection performance depending on the characteristics of specific applications, e.g. image and noise types. On the other hand, other criteria with different detection schemes could be introduced to derive from solutions (6) better edge detectors.

IV. CONCLUSION

In this paper, a generalized linear edge detection method is introduced. Unlike existing operators, the introduced method is entirely developed in discrete form rather than discretized from continuous form. Modeling the edge detection process as a convolution with analytically expressing as constraints the aims to obtain high convolution response at the edges and zero elsewhere leads to a general set of solutions characterized by antisymmetrical impulse responses. Using an optimization criteria to reduce the noise effect, we derived a parametric filter which illustrates the proposed method. The parametrical
aspects of the derived operator are related to the width and the magnitude of the convolution kernel, which can be seen as a mean to control the filter visibility around the edge through the weighted coefficients of the neighbours taken into account to compute the edge information. As perspective, the possibility of adjusting the operator height and width should be exploited to optimize the detection performance for specific image or noise types. It would be useful to study the spectral properties of the proposed operator with an exploration of optimization possibilities directly in the frequency domain.

Finally, the proposed method provides a unified theoretical framework for all existing linear methods that all have the shape of an antisymmetric convolution kernel. On the one hand, this allows a better objective comparison of linear edge detection techniques by defining for example new unified performance criteria. On the other hand it opens the possibility of exploration of new anti-symmetric functions other than classical functions based on Gaussian or exponential forms for edge detection.

REFERENCES


Fast Hierarchical depth map computation from stereo

Vinay Kaushik, Brejesh Lall
Dept. of Electrical Engineering
IIT Delhi, India
India
{eez158117, brejesh}@ee.iitd.ac.in

Abstract—Disparity by Block-Matching stereo is usually used in applications with limited computational power in order to get depth estimates. However, the research on simple stereo methods has been lesser than the energy-based counterparts which promise a better quality depth map with more potential for future improvements. Semi-global-matching (SGM) methods offer good performance and easy implementation but suffer from the problem of very high memory footprint because it’s working on the full disparity space image. On the other hand, Block-matching stereo needs much less memory. In this paper, we introduce a novel multi-scale-hierarchical-block-matching approach using a pyramidal variant of depth and cost functions which drastically improves the results of standard block matching stereo techniques while preserving the low-memory footprint and further reducing the complexity of standard block matching. We tested our new multi-block-matching scheme on the Middlebury stereo benchmark. For the Middlebury benchmark we get results that are only slightly worse than state-of-the-art SGM implementations.

Keywords—multi scale hierarchical block matching, disparity map computation, zero-mean normalized cross-correlation, Gaussian pyramid.

I. INTRODUCTION

Estimating depth information from a stereo-camera is still one of the most versatile solutions for 3-D sensing with a wide application range in robotics[1][2], intelligent vehicles and also space science. The drawback of estimating depth with a stereo camera system, with respect to direct systems like LIDAR or time-of-flight, is the comparably high computational effort which is necessary to extract the depth Information from the stereo images by finding correspondences. On the benefit side stereo cameras provide depth information with a very high spatial resolution which is a prerequisite for obstacle avoidance or path planning. Furthermore, the images provided by the cameras allow for other usage like object recognition or ego-motion estimation. In the literature there are two main categories of algorithms for finding the stereo correspondences: local and global methods. Local methods typically find correspondences by matching patches of one stereo image to the other image. In contrast, global methods typically optimize for an energy function that describes the best transformation of one image into the other[3]. Usually this involves some smoothness or regularization terms in order to tackle NP-completeness for feasible processing. Apart from these two large groups there is one method that is located in between: semi-global-matching (SGM)[4]. On the one hand SGM is also based on an energy-functional, on the other hand it does not employ a fully global optimization but optimizations along one-dimensional paths. This semi-global optimization scheme has a depth accuracy that comes close to global stereo methods but with a much lower computational complexity. Due to this, SGM has become very popular, especially in the domain of intelligent vehicles. One major drawback of SGM is its high memory footprint because it requires the full disparity space image (DSI). This property makes it very challenging to bring SGM to low-energy hardware means like FPGA.

In contrast, local stereo methods based on block-matching are very easy to port to various hardware architecture because they need only a small part of the DSI at a time and the processing is embarrassingly parallel. The downside of local methods is a generally lower accuracy and density of the resulting depth maps. In this paper, we will introduce a novel multi-scale-hierarchical-block-matching (MSIBM) scheme that uses the smallest scale Gaussian stereo pair to compute the reduced depth map and uses the depth at that lower scale to optimally compute the depth at higher scale. This goes on hierarchically to get the final depth.

This scheme leads to a significant improvement in both speed and accuracy over standard block-matching (BM) stereo while still preserving the low-memory and high-parallelization properties. Moreover, it is more robust to image noise. Our experiments with this novel block-matching scheme on the Middlebury stereo benchmark show a major improvement with respect to standard BM stereo.

II. PROBLEMS IN STEREO BLOCK-MATCHING

The basic idea of standard block-matching (BM) stereo is very simple. By correlating image patches (called blocks or filters) between the left and right stereo images, correspondences between the images are found. The position difference of a correspondence is called disparity and is inversely coupled to the distance. There is a large bunch of cost functions [10] used as matching criteria; however, typically sum of absolute difference (SAD), normalized cross-correlation (NCC), rank transform (RT)[5], census transform (CT)[6] zero-mean normalized cross-correlation (ZNCC)[7] are used.

Disparity is defined as the distance between similar points in 2 images. To find those similar points various standard block matching approaches have been defined. The similarity is defined on some mathematical score such as correlation, sum of
absolute difference (SAD), sum of squared difference (SSD), NCC and ZNCC. For a NxN image having maximum disparity D, the disparity of a point is computed by comparing the pixel in right image to a pixel in the left image (on the same epipolar line) D number of times. Considering a block of size MxM, the complexity of the program to compute disparity map is an order of ‘D’. There are several disadvantages of standard block matching techniques. Computing depth maps on higher resolution images not only takes time, but also creates artefacts. Little noise can lead to a very poor disparity map. Many post processing steps (filters) such as WLS filter and anisotropic median filter are applied to improve the generated map. This post processing is not possible in real time applications. Not only it takes time but excessive global filtering also leads to loss of useful depth information.

We propose a novel depth map computation mechanism using a Stereo-Gaussian pyramid that eliminates the requirement of post-processing (filtering) of depth images, takes less time and creates maps for different spatial resolution that are more accurate than standard block matching methods.

We compute depth maps from stereo images that are accurate as well as take minimal time as compared to other algorithms. Using parallelization at its root level helps in performing optimizations in parallel and saves crucial time. We use a Stereo-Gaussian pyramidal variant to compute depth maps of multiple spatial resolution and hierarchically fuse them to compute a map that is more accurate and needs less computations than standard block matching scheme.

III. APPROACH

A standard block matching algorithm uses a cost function for matching image patches along epipolar lines to find the best match, thus defining the disparity of the point. For every pixel, the algorithm searches from 0 to d_{max}, where d_{max} is the maximum disparity level for that stereo pair. The computation is not only time consuming, but the disparity filtering is done as a post processing step to the disparity image. There have been various ways which reduce the disparity search along the epipolar lines, but all such methods consider certain probabilistic assumptions and are themselves an overhead for the system. We define a novel algorithm which drastically reduces the number of comparisons required to compute the disparity for a pixel.

For a stereo image pair, we first define a Depth Search Image (DSI) as follows:

\[
E^k_{(i,j)}(z) = f_{ZNCC}(I_{(i,j)}, R_{(i,j)+z})
\]  
(1)

This defines the matching cost for pixel \((i,j)\) in left image at disparity \(z\), where \(z \in \{N^+; 0 \leq z \leq d_{max}\}\) and \(f_{ZNCC}\) computes a zero-mean normalized cross-correlation between patches L and R. The ZNCC is defined as:

\[
ZNCC(I_1, I_2, u_1, v_1, u_2, v_2, n) := \frac{1}{\sum_{u_1+v_1=n} \sum_{u_2+v_2=n}} \sum_{u_1+v_1=n}(I_1(u_1+v_1,z) - \bar{I}_1)(I_2(u_2+v_2,z) - \bar{I}_2)
\]  
(2)

Where, \(I_1\) and \(I_2\) are stereo images and the patches L and R are of size \(2n + 1\) with \((u_i, v_i)\) as patch centres.

We construct a K-level Gaussian pyramid for both left and right pair of stereo images. The maximum disparity at \(k^{th}\) level is defined as \(d_{max}^k = \frac{d_{max}}{2^k}\), where \(d_{max}\) is the maximum disparity level for the original image.

The matching block size for \(k^{th}\) level decreases by a factor of 2 at every level of Gaussian pyramid. The disparity map computation is further divided into 2 parts. Fig. 1 gives the block diagram of the disparity map computation scheme. Taking the highest level image, we perform standard block matching at that scale to find its disparity map. We further refine the map by performing optimization of pixels with poor matching cost.

A. Selecting Disparity for highest Gaussian level stereo pair

Taking the highest level image, we perform standard block matching at that scale to find its disparity map. We further refine the map by performing optimization of pixels with poor matching cost.

We compute disparity as the best cost match from the Disparity Search Image. Thus,

\[
d^k(i,j) = \arg\max_{z \in \{0,d_{max}^k\}} E^k_{i,j}(z), \text{ and } C^k_{i,j} = \max_{z \in \{0,d_{max}^k\}} \{E^k_{i,j}(z)\}
\]  
(4)

Here, \(d^k_{i,j}\) is the calculated disparity for the pixel \((i,j)\) and \(C^k_{i,j}\) is the matched cost for pixel \((i,j)\).

B. Disparity Refinement

For disparity refinement, we compute an average DSI for every pixel with low matching cost and select the disparity with maximal average matching cost, thus considering disparity as correct if multiple disparity searches around the neighborhood of the point lead to same disparity. Thus,

\[
\tilde{E}^k_{i,j}(z) = \frac{1}{(m,n) \notin n(I_{(i,j)})} E^k_{m,n}(z)
\]  
(5)

Here, \(\tilde{E}^k_{i,j}(z)\) is the average DSI for the pixel \((i,j)\). The final disparity at level \(k\) is given by:

\[
d^k_{i,j} = \begin{cases} 
\hat{d}^k_{i,j}, & \text{if } C^k_{i,j} > \alpha \\
\arg\max_{z \in \{0,d_{max}^k\}} \tilde{E}^k_{i,j}(z) & \text{otherwise}
\end{cases}
\]  
(6)

Similarly, the matched cost at level \(k\) is given by:

\[
C^k_{i,j} = \begin{cases} 
\hat{C}^k_{i,j}, & \text{if } C^k_{i,j} > \alpha \\
\max_{z \in \{0,d_{max}^k\}} \tilde{E}^k_{i,j}(z) & \text{otherwise}
\end{cases}
\]  
(7)
Here, \( d^k_{i,j} \) is the final disparity for the pixel \((i,j)\) and \( C^k_{i,j} \) is the matched cost for pixel \((i,j)\).

### C. Hierarchical Disparity Computation

We then upsample the smaller depth map as well as the cost map using nearest neighbour and bi-cubic interpolation respectively as interpolation method to generate a depth and cost map for lower Gaussian level of stereo pair. Thus, \( d^k_{i,j} = NN(d^k_{i,j}^{+1}) \) and \( C^k_{i,j} = BI(C^k_{i,j}^{+1}) \) are the interpolated depth and cost values at \( k^{th} \) level. Here, NN is the nearest neighbor interpolation method and BI is the Bi-cubic interpolation method.

This disparity image is not an optimal one but acts like a prior for computing the disparity map at this level. Depending on the interpolated cost, we divide the disparity computation in 2 steps.

\[
d^k_{i,j} = \begin{cases} 
\arg\max \limits_{z} E^k_{i,j}(z) & \text{if } C^k_{i,j} > \beta \\
\arg\max \limits_{z \in \{0, d^{\max}_{\text{max}}\}} E^k_{i,j}(z) & \text{otherwise}
\end{cases}
\]

\[
C^k_{i,j} = \begin{cases} 
\max \limits_{z \in \{0, d^{\max}_{\text{max}}\}} E^k_{i,j}(z) & \text{if } C^k_{i,j} > \beta \\
\max \limits_{z \in \{0, d^{\max}_{\text{max}}\}} E^k_{i,j}(z) & \text{otherwise}
\end{cases}
\]

Here, \( d^k_{i,j} \) is the computed disparity at level \( k \) with matching cost \( C^k_{i,j} \).

To compute the optimal depth map at \( k^{th} \) level, instead of performing comparisons till maximum disparity, we use standard block matching only if the pixel’s matching cost is very low (i.e. a bad match in case of ZNCC). For most of the pixels with good matching cost, we perform disparity refinements i.e. we begin matching near the disparity obtained from the interpolated image. The number of comparisons for refinement per pixel depends on the level of the image, and in our case since we halved the image to construct Gaussian pyramid, it takes just 3 comparisons per pixel to find the optimal disparity of the larger stereo pair.

After this step, selective median filtering is applied to every bad pixel to further optimize the disparity map.

### D. Selective Median Filtering

\[
\begin{array}{cccccccc}
0.63 & 0.72 & 0.28 & 0.63 & 0.55 & \text{a)} & \text{b)} \\
0.72 & 0.21 & 0.53 & 0.31 & 0.86 & \text{a)} & \text{b)} \\
0.85 & 0.93 & 0.26 & 0.32 & 0.66 & \text{a)} & \text{b)} \\
0.95 & 0.78 & 0.15 & 0.83 & 0.63 & \text{a)} & \text{b)} \\
0.78 & 0.83 & 0.21 & 0.32 & 0.82 & \text{a)} & \text{b)} \\
\end{array}
\]

Fig. 2. a) Cost match for a 5x5 patch b) Disparity Map for a 5x5 patch. Median filtering is performed on selected green colour disparity values.

To further refine the disparity image, we apply a variant of median filtering on the pixels with low matching cost. For every such pixel, we apply a 5x5 patch around it and discard the pixels with bad matching cost from that patch. As shown in the above

\[
d^k_{i,j} = \begin{cases} 
d^k_{i,j}^{\text{median}} & \text{if } C^k_{i,j} > \alpha \\
d^k_{i,j}^{\text{th}} & \text{otherwise}
\end{cases}
\]

Here \( d^k_{i,j} \) is the optimal disparity map of the \( k^{th} \) level. We follow this approach hierarchically to compute the disparity map for the original image.

The above approach incorporates the best of information from the lower scale level of Gaussian pyramid and block matching at current scale to compute a disparity map which is more robust as well as faster than the standard block matching mechanisms.

### IV. Results

Middlebury V3 dataset [8] is used for experimentation. It consists of 15 test images. It introduces more challenging scenarios such as high-resolution stereo pair images, varying exposure and lighting settings, imperfect rectification, etc. We compare our technique with several other state of the art BM and SGM methods. We also provide the runtime evaluations on the methods which run on CPU based platforms (Table I). The runtime is computed as an average of the time taken to compute depth for the entire dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>Environment</th>
<th>Average Runtime (in min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BM stereo</td>
<td>MATLAB</td>
<td>50</td>
</tr>
<tr>
<td>BM stereo (multithreaded)</td>
<td>MATLAB</td>
<td>10</td>
</tr>
<tr>
<td>Displets v2 [9]</td>
<td>MATLAB + C++</td>
<td>4</td>
</tr>
<tr>
<td>PSPO [10]</td>
<td>MATLAB + C++</td>
<td>5</td>
</tr>
<tr>
<td>ISF [12]</td>
<td>C++</td>
<td>10</td>
</tr>
<tr>
<td>Ours</td>
<td>MATLAB</td>
<td>2</td>
</tr>
</tbody>
</table>

Our method performs reasonably well in terms of both speed as well as accuracy. The method is comparable to the best SGM methods and outperforms all BM methods. We evaluate our method on basis of average error computed by the Middlebury Eval 3.0 toolkit (Table II). We use a block size of 11x11 for the lowest level disparity image. The threesholding parameters include \( \alpha \) and \( \beta \). If \( \alpha, \beta \) tends to 0, although the method takes less time but there is no disparity refinement. If \( \alpha, \beta \) tends to 1, it recomputes the disparity of every pixel on the basis of its neighbors which increases computational complexity drastically and softens disparity boundaries. Thus we select \( \alpha \) and \( \beta \) as 0.9 for our experiments as it gives the optimal result.

Since it is at the highest level, it needs less number of comparisons. And since we use a Gaussian blur over original image, the matching is robust to some amount of noise in the
image. Fig. 3 shows the disparity maps of few example images of the Middlebury V3 dataset.

![Disparity Maps](image)

![Fig. 3. 1st and 2nd Row: Original Image, 3rd and 4th Row: Disparity Map](image)

<table>
<thead>
<tr>
<th>TABLE II. AVERAGE DISPARITY ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adirondack</td>
</tr>
<tr>
<td>MotorcycleE</td>
</tr>
<tr>
<td>Jadeplant</td>
</tr>
<tr>
<td>Motorcycle</td>
</tr>
</tbody>
</table>

V. CONCLUSION

We have proposed multi scale hierarchical block matching algorithm using pyramidal variant of depth and cost maps. We exhaustively evaluate it against disparity computation techniques which require low processing power and demonstrate that the proposed approach achieves high accuracy while having low computational cost. This can be attributed to the selective cost optimizations using lower scale depth map and cost map as a prior limiting the disparity search space.

REFERENCES


Fast Context-Annotated Classification of Different Types of Web Service Descriptions

Serguei A. Mokhov, Joey Paquet, Arash Khodadadi
Department of Computer Science and Software Engineering
Concordia University
Montreal, Canada
{mokhov, paquet}@cse.concordia.ca

Abstract—In the recent rapid growth of web services, IoT, and cloud computing, many web services and APIs appeared on the web. With the failure of global UDDI registries, different service repositories started to appear, trying to list and categorize various types of web services for client applications’ discover and use.

In order to increase the effectiveness and speed up the task of finding compatible Web Services in the brokerage when performing service composition or suggesting Web Services to the requests, high-level functionality of the service needs to be determined. Due to the lack of structured support for specifying such functionality, classification of services into a set of abstract categories is necessary.

We employ a wide range of Machine Learning and Signal Processing algorithms and techniques in order to find the highest precision achievable in the scope of this article for the fast classification of three type of service descriptions: WSDL, REST, and WADL. In addition, we complement our approach by showing the importance and effect of contextual information on the classification of the service descriptions and show that it improves the accuracy in 5 different categories of services.

I. INTRODUCTION

When two systems (e.g. Web Services) want to interact, a compatibility assessment which requires in-depth analysis considering the interface and conversational protocol of them needs to take place. As the authors in [1] argue, one way to speed up the assessment above is to apply Machine Learning methods to automatically classify high-level functionality of a system’s interface description, i.e. the highest level of abstraction of what the system does. This will result in restricting the scope of compatibility checks and consequently providing an overall performance gain when looking for matches between systems. In addition to increasing performance of compatibility assessment, the authors in [2] argue that classifying Web Services into different sets based on the tags (clustering) facilitates the task of Service Discovery. Moreover, the result of Service Classification can be very useful to the end-users when selecting services.

Service Classification or Categorization is the task of associating Web Service descriptions to a predefined set of categories which can considerably speed up and increase the effectiveness of the task of finding compatible Web Services in Brokerage or suggesting Web Services to the requests [1]. Categories or classes specify the purpose of the service and what it does at a high level. However, there is no structured support for specifying the abstract category to which the service belongs [1]. As a result, this classification task needs to be done manually or automatically.

There are two main approaches towards Service Classification: manual classification and automatic classification. According to [3], the former methods are very expensive, both in time, effort and consequently financially. The latter methods however are quite inaccurate and do not in general provide quality annotations but cheaper than the former. Although the authors in [3] try to decrease the cost of manual classification by applying crowd sourcing techniques, it is still more expensive than the automatic methods. Due to the fact that cost plays an important role and because of the resources available to us, although we are aware that the annotations in many cases are inaccurate and the automatic classification may not be as accurate as manual methods, we use automatic classification approach as our primary methodology. We build on the considerable amount of research that has been carried out on the topic of automatic classification of a text document which has many practical applications [4].

The task of automatic classification of documents is usually tackled by applying Machine Learning techniques. These techniques use classifiers that have been automatically induced by estimation on a collection of documents which is called the training set [5]. Machine Learning methods can be divided into two broad categories: (1) supervised learning, where each document in the training set is already associated with a category by a human supervisor and (2) unsupervised learning, where documents are not associated with a category prior to the learning process and the Machine Learning method must find a meaningful division into categories. In this article we focus on the former method, which has generally been much more successful in most studies as pointed out in [1].

Context, in the Web Services environment is any information about the service consumer, service provider, and communication protocols. Hence, content of the service descriptions and any information related to them is considered as a context for the service.

In [6] we discussed harvesting and storing Web Service descriptions and their contextual information from different sources. Eventually, we found 72,454 unique service description URLs including 39,288 WSDL URLs, 1,830 WADL URLs, and 31,336 HTML page URLs describing RESTful services. From these URLs we stored 48,161 actual service
description files including 16,096 WSDL descriptions, 450 WADL descriptions, and 31,615 HTML files describing RESTful services. We constructed a repository of Web Service descriptions and their contextual information. In this article we try to find the highest accuracy achievable in the scope of this article by employing a wide range of Machine Learning and Signal Processing algorithms and techniques and putting the context into practice.

We use the open-source MARF framework and its MARFCAT application because they are designed as an input media type-independent investigation platform to execute a considerable number of experiments in a short amount of time and to assist selecting the best combinations of different available algorithms. In this application, we use signal processing techniques which use character-level (bi-grams) processing rather than syntax and semantic levels and we treat the descriptions as a signal which will be discussed in details in Section II-A. In Section III-C and Section III-C we discuss different algorithms and options available in MARFCAT.

Using MARFCAT as our investigation platform, we systematically test and select the best (a tradeoff between accuracy, recall, and speed) combination(s) of algorithm implementations (configuration) available to us for each type of service descriptions (Section III-B) and then use only those for the final classification of all service descriptions based on the classes defined in Section III-B. We will discuss our methodology in Section III-D.

II. BACKGROUND

A. MARF and MARFCAT

Modular Audio Recognition Framework (MARF), is an open-source collection of pattern recognition APIs and their implementation for unsupervised and supervised Machine Learning and classification. MARF was designed to act as a testbed to verify and test common and novel algorithms found in literature for sample loading, pre-processing, feature extraction, and training and classification, which constitute a typical pattern recognition pipeline [7]. Over the years, MARF accumulated a fair number of implementations for each of the pipeline stages which allows us to execute reasonably comprehensive comparative studies of algorithm combinations for the Service Classification purpose.

The pattern recognition process starts by loading a sample (e.g., an audio recording, a text, or image file), removing noisy and/or silent data and other unwanted elements (pre-processing), then extracting the most prominent features from it (feature extraction), and finally either training the system such that the system learns a new set of features of a given subject or classifying what the subject is. The outcome of the training process is either a collection of some form of feature vectors or their mean or median clusters, which are stored for every subject learned. The outcome of classification is the class that the system believes the subject belongs to and a score attached to it [7].

The loading stage in MARF starts with the interpretation of the files being scanned in terms of bytes forming amplitude values in a signal using either unigram, bi-gram, or tri-gram approach. Then, the pre-processing allows to be none-at-all (raw, or the fastest), normalization, traditional frequency domain filters, wavelet-based filters, etc. Feature extraction involves reducing an arbitrary length signal to a fixed length feature vector of what is thought to be the most relevant features in the signal, e.g., spectral features in FFT, LPC, min-max amplitudes, etc. The classification stage is then separated either to train by learning the incoming feature vectors (usually k-means clusters, median clusters, or plain feature vector collection, combined with neural network training) or testing them against the previously learned models [8].

MARFCAT is a MARF-based Code Analysis Tool, which was first exhibited at the Static Analysis Tool Exposition (SATE) workshop in 2010 [9], [10]. MARFCAT, as any MARF application, can be used for a wide array of recognition tasks, not only applicable to audio, but rather to general pattern recognition for various applications, such as in digital forensic analysis, writer identification, natural language processing (NLP) [11], among others. In particular, MARFCAT was used to analyze source and byte code to fingerprint, detect, and classify vulnerabilities and weaknesses in [12]–[14] and do the same for network packet traces [15].

B. Motivation to Use MARFCAT

The following are primary motivations justifying the use of MARFCAT in this work:

1) MARFCAT was successfully used in related source code and text analysis tasks, for specific vulnerabilities and defects as well as more general weakness categories as referenced earlier. At its introduction in 2010, it was arguably the first time such an approach was applied to text analysis and was deemed novel in these types of tasks. The most significant advantage of it was the processing speed compared to other code analysis tools. By extension this applied it to Web Services descriptions in various formats.

2) MARFCAT supports both signal processing and NLP pipelines. However, the signal pipeline was found by an order of magnitude faster than most parsing and NLP approaches [13], [14]. Thus, spectral analysis was proven beneficial in code analysis, source, and binary as well as network packet traces, and natural language processing. It is analogous to analyze the signal from a distant star, breaking it down into spectrum of emitted light in order to classify the chemical composition in terms of elements present in the star, i.e., to fingerprint them. MARFCAT similarly fingerprints a spectrum of text or any other media into bins related to different categories it was shown to learn from.

3) MARFCAT is very easy to quickly setup and do preliminary testing in search for good algorithms. It can also be used as a front-end for semantic- and ontology-based parsing classifiers to prioritize their work [12].
III. METHODOLOGY

A. Architecture

The architecture of this entire work is illustrated in Figure 1. This article focuses predominantly on service classification block illustrated in this figure. Specifically, Figure 1 illustrates the whole architecture and how the two steps of service collecting and Service Classification are connected.

![Fig. 1: Service collection/classification architecture](image)

Figure 1 depicts the Service Classification concrete architecture and components. As discussed in Section III-D, initially we test each sample type (Section III-B) independently in order to find the best configuration of MARFCAT for that sample type. At the end, after finding the best configuration for each sample type, we create the testing sets for each type from all the service descriptions in the repository which are not classified yet. Then we perform the final classification for each type based on the best configurations (algorithm combination + clustering method) found from the previous step and store the resulting classes associated with each snapshot in the snapshots info part of the service descriptions repository.

B. Data Sets, Classes, and Training Sets

Our service repository consists of two main repositories: (1) the Service Descriptions Repository stores service description URLs, providers, contextual information, and snapshots information and references. This is the main repository structure and serves as a basis for the generation of our data sets (snapshots) for the purpose of the current study. This repository is implemented as a SQL database. (2) The Snapshots Repository, is a file-based repository which stores snapshots of service descriptions, context files, and snapshots of service descriptions combined with their context files. Each snapshot is stored in a folder named with its service provider URL and linked to Snapshots information in the service descriptions repository. In [6], we have designed a web crawler to search the web for web service descriptions in order to create an web service repository able to store as many web service descriptions that we could find. In doing so, we have identified the three main service descriptions commonly used: WSDL, WADL and REST. In order to convey the current study, we store each service description and their context separately. Due to the nature of each type’s description’s characteristics and features, and to compare and analyze the results separately, we survey each service description type’s classification process independently. For WSDL and WADL files, we use the descriptions directly, which we found using our web crawler. On the other hand, for the service descriptions regarding REST services, which are often described using HTML files, we take an additional step before feeding them to MARFCAT because of the nature of these files which contain too much noise, e.g., using scripts to strip the code. In this step we remove all the HTML tags and unnecessary sections and only keep the raw text inside and store it in a separate text file and consider it as a new type of sample. Likewise, this step can be applied to WSDL and WADL files to remove all their tags. However, they do not contain much noise and MARFCAT will take care of noise removal in the pre-processing step. Therefore, this task were postponed to the future work because it will multiply the number of the tests to be applied. As a result, our repository will contain four general types of samples:

- WSDL files (.wsdl)
- WADL files (.wadl)
- HTML files (.html)
- Tags-Filtered description files (.txt)

Another dimension which is added to each of these types is their contextual information. In order to show the effect of context on the classification and to find the best configuration, we define three type of samples with respect to the contextual information for each of general types defined above:

- Plain files (description files without any context added to them)
- Combined with context files (plain descriptions + context)
- Only context files (files containing only the contextual information of service descriptions)

Each of these data sets are then to be loaded into MARF and processed following the data flow shown in Figure 2. We use 5 classes for the classification with respect to previous research [16], [17], [18], the most popular categories in ProgrammableWeb\(^1\) and more importantly the nature of Web Service descriptions in the repository and their intersections:

- Weather
- Social
- Tourism
- Entertainment
- Financial

As mentioned in Section II-A, in order to classify the service descriptions which are stored in the repository, we need training sets for each of these classes. These sets need to be chosen with minimal intersections to be definite candidates for the class. In addition, for the testing purposes to find the best combination of configurations which will be discussed in Section III-D, we need testing sets for each of the classes. Therefore, we manually classify 500 instances (100 per each class) for WSDL and REST files. However, for WADL files because of the inadequacy of the files in the repository as

\(^{1}\)http://www.programmableweb.com/
discussed in [6, Chapter 3], only for weather, social and financial 10 definite matches can be found. For tourism 3 candidates and for entertainment only 2 candidates are chosen.

C. MARFCAT Configuration

We use MARF and its application MARFCAT to find the best algorithm combinations for each sample type which were mentioned in Section III-B. We use the fast script of MARFCAT which performs the algorithms illustrated in Figure 2 in each step of its pipeline: 1 Loader, 4 techniques in the Preparation stage, 9 algorithms in the Pre-processing stage, 4 algorithms in the Feature Extraction stage, and 6 Distance Classifier algorithms. The combination of these algorithms will result in 864 permutations which we test in each of our cases as discussed in Section III-D. The following is a brief description of some of the algorithms and options that we have used in this research.

Algorithms Used by MARFCAT

The specific algorithms come from the classical literature and other sources and are detailed in [7], [15]. The below is a summary of some algorithms corresponding to [7] with a brief description:

Fast Fourier Transform (FFT): A version of the Discrete Fourier Transform used in FFT-based filtering as well as feature extraction [19]. It is also used in FFT-based filters (both forward and inverse FFT to reconstruct the signal after filtering). Uses 512 frequencies by default (empirically determined by the MARF project).

Linear Predictive Coding (LPC): Used in feature extraction, which evaluates windowed sections of signals and determines a set of coefficients approximating the amplitude vs. frequency function. Uses 20 poles by default.

Distance Classifiers: Various distance classifiers (Chebyshev, Euclidean, and Minkowski [20], Mahalanobis, Diff (internally developed within MARF, roughly similar in behavior to the UNIX/Linux diff utility [21]), and Hamming).

Cosine Similarity Measure: Cosine similarity measure was thoroughly discussed in [22] and often produces the best or near best accuracy in MARF in many configurations.

MARFCAT Options

All mentioned algorithms are selected as options in a scripted manner exhaustively at the first stage in order to select the candidate best options for subsequent classification. Not all combinations are necessarily optimal or have effect together (e.g., noise removal uses low pass filter at pre-processing and then if low pass filter is applied, it doubles the work, without additional filtering effect), but they are easy to automate and there is no dependency assumptions between algorithms at different stages keeping them decoupled and re-usable. We survey some of these options to find the best configuration for each type of service description which will be discussed in Section III-D.

In order to be able to classify samples into different classes, an automatic classifier determines the salient properties of the samples and puts them into different feature vectors. This process is called feature extraction [1]. In MARF there are different ways of storing and matching feature vectors that MARFCAT takes advantages of from a specific class. These are referred to as clustering options in MARFCAT and can be customized:

- k-means clusters (mean option)
- median clusters (median option)
- plain feature vector collection (no clustering option)

D. Testing Methodology

As discussed before in Section II-A, we use MARF and its application MARFCAT to find the best algorithm combinations for each service description types which were mentioned in Section III-B. In order to perform this task, MARFCAT defines two processes which were discussed in Section II-A: (1) The learning process in which the feature vectors are extracted and the system learns the classes from the training set and (2) the testing or classification process in which the testing set is classified based on the previously learned models.

In order to evaluate the performance of the classifiers, we compute different evaluation measures. These measures are usually presented as percentages. Consider for a given class \( C \), \( n_t \) samples are expected, i.e., are labeled with \( C \). The classification system classifies (labels) \( n_s \) samples as \( C \) including \( n_c \) correct samples (true positives) and \( n_n \) incorrect samples (false positives).

**Total Accuracy** \( a \): Total accuracy is defined as the fraction of the samples which were classified in the same class as expected in total: **Precision** \( p \): Precision is defined for each class as the fraction of classified items which are relevant, i.e., expected in that class: We also compute the macro precision which is the average of precision over all classes. **Recall** \( r \): Recall (also called sensitivity) is defined for each class as the fraction of relevant items which are classified: We also compute the macro recall which is the average of recall over all classes. **F-Measure** \( f \): F-Measure is defined for each class as the harmonic mean of precision and recall: We also compute the macro \( F \)-Measure which is the average of \( F \)-measure over all classes. **Classification Time**: Classification time is the total execution time of the classification process over the data set.

\[
\alpha = \frac{\sum_{i=1}^{C} n_c}{\sum n_t}; \quad p = \frac{n_c}{n_s}; \quad r = \frac{n_c}{n_t}; \quad f = 2 \frac{p \cdot r}{p + r}
\]
In our methodology, initially we survey each description type (Section III-B) independently in order to find the best algorithm combinations considering the above-mentioned measurements. In addition, there are also other options which MARFCAT provides and were referred in Section III-C as clustering and frequency options. We test all three clustering options in all cases to find the best algorithm combinations. On the other hand, because the frequency change did not have any effect on the precision (see Section IV) when tested in the best case, we ignored it for the other cases. Therefore, we find the best configuration of MARFCAT for each sample type which consists of an algorithm combination using a specific clustering method.

As argued in Section III-B, we chose 5 classes and we manually classified 500 instances (100 per class) for each service description type. As discussed in Section III-B, contextual information adds another dimension to the samples and adds 2 more sample types (plain + context, and only context) for each of the service description types. In order to completely survey the possible cases and find the best configuration(s), we train and test on all type of samples exhaustively.

Figure 3 illustrates our methodology which forms 72 different cases based on the sample types and clustering options. Each case consists of one row from each of the blocks; one description type, training and testing on which type of file considering contextual information. We tested 864 algorithm permutations in the Signal Processing pipeline for each case and illustrated in Figure 2.

Fig. 3: Testing Methodology

In this exhaustive test process we choose randomly a smaller set of the manually classified instances in order to keep the tests simple and applicable in a shorter amount of time (62,208 runs). After finding the best cases, we increase the sets and use all 500 instances for each of service description types. Using these sets we perform another exhaustive search on the algorithm combinations in order to find the best algorithm combination.

Using the best case and the best algorithm combination, we perform a 10-fold cross-validation in order to give an insight on how the model will generalize to an independent dataset and to reduce variability. In this procedure, we split the data randomly into 10 pieces and run the classification 10 times using one of the pieces (10%) as the testing set and the rest (90%) as the training set in a way that each sample is present once and only once in the testing set among the runs and then, average all the results.

In order to show the effect of the use of context on the classification, we study the best configuration without any contextual information added, and the best one through all other cases with context. As a result, we perform the 10-fold cross-validation for each of these cases and compare the results in order to illustrate the effect of context. We then compare the evaluation measures (total accuracy, macro recall and precision, macro F-Measure, and the classification time) in order to show the effect of context. In addition, we compare the performance of our classification for WSDL files with the literature in order to give an insight on how close our classification stands. However, we could not find any related work for classification of WADL or REST descriptions to compare.

At the end, after finding the best configuration for each sample type, we create the sets from all the service descriptions in the repository which are not classified yet and perform the classification for them and store the results in the repository.

IV. RESULTS AND EVALUATION

As discussed in Section III-D, in order to classify the service descriptions, we first find the best configuration of MARFCAT (best algorithm combination + clustering option). In addition, we add the contextual information to the classification, our hypothesis being that adding contextual information to the files will improve the performance of the classification.

Using this approach, as illustrated in Section III-D, based on the sample types and considering the contextual information and the clustering options, we survey 72 different cases. As illustrated in Figure 2, we exhaustively test 864 algorithm permutations for each case. As discussed in Section III-B, we classify and survey each service description type individually. For each service type, we survey 18 different cases in order to find the highest accuracy achievable without considering any contextual information (training on Plain files and testing on Plain files) and with the contextual information in effect (e.g., training on Plain + Context files and testing on Plain + Context files).

A. WSDL Classification Results

Table I depicts the highest accuracy achievable by exhaustively testing 864 algorithm permutations for each case of WSDL files using different clustering options which were mentioned in Section III-C.

The highest accuracy without considering context (Train on Plain-Test on Plain column) is achieved by using the No Clustering option: 64 percent. The highest accuracy with considering context (other columns) is achieved by using the No Clustering option and using Plain + Context files in the training and Plain files in the testing: 76 percent.

As discussed in Section III-D, after finding the best case which is using the No Clustering option and using Plain + Context files in the training and Plain files in the testing, we increase the sets and perform another exhaustive search in order to find the best algorithm combination.

As the results depicted, the best result for the classification of WSDL files is achieved by training on Plain + Context files and testing on Plain files using the following configuration:

- No Clustering option (discussed in Section III-C)
• –silence for preparation, –endp for pre-processing, –lpc for feature extraction, –eucl for classification

As mentioned in Section III-C, –silence option removes near-zero gaps from the data. There are usually many white-spaces and empty parts in the WSDL files that are normalized close to zero or silence gaps appear due to low-pass filtering.

As a result this preparation technique helped to improve the overall classification combination. Additionally, as mentioned in Section III-C from a theoretical point of view, LPC works well with compressed form of signal, such as with local minimums and maximums with silence removed. In addition, Euclidean distance (which is sensitive to high-dimensional vectors) works better with less varied 20-sized vectors, such as produced by LPC combined with endpointing.

As discussed in Section III-D, we perform 10-fold cross-validation based on this configuration in order to give an insight on how the model will generalize to an independent dataset and to reduce variability. Table II depicts the cross-validated results including the evaluation measures which were defined in Section III-D.

We compare the performance of our classification for WSDL files with the literature in order to give an insight on how close our classification process is to the literature. However, most of the research has been carried out for semantically-defined files (using Ontology Language (OWL-S) [23] and Web Service Modeling Ontology (WSMO) [24]) which are not available at a large scale and are a small subset of available service description files. The authors in [1] used different techniques for feature extraction such as Bag of Words variances and different algorithms for machine learning such as Support Vector Machines (SVM) variances and compared them in order to find the best classification performance for WSDL files. Finally, the best combination was the result of employing Support Vector Machines and use a feature extractor that is tailored to the task of WSDL classification by using its structure, in particular the identifiers. Table II (first row) depicts the results of their work including the same evaluation measures except the classification time, which is not presented as part of their results.

Although the data sets from which the tests are performed are different, we can conclude that our classification accuracy is very close to the best result from the literature without any customization on the preprocessing, feature extraction, and classification based on the WSDL files. MARFCAT offers a good tradeoff between precision and speed and helps us to validate the hypothesis on the positive effect of context on classification results.

In order to show the effect of contextual information, as discussed in Section III-D, we perform another 10-fold cross-validation on the same configuration without considering any contextual information and training on the Plain files and testing on the Plain files. Table II (second row) depicts the cross-validated results including the evaluation measures without considering any contextual information.

The results demonstrate that contextual information is improving the performance of the classification even though it is increasing the classification time due to increase in the file sizes because of the added context information.

Finally, we use the best configuration (using contextual information) in order to perform the final classification of all the WSDL files which class is unknown. Table III depicts the number of instances which was classified inside of each class. These results are based on the performance of the current classification tool. Currently, the results cannot be verified because the actual classes are not known. In order to validate, all the instances need be classified by human contribution by using approaches such as crowd-sourcing.

B. WADL Classification Results

WADL descriptions are not popular through service providers and REST services are not widely described using WADL in the Web. As a result, we could not find as many instances for them as we could for WSDL and REST descriptions. The scale of the samples are not as much as the WSDL and REST samples and we discard the 10-fold cross-validation for these files. However, for the sake of completeness we perform a 2-fold cross-validation (swapping training and testing set and averaging the results) and we compare the results with the context and without considering any contextual information.

Table IV depicts the highest accuracy achievable by testing algorithm permutations for each case of WADL files using different clustering options.

The highest accuracy without considering context (Train on Plain-Test on Plain) is achieved by using the No Clustering option: 64.71 percent. The highest accuracy with considering context (other columns) is achieved by using the No Clustering option and using Plain + Context files in both training and testing: 76.47 percent.

Table V depicts the result of two-fold cross-validation of the best cases.

C. REST Files Classification Results

As discussed in Section III-B, because HTML files contain too much noise, e.g., script code, we define a new type of sample for REST HTML files and remove all the tags and unnecessary sections and only keep the raw text inside and store it in a separate text file. We survey both sample types in order to find the best case for classification of REST service

<table>
<thead>
<tr>
<th></th>
<th>TrainPlainCtx</th>
<th>TrainPlain</th>
<th>TrainPlainCtx</th>
<th>TrainPlain</th>
<th>TrainCtx</th>
<th>TrainCtx</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>60</td>
<td>52</td>
<td>48</td>
<td>52</td>
<td>36</td>
<td>28</td>
</tr>
<tr>
<td>No Clustering</td>
<td>68</td>
<td>64</td>
<td>76</td>
<td>64</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>Median</td>
<td>56</td>
<td>56</td>
<td>52</td>
<td>52</td>
<td>28</td>
<td>32</td>
</tr>
</tbody>
</table>
As discussed in Section III-D and similar to WSDL files, we perform 10-fold cross-validation based on this configuration in order to give an insight on how the model will generalize to an independent dataset and to reduce variability. Table VII depicts the cross-validated results including the evaluation measures which were defined in Section III-D.

The performance is lower in comparison with WSDL files due to the lack of common structure and high variability of these pages that describe RESTful services in different structures and terminologies and not in a structured format specific to describing Web Services like WSDL files. However, the performance is still significantly higher than the random baseline would have been. Unlike WSDL files, we could not find any related work for classification of REST descriptions in the literature in order to compare with. As far as we know, this work is the initial step towards the classification of REST descriptions. We discuss in Section IV-D how the performance of classification of REST descriptions could be improved.

In order to show the effect of contextual information, as discussed in Section III-D, we perform another 10-fold cross-validation on the same configuration without considering any contextual information and training on the Plain files and testing on the Plain files.

Table VII illustrates the effect of adding contextual information to the the REST tags-filtered files on total accuracy, macro precision, macro recall, macro F-Measure, and classification time.

The results depict that contextual information improves the performance of the classification even though it is increasing the classification time due to increase in the file sizes because of the context which is added to them. However, for REST tags-filtered files it is not improving the accuracy as much as for WSDL files due to the nature of these descriptions, which are not defined in a structured format specific to describing Web Services. In other words, because they contain phrases which are more similar to the contextual information phrases, context is not adding much discriminant features to the REST description files.

As a result, we use this configuration to perform the final classification of all REST files. We use the same training result which is the result of training on Plain + Context REST tags-filtered files and the Plain REST tags-filtered files for the testing set to be classified.

Finally, we use the best configuration (using contextual information) in order to perform the final classification of all REST description files which class is unknown. Table VIII depicts the number of instances, which were classified inside of each class. Similar to the WSDL files as discussed in Section IV-A, these results are based on the performance of the current classification tool. Currently, the full complete
classification results cannot be verified because the actual classes are not known. In order to validate our complete data set, all the instances need to be classified by human contribution by using approaches such as crowd-sourcing.

D. Evaluation

WADL descriptions are not popular on the Web and REST services are not widely described using WADL descriptions. As a result, as discussed in Section III-B, because of the low number of WADL files in the repository, we were not able to find the same number of samples for the classes in comparison with other sample types. The scale of the samples are not as much as the WSDL and REST samples and we discard the 10-fold cross-validation for these files. However, for the sake of completeness we performed a 2-fold cross-validation (swapping training and testing set and averaging the results) and we compared the results with the context and without considering any contextual information. Despite having to rely on limited data sets, our results show that the use contextual information does increase the effectiveness of WADL classification.

As the results depict, the accuracy is generally lower for REST HTML files in comparison with WSDL files because they have more noise, e.g., JavaScript and markup code, and natural language segments. However, after filtering the tags and cleaning-up these files and storing the raw text inside in a separate text file, the accuracy increased in general and it helped the classification accuracy. Although, the performance for them is still generally lower in contrast with WSDL files due to the nature of these descriptions which are not defined in a structure format specific to describing Web Services and as a result, have high variability due to using different structures and terminologies embedded in the HTML documents. However, the resulting performance is still higher than the random baseline after 10-fold cross-validation. Unlike WSDL files we could not find any related work for classification of REST descriptions in the literature in order to compare with. As far as we know, this work is the initial step towards the classification of REST descriptions. The performance of the classification for these files can be improved using an approach to extract the most prominent features specific to these files before performing the classification. One way to achieve such goal is to extract all resource URIs using a regular expression extraction process. However, because the URIs are also defined in different structures, formats, and shortcuts and have variability in the files, it requires significantly more experimentation to be done at the semantic-level processing.

No Clustering option, which was mentioned in Section III-C, is found as the best clustering option in the best configurations of MARFCAT for all of the three types of service descriptions. This option disables clustering the training and testing sets’ individual class’s feature vectors in MARF, i.e., it uses all of the feature vectors of the instances which we passed for a specific class in the training set and calculates their distance to all of the feature vectors of the instances which we passed for a specific class in the testing set instead of using only one feature vector (mean or median). As a result, the space and the time complexity increases. However, because our priority in finding the best configuration is the highest accuracy, we chose this option.

As the results depict, the algorithm combinations which are found as the best combinations in the best configurations of MARFCAT vary throughout the different types of service descriptions. The reason is that this experiment is data-driven and the results is based on the input data. As a result, because the structure and nature of each of these types is different and also due to the manual choosing of training and testing sets for one type regardless of the other types, the aforementioned algorithm combinations vary.

The effect of adding contextual information to the WSDL files is illustrated in Table II. The results depicts that contextual information is improving the performance of the classification for both cases even though it is increasing the classification time due to increase in the file sizes because of the context which is added to them. The context has less effect on the precision of REST tags-filtered files in comparison with WSDL files due to the nature of these descriptions which are not defined in a structure format specific to describing Web Services.

V. CONCLUDING SUMMARY

In this article, we discussed our methodology of combining Machine Learning and Signal Processing techniques and employing contextual information in order to automatically classify Web Service descriptions. We defined 72 different cases based on the sample types, clustering options, and the contextual information which we survey 864 combinations of algorithms and techniques in each. In Section IV we measured and illustrated the results of Service Classification including the resulting accuracies of 72 different cases based on the clustering options and adding contextual information, cross-validated results for the best cases, the effect of adding contextual information to the samples on the classification. In addition, for WSDL files we compared the same evaluation
measures with the literature in order to give an insight on how close our classification process is to the literature. Unlike WSDL files we could not find any related work for classification of REST descriptions in the literature in order to compare. As far as we know, this work is the initial step towards the classification of REST descriptions. We found and presented the best configuration of MARFCAT (best algorithm combination + clustering option) which will result in the highest precision for each type of service description. In addition, we added the contextual information to the classification and showed that it improves the performance of the classification and validated our hypothesis.

REFERENCES


---

**TABLE VI:** Classification accuracy across data sets for REST tags-filtered files

<table>
<thead>
<tr>
<th>Category Name</th>
<th>Number of Instances</th>
<th>TrainPlainCtx</th>
<th>TestPlainCtx</th>
<th>TrainPlainCtx</th>
<th>TestPlainCtx</th>
<th>TrainPlainCtx</th>
<th>TestPlainCtx</th>
<th>TrainPlainCtx</th>
<th>TestPlainCtx</th>
<th>TrainPlainCtx</th>
<th>TestPlainCtx</th>
<th>TrainPlainCtx</th>
<th>TestPlainCtx</th>
<th>TrainPlainCtx</th>
<th>TestPlainCtx</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weather</td>
<td>6414</td>
<td>48</td>
<td>40</td>
<td>36</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Social</td>
<td>6123</td>
<td>48</td>
<td>32</td>
<td>52</td>
<td>44</td>
<td>44</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tourism</td>
<td>5560</td>
<td>40</td>
<td>44</td>
<td>48</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Financial</td>
<td>6097</td>
<td>29.67</td>
<td>28.00</td>
<td>28.46</td>
<td>29.67</td>
<td>28.00</td>
<td>29.60</td>
<td>29.60</td>
<td>28.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Entertainment</td>
<td>6122</td>
<td>28.00</td>
<td>28.46</td>
<td>29.60</td>
<td>27.67</td>
<td>29.60</td>
<td>29.12</td>
<td>29.60</td>
<td>29.60</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE VII:** Effect of context on cross-validated REST classification

<table>
<thead>
<tr>
<th>Category Name</th>
<th>Number of Instances</th>
<th>without context</th>
<th>with context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weather</td>
<td>6414</td>
<td>28.00</td>
<td>29.60</td>
</tr>
<tr>
<td>Social</td>
<td>6123</td>
<td>28.46</td>
<td>29.67</td>
</tr>
<tr>
<td>Tourism</td>
<td>5560</td>
<td>28.00</td>
<td>29.60</td>
</tr>
<tr>
<td>Financial</td>
<td>6097</td>
<td>27.67</td>
<td>29.12</td>
</tr>
<tr>
<td>Entertainment</td>
<td>6122</td>
<td>29.60</td>
<td>29.60</td>
</tr>
</tbody>
</table>

**TABLE VIII:** Final REST classification results

<table>
<thead>
<tr>
<th>Category Name</th>
<th>Number of Instances</th>
<th>No Clustering</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weather</td>
<td>6414</td>
<td>48</td>
<td>40</td>
</tr>
<tr>
<td>Social</td>
<td>6123</td>
<td>48</td>
<td>32</td>
</tr>
<tr>
<td>Tourism</td>
<td>5560</td>
<td>52</td>
<td>48</td>
</tr>
<tr>
<td>Financial</td>
<td>6097</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>Entertainment</td>
<td>6122</td>
<td>40</td>
<td>40</td>
</tr>
</tbody>
</table>
Abstract—Contactless physiological monitoring based on radars is commonly performed when the subjects are immobile. However, in real life applications, subjects need to be monitored as they go about doing their daily activities. As the characteristics of the demodulated radar return signals vary with the level of activity, algorithms that provide robust estimation of vital signs under sedentary conditions will not be able to provide reliable estimates of vital signs when the level of human activity is high. Hence, level of human activity needs to be determined before any algorithm for vital sign estimation can be applied. This paper proposes a method for classifying 3 levels of human activity in radar return signals acquired using a single channel CW Doppler radar system. The proposed method may be used for selection of appropriate estimation algorithms for vital signs depending on the level of activity. In total, 43 features are extracted from the raw data. Six classifiers are tested and four subspace mapping techniques are used to reduce the dimensionality from 43 dimensions to 6. The top performing classifier for the 43 dimensional data is a random forest classifier which achieves 96.1% overall accuracy, while the top performing classifier in 6 dimensional feature space is tied between the random forest and multilayer perceptron with 95.1% overall accuracy.

I. INTRODUCTION

RADAR sensors offer non-contact sensing capabilities that make them well suited for the field of human physiology monitoring. Estimation of breathing and heart rate from radar return signals is established in literature [1]. One common issue discussed in many works is the inability of estimation algorithms to produce accurate results during periods of gross body movement [2]. Large body movements typically obscure smaller physiological signals such as breathing and heart rate. These body movements need not even be large; movements as small as typing on a laptop may be significant enough to cause estimation errors [3]. If radar is to be used for continuous monitoring of human vitals, this problem of gross body movement must be overcome.

Current state-of-the-art non-contact physiological monitoring systems are able to provide reliable vital sign estimates only in sedentary conditions. Large body movements can obscure the micro-Doppler variations due to chest-wall and abdominal movements. Estimation algorithms that produce vital sign estimates when the subjects are sedentary cannot be expected to work well when there are large body movements. The degradation in accuracy and confidence of estimates should be anticipated when the radar returns are contaminated by large body movements. Since the presence of body movements may not be known a priori, it is necessary to have algorithms in place to determine the presence of large body movements before estimation of vital signs is attempted. This is a paradigm shift from the existing way of obtaining estimates from non-contact physiological monitoring systems.

Received signals acquired while the subject is sedentary can be processed with linear signal processing techniques such as breathing and heart beat peak detection using FFT because there are no spectral components related to gross body movements overlapping spectra related to physiological signals in the frequency domain. Radar return signals in which a subject is either moving while sedentary (moving arms, legs, head, etc.) may have spectral components overlapping the breathing and heart beat signal spectra in the frequency domain. Because of this, non-linear signal processing techniques such as empirical mode decomposition (EMD) [4] must be used to decompose the signal into separate independent components. Even though the nonlinear approach may work for all signal types, it may be expensive computations wise when compared to linear approaches. A system, such as in Figure 1, could be designed to apply linear signal processing algorithms when the subject is sedentary, and non-linear signal processing algorithms when the subject is moving. Classification of ‘stationary’, ‘forward moving’, and ‘walking’ was performed in [5] with high accuracy in both unobstructed and through wall sensing cases. In [6], the classes ‘walking’, ‘running’ and ‘crawling’ were discriminable. Both [5] and [6] however, used quadrature radar architectures. Quadrature CW architectures have two 90 degree phase separated channels. The main benefit of using a quadrature type architecture is that phase information from the reflected radar signals can be extracted using either arc-tangent demodulation or Gram Schmidt Orthogonalization, eliminating non-linear distortions caused during demodulation. The problem with quadrature architectures however, is that they suffer from channel imbalance issues caused by phase errors, amplitude errors, and DC offsets in the channels [7]. These channel imbalances result in baseband signal distortion and reduced signal to noise ratio [8]. The Gram-Schmidt orthogonalization method can be used to eliminate this issue, however the phase error of the quadrature channel must be measured in a laboratory setting. These are motivating reasons for developing a classification system using
a single channel CW Doppler radar in this work.

A previous work proposed a classifier with 89.67% average overall accuracy [9] classifying data samples as 'breathing', 'stop breathing' and 'erratic signal' using a single channel CW Doppler radar. The classifier used 43 features which were mapped into 8 dimensions using uncorrelated linear discriminant analysis (ULDA) and the data was recorded with the radar placed at the same level of the subject and with the subject facing the radar. In the data collection for this paper, data was recorded with the radar mounted 2.70m above floor level and was collected in different orientations of the subject relative to the radar. This was done to better mimic what would be encountered in real life applications of the system- locating the radar in a place that allows it to irradiate a large portion of the room rather than a small section, and allowing the subject to be in different orientations relative to the radar.

In this paper 6 classifier types are tested. These classifiers accept 10 seconds of demodulated radar data and classifies it as one of three classes- 'sedentary and still', 'sedentary and moving', or 'walking'.

II. METHODOLOGY

1) Radar: The radar used in the data collection for this paper is a single channel binary phase coded Continuous Wave (CW) Doppler radar prototype model built by K&G Spectrum (Gatineau, Quebec) with an operating frequency of 24.125GHz. The beam-width of the transmitting and receiving antennas is 20 degrees in the horizontal axis and 70 degrees in the vertical axis. The raw data returns are recorded in a matrix format with \( m \) rows corresponding to 'radar frames' recorded at 905 S/s, and 9 columns corresponding to range 'bins'. The range bins correspond to 0.75 m distance in the radial direction from the radar antenna.

2) Classes: The three classes of activity levels chosen for this experiment are:
   - **Sedentary and still (Class 1)**: the subject is sitting, standing or lying in place, breathing normally or holding their breath and are not moving any body part. This class contains radar returns in which the subject is either breathing or holding their breath.
   - **Sedentary and moving (Class 2)**: the subject is sitting, standing, or lying in place, breathing normally and are moving one or more body parts. The level of movement as well as the body part(s) being moved varies throughout the recording; there was no specific protocol for moving, it was kept random so the developed classifier would not suffer from generalization problems.
   - **Walking (Class 3)**: the subject is breathing normally and is walking towards and away from the radar in a straight line at approximately 0.5m/s.

With exception of the walking class, each activity class is independent of posture and location within the room. Because of this, the relative angle and distance between the subject’s thorax and the radar varies. This was done so that the data represents as closely as possible what would be encountered in a real environment where the subject is not following controlled test cases.

3) Data Collection: The data collection was performed in a simulated penitentiary cell at Carleton University (Ottawa, ON). The room measured 3.15x3.35x2.95 m and contained a bed constructed of a cushion overlaying a chip board supported by 6 cinder blocks, a stainless steel toilet and sink unit, a plastic chair (not a fixed location) and a metal table located directly under the radar unit in one of the radars blind spots (Figure 1). The radar was mounted 2.70 m above the floor in one of the room’s corners. The walls of the room were made of concrete masonry units which should not be transparent to electromagnetic signals at the operating frequency of the radar used in this experiment. Thus, there should not be interference caused by any motion outside of the room. During data acquisition only one human was present in the radar field (other experimenters were in a corner of the room that was not within the field of the radar’s antenna).

There were five different types of test protocols for this experiment:
   - Lying on the bed in the left lateral recumbent position facing the radar. One minute of normal breathing followed by one minute of holding breath (or as long as comfortable) finished with one minute of normal breathing with arm/head movements. The total duration is three minutes.
   - Sitting on the bed facing the radar, back straight and hands resting on knees. One minute of normal breathing followed by one minute of holding breath (or as long as comfortable) finished with one minute of normal breathing with arms/head movements. The total duration is three minutes.
   - Standing at mark C on the floor facing radar. One minute of normal breathing followed by one minute of holding breath (or as long as comfortable) finished with one minute of normal breathing with arms/head/knees movements. Repeated twice at marks B and A on the floor. The total duration is nine minutes.
clearance was obtained for conducting these experiments. Appropriate ethics

These test protocols (totaling 27 minutes) were performed by four subjects (3 male and 1 female). These test protocols (totaling 27 minutes) were performed by four subjects (3 male and 1 female). Appropriate ethics clearance was obtained for conducting these experiments.

- Standing at mark A on the floor with back to radar. One minute of normal breathing followed by one minute of holding breath (or as long as comfortable) finished with one minute of normal breathing with arms/knees/head movements. Repeated twice at marks B and C on the floor. The total duration is nine minutes.
- Walking slowly back and forth between marks C and A on the floor, normal breathing for three minutes. The total duration is three minutes.

These test protocols (totaling 27 minutes) were performed by four subjects (3 male and 1 female). Appropriate ethics clearance was obtained for conducting these experiments.

4) Preprocessing: The data was segmented and saved into one minute files. Each 1 minute segment of radar data (which consisted of 54300 samples in each of the 9 zones) was saved into a CSV file. 214 CSV files in total were created. The zone in which the subject was present was estimated by finding the highest energy zone (after the mean of the signal was removed). Non-overlapping data segments of ten seconds were used for processing. This means that each of the one minute files contained six separate segments (642 total samples). The reason for using non-overlapping segments is that each of the segments will be unique to the classifier so that is cannot be trained and tested on the same portions of data. Of the 642 segments, 67 were removed because they were either corrupted with noise, they were transitional segments (i.e. the sample was composed of data that belonged to two classes), or had been zero padded during the segmentation process. This left 575 segments for classification purposes. The segments were then filtered with a second order 0.08Hz high pass Butterworth filter (for removal of DC bias due to clutter) and a second order 20Hz low pass Butterworth filter (for removal of possible high frequency noise including 60Hz power line interference).

5) Feature Extraction: In total 43 features were extracted from each signal segment. The features included 11 time domain features and 32 frequency domain features. The time domain features included: max-correlation of original signal, max-correlation of signal in 0.2-0.333Hz band, max-correlation of signal in 0.667-3Hz band, root mean square value, zero crossing rate, turns count, variance, skewness, kurtosis, mobility, and form-factor. The frequency domain features included: total power, mean frequency, median frequency, spectral variance, spectral skewness, fractional power in 0.2-0.667Hz band (corresponding to breathing range and second harmonic), fractional power in 0.667-3Hz band (corresponding to heart beat range), fractional power in three noise frequency bands- fractional power in 3-5Hz band, fractional power in 5-11Hz band, fractional power of signal greater than 11Hz, all 15 possible ratios of the aforementioned fractional powers, fractional power in 0.08-0.35Hz band (corresponding to range of fundamental breathing frequency), fractional power in 0.36-0.7Hz band (corresponding to range of second breathing frequency harmonic), the ratio of the fundamental and second breathing frequency harmonic, Shannon entropy of entire spectrum, Shannon entropy of breathing range, and Shannon entropy of heart beat range. The frequency domain features were extracted from the signal spectrum generated from a 216 point Welch-Periodogram.

6) Feature Space Reduction: Four different mapping techniques were used to reduce the dimensionality of the feature space. The techniques included isomap, principal component analysis (PCA), Sammon and T-SNE [10]. The data set was found to have an intrinsic dimensionality of 6, and hence a 6 dimensional feature space was constructed through the mapping techniques [10].

7) Classification: Six classifiers were trained and tested on both the 43 and 6 dimensional feature data. Training and validation was performed using 10 fold cross validation. Initially the data set had 318 instances of Class 1, 187 instances of Class 2 and 71 instances of Class 3. The classes were balanced by under-sampling the over-represented classes and oversampling the under-represented classes. This left 192 instances per class. In order to remove the influence that individual units or scales of measurement had on the feature set, all features were converted to z scores. The z score has zero mean and unit standard deviation.

Simple classifiers such as K-nearest neighbors and naïve Bayes were used for baseline measures of performance. More complicated classifiers such as the random forest and multilayer perceptron, should give higher performance compared to the simple classifiers [11].

A. Naïve Bayes

Naïve Bayes classifiers work by computing the conditional probability of an observation belonging to a particular class using Bayes theorem [12]. Naïve refers to the fact that the classifier assumes independence between all features. The training data is used to construct the probability model, and class is assigned to new data based on maximum likelihood.
B. K-Nearest Neighbors

K-nearest neighbors is a simple classifier that assigns classes based on the most similar data from the training set. When a new data sample is passed to the classifier, the K nearest data points in the feature space are found and the class that the majority of the data points belong to is assigned to the new data point [13]. Because it is a majority rules classifier, K should be an odd integer in order to avoid ambiguity in the voting process. In this paper K-NN is tested for K=1, 3 and 5. This classifier is a non-parametric type, which means it is data driven. This means that a larger data set is required for training compared to parametric classifiers. Because it is non-parametric, the decision boundary that is created can be highly complicated and result in over-fitting to the training set.

C. Random Forest

Random forest classifiers are built using many decision tree classifiers which are trained on random subsets of the data [14]. The prediction made by the random forest classifier is the mode of the combined outputs from each decision tree. This helps correct the issue of over-fitting that is inherent in decision tree classifiers because they are non-parametric. In this paper, 100 random trees were trained for each random forest classifier.

D. Multilayer Perceptron

Multilayer perceptrons (MLP) are feedforward ANNs. MLP classifiers have at least three layers (input, output and one or more hidden layers) [15]. The nodes of the hidden layers are non-linear, sigmoid, activation functions. In this paper the MLP used had a single hidden layer composed of \((\#\text{of attributes} + \#\text{of classes})/2\) nodes. MLP can become much higher in complexity than other classifiers, depending on the number of hidden layers. With increased complexity however, they can achieve high performance not capable with other simple classifiers.

III. RESULTS

The results of the classifiers can be seen in Table 1. The results represent the performance of the classifiers as tested on the hold out sets during 10 fold cross validation. The top performing classifier in terms of overall accuracy, or true positive (TP) rate is the random forest for 43 dimensions. TP rate is an important performance metric in this application because it represents the proportion of data samples that are being properly dealt with in the estimation stage. Similarly, false negative (FN) rate is important since it represents the proportion of data samples which the system applies the incorrect estimation algorithm on. Ideally a classifier should be chosen which has an acceptable compromise between TP and FN rates. Fortunately the random forest classifier with 43 dimensions also has the lowest FN rate among the 6 classifiers tested. The reason the random forest performs highest among the classifiers could be the result of the other classifiers either over-fitting to the data (downside to non-parametric classifiers) or incorrect assumptions about the underlying structure of the features (naïve Bayes).

An important result from this work is the performance of the six classifiers when the dimensionality of the feature space is reduced to 6. The overall accuracy of the classifiers changes only slightly even when 86% of the dimensions are eliminated. The reason this is possible is because the problem has an intrinsic dimensionality of 6 [10]. This means that the information removed from the feature space during the mapping procedure is almost entirely redundant. The top performing classifier in 6 dimensions is tied between the random forest (PCA mapping) and multilayer perceptron (TSNE mapping), each with TP rate of 0.951 and FN rate of 0.024.

In practical applications of non-contact human monitoring using radar, the computational complexity of the classifiers may be the deciding factor for which classifier is chosen. Random forest may be the highest performing classifier, but its computational complexity compared to a simpler classifier may make it less practical for real time decision making. Further analysis of computations required for each classifier, for training and for prediction, must be analyzed to determine which classifier is best for real time applications.

IV. CONCLUSION

Practical applications for non-contact physiological monitoring require the ability to able to process signals during periods of stillness or movement. In this paper 6 classifiers were tested on data containing three classes of movement- ‘sedentary and still’, ‘sedentary with movement’ and ‘walking’. These classifiers included naïve Bayes, K-nearest neighbors for K=1, 3, 5, random forest and multilayer perceptron. All 6 classifiers were trained and tested using 10 fold cross validation with class balanced sample sets. The results are the averaged results from the 10 cross validation folds. Feature space reduction was applied using four different mapping techniques. All classifiers obtained greater than 92% overall accuracy. The top performing classifier in terms of overall accuracy and false negative rate was the random forest with 96.1% accuracy and 0.020 FN rate. When the dimensionality of the feature space was reduced from 43 to 6 dimensions, the random forest and multilayer perceptron achieved an overall accuracy of 95.1% overall accuracy with FN rate of 0.024.

V. LIMITATIONS AND FUTURE WORK

The next step in this work is to combine the proposed classifier with breathing estimation and stop-breathing detection algorithms into a real time system that can then be tested in the simulated cell with subjects in a wider set of test cases. Because this system is designed for use in a penitentiary cell with a single occupant it should be modified to train solely on the data from that occupant so that it is tailored to their specific physiology. Since the level of movement of a human
subject can vary significantly, the classifier could be modified to use fuzzy logic in assigning a level of activity along a number line rather than ordinal class values. This would allow for a activity level threshold to be determined heuristically at which the accuracy of breathing and heart rate estimation drops below a certain accuracy. Feature evaluation methods should be used on the original 43 dimensional feature set to analyze each feature individually, and determine which are the strongest features. This could be used to design features that are capable of providing better class discrimination and reducing misclassification errors.

ACKNOWLEDGMENT

The authors would like to thank Correctional Service Canada for funding this project. The project is additionally funded through NSERC Discovery and OGS grants.

REFERENCES


Abstract— This paper presents a comparison of a shape-based and texture-based feature extraction technique. The method use shape features extracted by curvelet transform and texture features extracted by local binary pattern for classification of normal and abnormal mammograms and further classification of abnormal mammograms into cancerous (malignant) or non-cancerous (benign) masses. In this experiment, normal and abnormal (mass only) mammograms were obtained from Mammographic image analysis Society (MIAS). The breast images were pre-processed and segmented using histogram normalization and active contour, respectively. Then K-Nearest neighbor classifier was used for classification to evaluate the features extracted by curvelet transform and local binary pattern. The classifier produced a classification accuracy of 73.3% using curvelet transform features and 83.3% using local binary pattern for normal/abnormal classification phase. For benign/malignant classification phase, accuracy of 72% was obtained using curvelet transform features and 84% using local binary pattern features. The experimental results suggest that the texture-based feature extraction technique yield better classification results than the shape-based feature extraction technique.

Keywords— mammograms, features, shape-based, texture-based, mass, feature extraction, benign, malignant

I. INTRODUCTION

Cancer is an abnormal continual multiplication of cells. The cells divide uncontrollably and may grow into adjacent tissue or spread to distant parts of the body. These cells eventually become large enough to produce lumps, masses, or tumour that can be detected [1]. Breast cancer is the most common type of cancer afflicting women in most parts of the world [2].

The initial symptom of a breast cancer is the formation of a lump. This is due to tiny deposits of calcium called micro-calculifications and tumors called masses [3]. Breast mass is used to indicate a localized swelling or tumor in the breast which is usually described by its location, size, shape and margin characteristics [4]. A mass can either be benign (breast-abscess, fat or necrosis) or malignant (cancer) [5].

Early detection of breast cancers often leads to more effective treatment with fewer side effects. However, its early detection is difficult since there are no symptoms during the first stages of breast cancer development [6]. Different imaging techniques such as magnetic resonance, mammography and ultrasound images are possible for early detection of breast cancer [7].

Mammography is at present, the best available examination for the detection of early signs of breast cancer and it can reveal pronounced evidence of abnormality, such as masses [8]. Different imaging reports are used for mammogram findings and results among which such is Breast Imaging Reporting and Data Systems (BIRADS), a communication tool and a standardize mammographic report used in most of the countries with breast cancer screening program [9]. Despite the huge success of mammography in mass detection, it is still very difficult for radiologists to interpret a mammogram correctly.

Mass detection from mammograms is challenging because abnormal masses usually mix with the homogeneous tissues of the breast, the images provided by different patients have different dynamics of intensity and present a weak contrast and the size of the significant details can be very small [4]. An automated system can help overcome these problems. The most recently employed measure to interpret a mammogram more reliably and efficiently is via the use of Computer Aided Detection/Diagnosis (CAD) system [10].

Computer Aided Detection (CADe) system is designed to aid radiologists detection of visible findings that are suspicious for lesions while Computer Aided Diagnosis (CADx) system are designed to assist to achieve accurate classification of the detected regions [11]. A CAD system is generally composed of four steps; pre-processing, segmentation, feature extraction, and classification [12]. However, the success of a CAD system depends directly on the efficiency of the Feature Extraction Technique (FET). The accuracy of classification depends directly on the effectiveness of the feature extraction stage [13].
Several features are extracted from digital mammograms including texture features, intensity features and shape features [14]. Although, there are different types of abnormalities present which results in cancer; they are: Masses, micro classification, architectural distortion and bilateral asymmetry. These types of cancer could be detected easily at early stage i.e. benign stage and if the abnormality gets denser and spreads in a large area it becomes Malignant.

This paper focuses on extraction of masses only. Masses are characterized by their location, size, shape and margin [15] and the large variation in size and shape in which masse can appear, make mass extraction a challenging task for researchers. In additional, at the most of cases, mammograms exhibit poor image contrast tissue density (fatty, dense or glandular), then tissue can overlap with breast tumor region as the mass abnormality [16]. This research compared shape and texture feature classification for identifying presence of cancer and classification of masses so that further classification could be done to find cancerous and non-cancerous masses using accuracy, sensitivity and specificity as metrics. The various steps in detection and classification of masses in this work are:

i) Pre-processing using Histogram Equalization
ii) Segmentation using Active Contour
iii) Extraction of Shape Features using Curvelet Transform
iv) Extraction of Texture Features using Local Binary Pattern
v) Classification of shape and texture features using KNN

II. RELATED WORK

Mass detection problem has attracted the attention of many researchers, and many techniques have been proposed so far. Most of the existing methods differ in the types of features that have been used for false positive reduction and benign-malignant classification and the way these features have been extracted.

Wei et al., [17] extracted multi-resolution texture features from wavelet coefficients and used them for the discrimination of masses from normal breast tissues, using linear discriminant analysis for classifying the ROIs as mass or non-mass. This method was tested with 168 ROIs containing biopsy-proven masses and 504 ROIs containing normal parenchyma, and resulted in 0.89 and 0.86 for the training and test groups. The system developed in this research is evaluated based on discrimination and classification performance only.

Liu, Babbs and Delp [18] proved that the use of multi-resolution analysis of mammograms improve the effectiveness of any diagnosis system based on wavelets coefficients. The achieved successful rate was 84.2% because of the absence of image pre-processing technique in the developed system and the size of the dataset used for testing was small (50 images were used)

Meenalogosini, Janet and Kannan [19] proposed an approach to develop the CAD of breast cancer. Texture of segmented image is extracted using Grey Level 0Co-occurrence matrix and Local Binary Pattern method. Extracted features are classified using Support Vector Machine. The result shows 98.8% sensitivity and 97.4% specificity. The system was developed using two different databases with different pixel background.

III. IMAGE ACQUISITION

Mammograms used in this research are retrieved from the MIAS (Mammographic Image Analysis Society) database which consists of 322 images, representing 161 breast pairs in the Medio-Lateral Oblique (MLO) view. The images originated as the product of the film-screen mammogram process in the United Kingdom National Breast Screening Program [20]. A typical mammogram is displayed in Fig. 1 [21]. The coordinates of the center of each diagnosed abnormality were provided along with the approximate radius of a circle enclosing the abnormality.

Three Hundred (300) mammographic images were selected such that Two Hundred and Ten (210) of them were normal, Fifty 50 of them were benign and Forty (40) of them were malignant. The original size of the mammogram (1024 x 1024 pixels) was reduced to half of the original size in order to reduce the computation time and increase the classification accuracy of the system. The mammograms were cropped from the edges to remove all unwanted element. Combination of Binary Mask and free hand tools which is a MATLAB built-in function was used for the cropping of the mammograms. The used mammograms were being previously classified by radiologists in order to be possible to study the efficiency of the research.

IV. PREPROCESSING

Histogram is a function showing the number of occurrences of a particular gray-level value, it shows how individual brightness levels are occupied in an image, the image contrast is measured by the range of the brightness levels. Histogram normalization is one of the best methods for enhancing the contrast of an image. In this research, the histogram of the original image is transformed by using its normalized cumulative sum.

If the overall brightness of an image is controlled by a level l and the range is controlled by “k”, the brightness of the points in the new image N can be related to the brightness in old image O by using equation (1).

\[ N_{x,y} = K \times O_{x,y} + 1 \quad \forall x, y \in [1, N] \quad (1) \]

This function maps the values in the original image to new values in the new image created. Then the intensity values of the original image were mapped to new intensity to give a uniform histogram of intensity values and for normalizing the intensity using equation (2) [22].
\[ N_r, N_m = \frac{N_{max} - N_{min}}{O_{max} - O_{min}} \times (O_{max} - O_{min}) + N_{min} \quad \forall i, y \in 1, N \] (2)

Pre-processing operation was carried out in the Medio-Lateral Oblique Mammograms (MLO) views, where the pectoral muscle in the mammogram is slightly brighter compared with the rest of the breast tissue. The pectoral muscle and the breast part edges are two relatively homogenous regions meeting at a boundary. MATLAB function imadjust was used for contrast and brightness adjustment.

V. SEGMENTATION

The cancerous region was segmented from the fatty mammogram using active contour algorithm. Active contour algorithm uses energy measure of a snake \( E_{snake} \) contains internal and external forces. The internal forces regulate the ability of the contour to stretch or bend at a specific point while preserving some degree of geometric smoothness while the external forces attract the contour to specific image features using equation (3) [23].

\[ E_{snake} = \int_{v(s)} E_{int}(v(s)) + E_{image}(v(s)) + E_{ext}(v(s)) \, ds \] (3)

Where \( E_{int} \) represents the internal energy of the contour due to bending or discontinuities, \( E_{image} \) represents the energy from the image forces, \( E_{ext} \) represents the optional external constraint energy and \( v(s) \) represents the set of points of the snake contour. The snake's general form, described by Kass et al. (1987) represents a contour by a vector having the arc length. The aim of the snake is to evolve by minimizing equation (3).

New snake contours are those with lower energy and are a better match to the target feature (according to the values of \( E_{int}, E_{image}, \) and \( E_{ext} \)) than the original set of points from which the active contour has evolved. In this manner, a set of point \( v(s) \) is chosen such that the change in energy measure is zero \( \frac{\partial E_{snake}}{\partial v} = 0 \). The internal energy that maintains the snake’s structure and resists singular deformations is presented in equation (4).

\[ E_{int} = \alpha(s) \left| \frac{\partial v(s)}{\partial s} \right|^2 + \beta(s) \left| \frac{\partial^2 v(s)}{\partial s^2} \right|^2 \] (4)

The values of \( \alpha \) and \( \beta \) at a point determine the extent to which a contour is allowed to stretch or bend. If \( \alpha \) is zero, discontinuities can occur at that point. If \( \beta \) is zero, curve discontinuities are permitted. The total image energy, \( E_{image} \) presents three different energy functionalities, attracting the snake to lines, edges and terminations is presented in equation (5), their energy is denoted \( E_{line}, E_{edge} \) and \( E_{term} \) respectively, and are controlled by weighting coefficients \( w_{line}, w_{edge} \) and \( w_{term} \) respectively.

\[ E_{image} = w_{line} E_{line} + w_{edge} E_{edge} + w_{term} E_{term} \] (5)

Active contour method was used to locate the mass and its boundaries. Equation (3) was assigned to put a label to every pixel in an image such that pixels with the same label share certain characteristics pixel in an image such that pixels with the same label share certain characteristics. Forces were extracted from the image itself using (4).

The result of image segmentation was a set of segments that collectively cover a set of contours (ROI) segmented from the image. Each of the pixels in a region is similar with respect to any characteristic or computed property, such as shape, density or margin. The properties were computed using (5).

V. EXTRACTION OF SHAPE FEATURES

Shape features were extracted using Curvelet Transform (CT). The curvelet transform at different scales and directions span the entire frequency space using fewer coefficients for a given accuracy of reconstruction. CT has spatial and frequency parameters \( x \) and \( \theta \) respectively, represented in polar coordinates as \( r \) and \( \theta \). A pair of windows \( W(r) \) and \( V(r) \) is defined as radial window and angular window, respectively which obey the admissibility conditions as displayed in (6) and (7) and Fig. 2 [24].

\[ \sum_{j=0}^{N} w^{2}(2^{j}r) = 1, \quad r \in \left( \frac{3}{4}, \frac{3}{2} \right) \] (6)
\[ \sum_{t=0}^{N} v^{2}(t-l) = 1, \quad t \in \left( -\frac{1}{2}, \frac{1}{2} \right) \] (7)

![Fig. 2: Organization of the curvelet transforms](image)

For each \( j \geq j_a \) a frequency window \( U_j \) is defined in the Fourier domain by equation (8).

\[ U_j(r, \theta) = 2^{j} w(2^{j} r) \left( \frac{2^{j/2} \theta}{2\pi} \right) \] (8)

The support of \( U_j \) is a polar wedge defined by support of \( W \) and \( V \) and is applied to scale dependant windows with widths in radial and angular direction. Equation (9) is then used to obtain real valued curvelet.

\[ U_j(r, \theta) + U_j(r, \theta + \pi) \] (9)

Once the images were segmented as described, the frequency plane (the extracted ROI) obtained is divided into dyadic coronae using (6) and (7). Each corona is partitioned into angular wedges which abide by the parabolic aspect ratio using (8). Hence, orientation is taken into account in the scale-space description of the image to create a frequency window using (9) [25]. CT was applied on the segmented images at a specific scale and orientation. The approximate sub-band obtained from the transform was resized to \( a \times b \) and divided into \( k \) regions each of size \( m \times n \) pixels.
VI. EXTRACTION OF TEXTURE FEATURES

Texture features were extracted using Local Binary Pattern (LBP). For LBP operator, the coordinates of the center pixels are \((x_c, y_c)\) then the coordinates of his P neighbors \((x_p, y_p)\) on the edge of the circle with radius R can be calculated with the sine and cosine as presented in equation (10) and (11).

\[
x_p = x_c + R \cos \left(2 \frac{\lambda p}{P} \right) \quad (10)
\]
\[
y_p = y_p + R \sin \left(2 \frac{\lambda p}{P} \right) \quad (11)
\]

If the grey value of the center pixel is \(g_c\) and the grey values of his neighbors are \(g_p\), with \(P=0, ..., P-1\), then the ROI (T) in the neighborhood of the pixel \((x_c, y_c)\) can be defined as equation (12). Although there is an invariant against the grey scale shifts, the differences are affected by scaling. To achieve invariance with respect to any monotonic transformation of the grey scale, only the signs of the differences are considered.

\[
T = \{g_c, g_o, g_{-1}, ..., g_{p-1}, g_r, -g_r\} \quad (12)
\]

This means that in the case a point on the circle has a higher grey value than the center pixel (or the same value), a 1 is assigned to that point and else it gets a 0 in equation (12), where \(s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}\). LBP for pixel \((x_c, y_c)\) can be produce on a neighborhood pixel \((R)\). A threshold is set using (12), if the neighbor value is higher than the center value of 1 is assigned to that position otherwise 0 is assigned. The sign parameter is set using (13) and the corresponding binomial weights using (14) [26].

\[
T = \{s(g_o - g_c), s(g_{-1} - g_c), ..., s(g_{p-1} - g_c)\} \quad (13)
\]
\[
LBP_{P,R(x_c,y_c)} = \sum_{p=0}^{P-1} s(g_p - g_r)2^p \quad (14)
\]

After identifying the LBP pattern of each pixel, the whole texture image is represented by building a histogram which is used as a feature extractor. The LBP histogram contains information about the distribution of the local micro-patterns, such as edges, spots and flat areas, over the whole image so it can be used to statistically describe image texture.

Basic LBP operator is displayed in Fig. 3 [12]. Extracting the texture features, the centre value is first fix and then compared with neighbor values. Equation (10) and (11) were used to assign parameters for the center pixel (P) and neighborhood pixel (R). A threshold is set using (12), if the neighbor value is higher than the center value of 1 is assigned to that position otherwise 0 is assigned. The sign parameter is set using (13) and the corresponding binomial weights using (14) to get the LBP value for each mammogram.

![Fig. 3: Basic LBP operator](image.png)

VII. CLASSIFICATION OF SHAPE AND TEXTURE FEATURES

The K-Nearest Neighbor (KNN) classifier is the first and foremost extension of the nearest neighbor classifier, which is a versatile multivariate statistical technique. The efficiency of KNN has already been verified against other statistical techniques and some neural networks. KNN has been used in the face recognition system and the outcome has significantly improved the accuracy of the system [17]. KNN algorithm is the classifier used to classify the extracted features into normal and abnormal mammographic images and further classifying abnormal mammographic image into benign and malignant.

Given a training set and a test pattern as presented in equation (15), KNN computes the similarity (distance) between the nearest k neighbors. The neighbors were taken from a set of features extracted for which the correct classification. The k-nearest neighbor algorithm is sensitive to the local structure of the data.

\[
e^2(k) = \sum_{i=1}^{k} (c_i - c_k) \quad (15)
\]

The algorithm was used to find a set of k objects in the training data that are close to the test pattern, and base the assignment of a label on the majority of a particular class of its neighborhood.

VIII. METHODOLOGY

The dataset used for the process of executing this research composed of Three Hundred (300) mammographic images from the MIAS database which includes Two Hundred and Ten 210 normal and Ninety (90) abnormal, the abnormal mammograms includes Fifty (50) benign and Forty (40) malignant masses. 60% (180 out of 300) set of the images were used for training and 40% (120 out of 300) were used for testing. The testing and training processes are demonstrated in Fig. 4.

![Fig. 4: Training and testing process of the developed methodology](image.png)

There was consistency in the dynamic range of the intensity values during the pre-processing process which makes the Region of Interest (ROI) clearer. The mammograms were segmented into foreground and background. The MIAS radiologist classification was stored as “actual value”, indicating normal images as 1, malignant as 2 and benign as 3 respectively. The effectiveness of the shape and texture feature extraction methods was evaluated and compared using KNN classifier. Two classification phases was done; first phase classified the breast images into normal and abnormal and the abnormal masses were further classified into benign and malignant for the second phase.
The metrics used for evaluating the performance of CT and LBP are sensitivity using equation (16) and (17), specificity using (18) and (19) and accuracy of the system using equation (20) and (21). Sensitivity measures how good the test is at detecting abnormal conditions. Specificity measures how good the test is at detecting normal conditions. Accuracy measures the degree of exactness of the system.

\[
\text{Sensitivity} \gamma_1 = \frac{TP}{TP + FN} \quad (16)
\]

\[
\text{Specificity} \gamma_2 = \frac{TN}{TN + FP} \quad (17)
\]

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (20)
\]

The indices used to determine the metrics are:

TN - Normal classified as Normal, FP - Normal classified as Abnormal, TP-Abnormal classified as Abnormal, FN - Abnormal classified as Normal, TN_{BM} - Benign classified as Benign, FP_{BM} - Benign classified as Malignant, TP_{M} - Malignant classified as Malignant, FN_{M} - Malignant classified as Benign

IX. RESULT AND DISCUSSION

The testing process was done for 120 mammograms containing 95 normal mammograms, 14 benign (non-cancerous) masses and 11 malignant (cancerous) masses. The mammograms were classified into Normal/Abnormal, while the abnormal masses were classified into Benign/Malignant according to BI-RADS. It was observed as demonstrated in TABLE I (a) that using CT features, the classifier correctly classified 70 mammograms as Normal while 25 mammograms were misclassified as Abnormal. 18 mammograms were correctly classified as Abnormal while 7 mammograms were wrongly classified as Normal for Normal/Abnormal classification phase.

For Benign/Malignant classification phase as displayed in TABLE I (b), 10 abnormal masses were correctly identified as Benign while 4 abnormal masses were misidentified as Malignant, 8 abnormal masses were correctly identified as Malignant and 3 abnormal masses were wrongly identified as Benign.

Using LBP features, the classifier categorized 79 mammograms as Normal, misclassified 16 mammograms as Abnormal, correctly identified 21 mammograms as Abnormal and misidentified 4 mammograms as Normal as displayed in TABLE II (a) for Normal/Abnormal classification phase.

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal (95)</td>
<td>Normal</td>
</tr>
<tr>
<td></td>
<td>Abnormal</td>
</tr>
<tr>
<td>79 (TN)</td>
<td>16 (FP)</td>
</tr>
<tr>
<td>Abnormal (25)</td>
<td>Normal</td>
</tr>
<tr>
<td></td>
<td>Abnormal</td>
</tr>
<tr>
<td>4 (FN)</td>
<td>21 (TP)</td>
</tr>
</tbody>
</table>

For Benign/Malignant classification phase, the classifier classified 12 abnormal masses appropriately as Benign, misclassified 2 abnormal masses as Malignant, correctly identified 9 abnormal masses as Malignant and misidentified 2 abnormal masses as Benign as shown in TABLE II (b).

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal (14)</td>
<td>Benign</td>
</tr>
<tr>
<td></td>
<td>Malignant</td>
</tr>
<tr>
<td>12 (TN)</td>
<td>2 (FP)</td>
</tr>
<tr>
<td>Abnormal (11)</td>
<td>Benign</td>
</tr>
<tr>
<td></td>
<td>Malignant</td>
</tr>
<tr>
<td>2 (FN)</td>
<td>9 (TP)</td>
</tr>
</tbody>
</table>

Performance of extracted features in terms of accuracy, sensitivity and specificity as shown in TABLE III (a) using; CT features are 73.3, 72.0 and 73.7%, respectively and 83.3, 84.0 and 83.2%, respectively using LBP features for Normal/Abnormal classification phase.

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (%)</td>
<td>CT</td>
</tr>
<tr>
<td></td>
<td>LBP</td>
</tr>
<tr>
<td>73.3</td>
<td>83.3</td>
</tr>
<tr>
<td>Sensitivity (%)</td>
<td>CT</td>
</tr>
<tr>
<td></td>
<td>LBP</td>
</tr>
<tr>
<td>72.0</td>
<td>84.0</td>
</tr>
<tr>
<td>Specificity (%)</td>
<td>CT</td>
</tr>
<tr>
<td></td>
<td>LBP</td>
</tr>
<tr>
<td>73.7</td>
<td>83.2</td>
</tr>
</tbody>
</table>

For Benign/Malignant classification phase, an accuracy, sensitivity and specificity of 72.0, 72.0 and 71.4%, respectively was obtained using CT features and 84.0, 81.8 and 85.7%, respectively using LBP features as displayed TABLE III (b).

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (%)</td>
<td>CT</td>
</tr>
<tr>
<td></td>
<td>LBP</td>
</tr>
<tr>
<td>72.0</td>
<td>84.0</td>
</tr>
<tr>
<td>Sensitivity (%)</td>
<td>CT</td>
</tr>
<tr>
<td></td>
<td>LBP</td>
</tr>
<tr>
<td>72.0</td>
<td>81.8</td>
</tr>
<tr>
<td>Specificity (%)</td>
<td>CT</td>
</tr>
<tr>
<td></td>
<td>LBP</td>
</tr>
<tr>
<td>71.4</td>
<td>85.7</td>
</tr>
</tbody>
</table>

The results from TABLE III (a) and (b) displayed that CT features using shape properties has a lower value of accuracy, sensitivity and specificity compared to LBP. This is because shape features include only the outer features of breast which are not sufficient enough for effective classification of mammograms. As shown in Fig. 5, LBP features out-perform CT features because it contains textural features of the breast image. Texture features can differentiate cancerous and non-cancerous masses better than shape features because it possesses the homogeneous tissues properties of the breast.
The presented results show that the sensitivity values for Normal/Abnormal classification phase is the same as the accuracy values for Benign/Malignant classification phase. This is to show that the two classification phases are interrelated.

![Bar chart showing the accuracy, sensitivity and specificity of CT and LBP features](image)

**REFERENCES**


Comparative Study of Iterative Back Projection and Discrete Algebraic Reconstruction Technique for Reconstruction of Low Resolution Images.

Adebola Oyinade Olowoye  
Dept.of Computer Science and Engineering  
Ladoke Akintola University of Technology (LAUTECH)  
Ogbomoso, Nigeria  
aoolowoye@pgschool.lautech.edu.ng

Temitope Tosin Adeyemo  
Dept.of Computer Science and Engineering  
Ladoke Akintola University of Technology (LAUTECH)  
Ogbomoso, Nigeria  
ttadeyemo@pgschool.lautech.edu.ng

Stephen Olatunde Olabiyisi  
Dept of Computer Science and Engineering  
Ladoke Akintola University of Technology (LAUTECH)  
Ogbomoso, Nigeria  
soolabiyisi@lautech.edu.ng

Elijah Olusayo Omidiora  
Dept of Computer Science and Engineering  
Ladoke Akintola University of Technology (LAUTECH)  
Ogbomoso, Nigeria  
eoomidioria@lautech.edu.ng

Temitayo Matthew Fagbola  
Dept of Computer Science  
Federal University Oye-Ekiti, Nigeria  
temitayo.fagbola@fuoye.edu.ng

Abstract—This paper presents a comparison between Iterative Back Projection (IBP) and Discrete Algebraic Reconstruction Technique (DART). In this experiment, images were acquired using a digital camera of 5Mega Pixel (MP). A total of 50 images were used. The Images were degraded by introducing noise, blur and shift in angle. DART and IBP were used to reconstruct the degraded images which were converted to grey scale. The techniques were evaluated using Peak Signal to Noise Ratio and Mean Square Error. The average Peak Signal to Noise Ratio (PSNR) gotten for IBP is 82.425 and the average Mean Square Error (MSE) gotten for IBP is 0.00077 while the average PSNR obtained for DART is 71.7025 and the MSE obtained is 0.00479. The experimental results suggest that IBP performs better than DART in reconstructing low resolution images.

Keywords—Low resolution; Reconstruction; IBP; DART.

Introduction

Image processing is a method employed to convert an image into digital form and perform some operations on it, in order to get an enhanced image or to extract some useful information from it [1]. It is a type of signal dispensation in which input is image, like video frame or photograph and output may be image or characteristics associated with that image. Usually image processing system includes treating images as two dimensional signals while applying already set signal processing methods to them.

Image processing can be done using analog or digital methods. Analog methods can be used for hard copies like printouts or photographs while digital processing technique helps in the manipulation of digital images by using computers [1]. Digital image processing is used to extract useful information from a given image of low quality.

High resolution (HR) means that the pixel density within an image is high, and therefore an HR image can offer more details that may be critical in various applications like medical images which are very helpful for doctors to make correct diagnosis [2].

The field of image processing has grown considerably during the last few decades with improvements in size, speed and cost effectiveness of the digital computers. Since high-resolution images and videos are important in many applications such as astronomy, military monitor, medical diagnosis and remote sensing, super resolution (SR) reconstruction has a great significance in obtaining images with more detailed information [3].

Super-resolution image reconstruction is a promising technique of digital imaging which attempts to reconstruct HR images by fusing the partial information contained within a number of under sampled low-resolution (LR) images of that scene during the image reconstruction process [4]. Super-resolution image reconstruction involves up-sampling of under-sampled images thereby filtering out distortions such as noise and blur. In comparison to various image enhancement techniques, super-resolution image reconstruction technique not only improves the quality of under-sampled, low-resolution images by increasing their spatial resolution but also attempts to filter out distortions [4].

Different techniques are available for the reconstruction of low resolution images but for the purpose of this experiment, IBP and DART are considered because they both achieve a resolved image in an iterative manner. Discrete Algebraic Reconstruction Technique (DART) stems from discrete tomography (DT), which is concerned with the problem of recovering images from their projections, where the images are assumed to consist of a small number (2 to 5) of gray values only [5]. If the set of gray levels is known or estimated
in advance, this prior knowledge can be exploited by the DT reconstruction algorithm.

Iterative Back Projection (IBP) based reconstruction is an inversion problem using the gradient descent method to find a solution. Since inverse problems are ill-posed problems, they need regularization or optimization of the solution [6]. The IBP technique is an ill-conditioned linear algebraic problem which resembles an undetermined system with never unique solution, hence can be solved using multi-objective optimization algorithms [7].

I. ITERATIVE BACK PROJECTION (IBP)

IBP technique was first proposed by Irani and Peleg [8], it can attain the High Resolution image interpolation and de-blurring simultaneously, while its motto is that the reconstructed HR image from the degraded Low Resolution image should produce the same observed LR image if passing it through the same blurring and down-sampling process. The IBP technique can minimize the reconstruction error by iteratively back-projecting the reconstruction error into the reconstructed image. IBP is a classical SR method with low computational complexity that can be applied in real time applications [9]. Figure 1 shows a pictorial representation of IBP which shows that with the help of an estimate of the reconstructed HR image and a model of the imaging process, a set of simulated LR images can be generated. Each simulated LR image is compared with the actual version and then the error can be used for correcting the estimated image. This method uses multiple simulated LR image of similar scene to find corresponding HR image [10].

This technique is very easy to understand but SR reconstruction is not unique due to ill posed nature of inverse problem. It should be noted that the selection of back-projection matrix affects the resolution. The original back-projection method is suffering from chessboard effect or ringing effect, especially at edges [11].

II. DISCRETE ALGEBRAIC RECONSTRUCTION TECHNIQUE (DART)

Discrete algebraic reconstruction stems from discrete tomography (DT), which is concerned with the problem of recovering images from their projections, where the images are assumed to consist of a small number (2 to 5) of gray values only [5]. Potential benefits of DT are an increase of the reconstruction quality and a reduction of the required number of projection images. The DT reconstruction problem, however, is generally under determined and the number of possible solutions can be substantial. To guide the reconstruction process toward an optimal as well as intuitive solution, the DART algorithm was proposed [5]. The basic algorithm of DART according to [12]:

Expressively, in each DART iteration, non-boundary pixels are assigned a given gray level in the prior known set of gray levels [12]. B as the boundary of the object in the thresholded image, which is defined as the set of all pixels that are adjacent to at least one pixel having a different gray level and I which denotes the remaining set of pixels such that all pixels in I are assigned their thresholded value. Then, idea of DART is to exclude the pixels in I from the regular Algebraic Reconstruction Method (ARM) iterations and, hence, only update the boundary pixels [12].

That is, several ARM iterations are performed on the pixels in B only. In this way, the number of variables in the linear equation system is significantly reduced, while the number of equations remains the same. In each iteration, the boundary pixels are allowed to vary independently, which may result in large local variations of the pixel values. To regularize the reconstruction algorithm, the boundary pixels are locally smoothed after applying ARM iteration on B. Subsequently, the resulting image is again thresholded, a new set of boundary pixels B is determined, and the next ARM iteration is run on the pixels in B. These steps are iterated until a certain convergence criterion is met [13].

Fig. 1. A representation of IBP.
III. REVIEW OF RELATED WORKS

High resolution (HR) images are very useful in different areas of life and are very applicable in areas like security, health o mention a few. Various researchers have explored the area of super resolution over the years using various methods.

Lin [14] developed an iterative back projection super resolution algorithm for low resolution image upsampling. A super resolution (SR) algorithm based on an iterative back project interpolation was proposed to obtain more reliable information in the edge area. For evaluation of the performance, the PSNR and the SSIM criteria of the proposed algorithm were compared with those of other methods. The experimental results of 28.743 dB in average PSNR and the 0.9165 in average SSIM show the superiority of the method. Furthermore, the computational complexity of the algorithm is reduced while maintaining similar image quality. However, it is computationally complex while Fagbola [15] developed a pose illumination invariant feature extraction technique for low resolution video feeds. IBP-MAP was developed using edge and non-edge details in Maximum A Posterior (MAP) priors. Resulting reconstruction error was estimated, minimized and back-projected by Iterative Back Projection (IBP). LDA-LBP-GWT was developed by fusing the facial features of Linear Discriminant Analysis (LDA), Local Binary Pattern (LBP) and Gabor Wavelet Transform (GWT) into a Single Feature Set (SFS). The SFS was optimized using particle swarm algorithm. The developed IBP-MAP technique produced PSNR of 33.77dB, ISNR of 11.82dB and NI of 34.00 and the developed LDA-LBP-GWT produced FA of 725, RA of 63.75%, RT of 103.65s and FR of 0.

Patel [16] addressed the problem of recovering a super resolved image from a single low resolution input. The technique is based on combining IBP method with the edge preserving Infinite Symmetrical Exponential Filter (ISEF). The method is applied on different type of images including face images, natural images and medical images; the performance was compared with a number of other algorithms, bilinear interpolation and nearest neighbor interpolation. The method showed marginal superiority to the existing method in terms of visual quality and PSNR. However, it is not reliable and efficient for highly degraded images when implemented in practice.

Zefreh et al. [5] developed a discrete algebraic reconstruction technique: a new approach for super resolution reconstruction of license plates. The researchers used DART to reconstruct a high resolution license plate from a set of low resolution camera images. The simulation and result shows that DART algorithm combines the efficiency of iterative algebraic methods from continuous tomography with the power of discrete algebraic reconstruction algorithm to compute an accurate HR image from a small number of LR images. However, one major drawback of DART is that it is less optimal.

Gompel et al. [13] developed a discrete tomography approach for super resolution micro-CT images: application to bone. The authors upsampled the reconstruction grid combined with DART algorithm in which the scanned objects are assumed to be composed of homogeneous materials. Their results show that the method generates reconstructions with significantly more details compared to conventional reconstruction algorithms. However, the method still suffers from optimality because the method used is not highly optimal for super resolution reconstruction.

IV. METHODOLOGY

The images used for this experiment were acquired using a digital camera of 5 mega pixel (mp). A total of 50 images were used in the course of this experiment. Images were acquired in (Joint Photographic Expert Group) JPEG formats. The images are kept in a file which was imported into the system one after the other as needed. Before the images are reconstructed, they are converted into grey scale. An example of the image used is presented in Figure 2.

![Fig. 2. A typical Image Used](image)

The initial HR image was degraded by introducing blurring, warping and downsampling to generate observed LR image using equation 1 [17]:

$$y_k = AH_k C(S_k)x + n_k = B_k (s_k)x + n_k (1)$$

Noise was introduced to the degraded image by using equation 2 [18]:

$$s_k = (\theta_k, c_k, d_k)^T$$

$$\theta_k$$ is the rotation angle, and $$c_k$$ and $$d_k$$ are the horizontal and vertical translations of the kth HR image with respect to the reference image x, the effects of downsampling, blurring and warping can be combined into a single system matrix $$B_k^N \times PN$$.

The performance metrics that will be used for the evaluation of the techniques are Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE).

$$\text{MSE} = \frac{1}{mn} \sum \sum_{m} \left\| f(i, j) - g(i, j) \right\|^2$$

(3)
Where m and n are the numbers of row and column in the input images respectively while g represents the matrix data of the degraded image and f is the matrix data of the original image.

\[
\text{PSNR is defined as:} \\
\text{PSNR} = 10 \log_{10} \left( \frac{\text{MAX}}{\text{MSE}} \right) \tag{4}
\]

where MAX is the maximum signal in the input image data type.

V. RESULTS AND DISCUSSION

The result obtained from the reconstruction of low resolution images using IBP and DART are presented in Tables (I, II, III, IV) respectively. In all the evaluations carried out, the PSNR and MSE were reported. For image 1, 10 iterations was selected, 10 max displacement, rotation angle 45, noise variance of 0.4 and blurring std of 0.4 with 4 number of LR images created, the result is shown in Table I. For image 2, a number of 8 iterations were made, 14 max displacement was selected, 27 rotation angle, 0.6 noise variance and 0.3 blurring std with 5 number of LR images created, the result is shown in Table II.

Image 3 was reconstructed with 12 numbers of iterations, 14 max displacement, 54 rotation angle, 0.2 noise variance and blurring std of 0.6 with 7 LR images created, the results are shown in Table III while image 4 was reconstructed with 9 iterations, 8 max displacement, rotation angle of 36, noise variance of 0.3 and 0.7 blurring std with 4 LR images created, the result is shown in Table IV. Table for four images were shown to show how each image evaluation was carried out.

Table I shows the performance evaluation results of the SRR method for image 1. The PSNR of IBP and DART measured in decibel (Db) are 77.14 and 71.99 respectively. This reveals that IBP produced the highest PSNR, followed by DART. In Table II, which presented the result of image 2, the PSNR of IBP and DART are 84.45 and 73.19 respectively.

A similar trend was presented in Table III where the PSNR of IBP and DART are 91.89 and 72.74 respectively, so also in Table IV with PSNR of IBP and DART are 76.22 and 68.87 respectively. By implication, IBP performs better that DART in terms of visual quality output.

Table II shows the performance evaluation results of the SRR methods for image 1. IBP and DART are presented in Tables (I, II, III, IV) respectively. In all the evaluations carried out, the PSNR and MSE were reported. For image 1, 10 iterations was selected, 10 max displacement, rotation angle 45, noise variance of 0.4 and blurring std of 0.4 with 4 number of LR images created, the result is shown in Table I. For image 2, a number of 8 iterations were made, 14 max displacement was selected, 27 rotation angle, 0.6 noise variance and 0.3 blurring std with 5 number of LR images created, the result is shown in Table II.

To evaluate the error from the techniques used, Mean Square Error was used. The lower the mean square error, the better the technique. From Table I, MSE for IBP and DART are 0.00125525 and 0.0041473 respectively. Likewise Table II shows that the MSE for IBP and DART as 0.000233589 and 0.00311425 respectively which implies that IBP is better in terms of error than DART.

Table III presents the result of image 3 with the PSNR of IBP and DART are 77.14 and 71.99 respectively, so also in Table IV with PSNR of IBP and DART are 84.45 and 73.19 respectively. By implication, IBP performs better that DART in terms of visual quality output.

Table IV shows the performance evaluation results of the SRR methods for image 4. IBP and DART are presented in Tables (I, II, III, IV) respectively. In all the evaluations carried out, the PSNR and MSE were reported. For image 1, 10 iterations was selected, 10 max displacement, rotation angle 45, noise variance of 0.4 and blurring std of 0.4 with 4 number of LR images created, the result is shown in Table I. For image 2, a number of 8 iterations were made, 14 max displacement was selected, 27 rotation angle, 0.6 noise variance and 0.3 blurring std with 5 number of LR images created, the result is shown in Table II.

To evaluate the error from the techniques used, Mean Square Error was used. The lower the mean square error, the better the technique. From Table I, MSE for IBP and DART are 0.00125525 and 0.0041473 respectively. Likewise Table II shows that the MSE for IBP and DART as 0.000233589 and 0.00311425 respectively which implies that IBP is better in terms of error than DART.

VI. SUMMARY OF THE RESULTS OBTAINED

The average of the evaluation results of SRR methods obtained from the 50 images presented in Table V. IBP produced an average PSNR of 83.801Db followed by DART which produced an average of 74.833Db. IBP produced an average MSE of 0.0009368 and DART produced an average of 0.00155289. It is evident that IBP is better than DART because the reconstruction error of IBP is lesser than that of DART which implies that IBP produces a better visual quality output than DART.
Reconstructing images are very useful in various fields like medicine because HR images provide more details which are useful in these fields. This paper presented a comparative study of IBP and DART super resolution techniques to enhance low resolution images.

Further research can be done in the development of super resolution techniques by combining two or more techniques so that the strength of one will compensate for the weakness of another in order to get a more visual quality output image.

ACKNOWLEDGMENT

The authors sincerely appreciate the support of TetFund for the anticipated financial support to attend this conference. We also thank the authorities of LAUTECH for recommending this work to TetFund and we sincerely appreciate the government of Nigeria, for the provision of funds to TetFund.

REFERENCES


Workshop: Image Mining - Mathematical Theory and Applications

Session 1 (195 min) (2 long papers, 7 regular papers)
Tuesday May 15, 2018: 14:40 - 18:00

Bernd Radig and Patrick Follmann
Training a classifier for automatic flash detection in million images from camera-traps

Bushra Jalil, Davide Moroni, Maria Antonietta Pascali, Ovidio Salvetti, Giuseppe Riccardo Leone and Massimo Martinelli
Multimodal image analysis for power line inspection

Nataly Ilyasova, Alexandr Kupriyanov, Rustam Paringer, Dmitriy Kirsh, Alexandr Shirokanev and Victor Soifer
Big Data Application for Smart Features Formation in Medical Diagnostic Tasks

Andrey Nasonov, Yakov Pchelintsev, Alexandra Nasonova and Andrey Krylov
A post-processing method for 3D fluorescence microscopy images

Yaoguang Zhong, Shawn Horvatic, Takuya Kaneko, Bing Ye and Jie Zhou
Automatic Video Mining in Animal Behavior Study using Statistical Shape Models

Alexei Baev, Vasilii Dubrovin, Aleksei Rozhentsov and Anna Eruslanova
Separation of arterial and venous vessels images from computer tomography

Sergey Dvoenko and Denis Pshenichny
On Conditionality of Pairwise Comparisons in Machine Learning

Christoph Theiß, Clemens-Alexander Brust and Joachim Denzler
Dataless Black-Box Model Comparison

Dimitri Korsch and Joachim Denzler

Session 2 (120 min) (3 regular papers, 6 short papers)
Wednesday May 16, 2018: 10:20 - 12:20

Galina Antonova
The features of the recognition of the region of efficiency of microwave devices

Mikhail Kopeliovich and Mikhail Petrushan
Approximation-based transformation of color signal for heart rate estimation with a webcam

Nikita Andriyanov, Vitaliy Dement'Ev and Konstantin Vasiliev
Development of the filtering algorithm for doubly stochastic images based on models with multiple roots of characteristic equations

Evgenii Semenishchev, Viacheslav Voronin, Igor Shraifel and Dmitri Chernishov

Copyright © 2018 by CENPARMI, Concordia University
The solution of the problem of simplifying the images for the subsequent minimization of the image bit depth
Viktor Chigrinskiy and Ivan Matveev

Nonlinearity of iris structure as way to improve recognition methods
Mikhail Korobkin, Gleb Odinokikh, Yurii Efimov, Ivan Solomatin and Ivan Matveev

Iris Segmentation in Challenging Conditions
Ivan Matveev and Vladimir Novik

Using optimal circular path method to match piecewise iris templates
Gleb Odinokikh, Yurii Efimov, Ivan Solomatin, Mikhail Korobkin and Ivan Matveev

Iris Anti-spoofing Solution for Mobile Biometric Applications
Viacheslav Voronin, Evgenii Semenishchev and Sos Agaian

Session 3 (190 min) (2 long papers, 5 regular papers, 2 short papers)
Wednesday May 16, 2018: 14:40 - 17:40

Afnan Garoot, Maedeh Safar, Nicola Nobile and Ching Suen
Computational Graphology Applied to Handwriting Images

Björn Barz and Joachim Denzler
Deep Learning is not a Matter of Depth but of Good Training (Extended Abstract)

Ali K. Hmood, Ching Y. Suen and Louisa Lam
Dynamic-HOG Descriptor for Structured Object Recognition: Case Study on Coins

Nikita Andriyanov and Vitaliy Dement'Ev
Increase efficiency of simple images segmentation using detectors based on doubly stochastic random fields

Igor Petrov and Vasily Ryazanov
Missing data imputation based on stochastic neighbor embedding

Elena Nelyubina, Vladimir Ryazanov and Alexander Vinogradov
Shape of Basic Clusters: Finding Coherent ELR-2s via Hough-type Transform

Viacheslav Antsiperov
Precedent-based Low Count Rate Image Intensity Estimation using Maximum Likelihood Distribution Descriptions

Igor Gurevich, Vera Yashina, Adil Tleubaev, Ayat Ospanov and Anton Vladimirov
A new mathematical method for automated identification of neurons on microscopic images

Igor Gurevich, Vera Yashina, Ayat Ospanov, Adil Tleubaev and Anatoly Fedorov
Automation of image analysis of the human fundus

Intellectual Information Technology for Symbol Extraction from Ill-structured Graphical Documents
In Germany, some of the highways, which cut through forests, are bridged to give animals the chance to cross them undamaged.

Some of these green-bridges are equipped with camera-traps to evaluate, which species and how many make use of this opportunity. The camera-traps generate huge amounts of images, which can not be analyzed by human specialists [1]. The Bavarian Highway Directorate gave us images and videos recorded by 6 cameras between end 2013 and spring 2017. The data volume amounts to 633,746 images. The cameras are installed at a bridge near the city Oberthulba. Most mammals are passing at night, 80 % between 5pm and 5am.

The pole on the left carries a camera. Most images are triggered at night and have to be illuminated by an infrared flashlight. In this example not only the recording camera triggered the flash but the opposite camera as well. In most cases the images corrupted in this way are useless outliers.

Our approach for the first step of the automatized
analysis to classify the passing species and to count the specimen is to setup a processing line of classifiers, which do the image mining to deliver these figures. Our goal is to minimize the human effort, e.g. in contrary to developing explicit image processing programs.

In this work we describe how Deep Learning methods ([2] and the survey part in [3]) can be applied to classify outliers in an image dataset using only a short annotation time.

The dataset at hand consists of images taken from several cameras on a highway bridge for game animals. The cameras were equipped with a motion sensor and a flash to capture animals like deer, boar, fox or bunnies passing by during day- or nighttime. The long-term goal of this project is to research the behavior of game animal (e.g. [4]) and how frequently these game animal bridges are used.

However, when an animal triggers the motion sensor and the flash of one camera (camera 1), quite often another camera (camera 2) on the opposite side of flash of camera 1 is visible (in the image of camera 2). In an additional set of images is the flash of a facing camera visible together with one or several animals that were targeted. Overall the dataset consists of three classes, namely <without flash>, <with flash> and <with flash and animals>. The task of this work is to automatically classify new, unseen images into one of these three classes.

In order to train and evaluate the classifier an excerpt of 2551 images from the whole dataset was labelled. The manual annotation of the images into the three classes took less than 1 hour, approximately 700 ms per image. There are 266 images in class <with flash> 39 in class <with flash and animals> and the remaining 2246 in <without flash>. That means, that images with flash, but especially images with flash and animals are underrepresented and therefore there is a bias towards the class without flash. A classification method that constantly returns the class <without flash> would already achieve an accuracy of 88%. In order to train HALCON’s1 enhanced Convolutional Neural Network (CNN), we split the dataset into a training, validation and test set using the ratio 3:1:1, respectively. The split is performed based on classes, such that all three sets contain the same ratio of classes. This leads to 1494 training images, 498 validation and 500 test images.

We fine-tuned the CNN that was pre-trained on a large classification dataset using the following parameters: We use a batch size of 128 images. The overall number of training epochs is 50. The learning rate is initialized with 0.001 and successively multiplied by 0.1 after 20 and 40 epochs.

As visualized, both the training and validation error drop to around 3% after the first epoch. Early stopping at iteration 520, i.e. epoch 47, further reduced the errors to 0% and 1.4%, respectively. The total training time on an NVIDIA GTX 1080 Ti GPU was approximately 5 minutes. Applying the trained model on the test set leads to an accuracy of 98.6%. The average time for preprocessing and classifying a single image is 12 ms. The confusion matrices for the validation and test set:

<table>
<thead>
<tr>
<th>Validation</th>
<th>ground truth</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>without flash</td>
</tr>
<tr>
<td>without flash</td>
<td>436</td>
</tr>
<tr>
<td>with flash</td>
<td>1</td>
</tr>
<tr>
<td>with flash and animal</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test</th>
<th>ground truth</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>without flash</td>
</tr>
<tr>
<td>without flash</td>
<td>438</td>
</tr>
<tr>
<td>with flash</td>
<td>0</td>
</tr>
</tbody>
</table>

1 http://www.mvtec.com
show that wrong classifications are made mostly for the class <with flash and animal>. The reason could be that there are only 23 images of this class in the training set and it is therefore very difficult for the model to learn its features. Another aspect might be that only parts of an animal might be visible since the animal is moving out of the image when it is captured. The following picture shows a typical false classification.

In summary, using a HALCON CNN classifier we achieve an accuracy of 98.6% at the task of identifying dataset outliers. At the same time, we reduce the classification speed from approximately 1250ms to 12ms, i.e. by a factor of over 100 compared to a human annotator.

The classifier processed the whole set of 633,746 images in less than 3 hours. The class statistics are:
- 76699 images in <with flash>
- 5196 images in <with flash and animal>
- 551,851 image in <without flash>

In future optimization, artificial data augmentation could be used to improve the accuracy. In particular, simulating flashes in images with or without animals should reduce the heavy bias towards images from <without flash>. The next step is to identify images from classes <without flash> and <with flash and animal> and locate the animals in the image. Difficult images are those in bad weather conditions or partial out of view animals.

Classification of the specimen and counting them would be the most challenging task for training the CNN with a reasonable effort in human annotation for training.

REFERENCES
Multimodal image analysis for power line inspection

Bushra Jalil, Giuseppe Riccardo Leone, Massimo Martinelli, Davide Moroni, Maria Antonietta Pascali and Ovidio Salvetti
Institute of Information Science and Technologies
National Research Council of Italy; Via Moruzzi, 1 – 56124-Pisa (IT)
Email: Name.Surname@isti.cnr.it

Abstract—The use of Unmanned Aerial Vehicles (UAVs) for environmental and industrial monitoring is constantly growing. At the same time, the demand for fast and robust algorithms for the analysis of the data acquired by drones during the inspections has increased. In this paper we provide a concise survey about a peculiar case study: the monitoring of the high-voltage power grid which includes: (i) the detection of the power lines and of the electric towers along with their components more subject to wear and tear; (ii) the diagnosis of maintenance status. In this work different algorithms from image processing are applied to visible and infrared thermal data, to track the power lines and to detect faults and anomalies. We applied Canny edge detection to identify significant transition followed by Hough transform to highlight power lines. The method significantly identify edges from the set of frames with good accuracy. The paper concludes with the description of the current work, which has been carried out in a research project, namely SCIADRO.

Index Terms—Image analysis, RGB Images, Infrared Images, Machine Learning, Wire detection, Insulators, Unmanned Aerial Vehicles

I. INTRODUCTION

The electricity distribution network must be constantly monitored to be safe and efficient. Hence, the detection of any defects at an early stage can save the life of the system, prevent damages, and predict future anomalies too. Much research on how to improve power lines detection and inspection has been carried out: the main objective is to reduce the time and cost of the monitoring, increasing the safety of the staff during the inspection, without losing in quality. The most common defects of the infrastructure are shown in Fig.1. Generally the inspection proceeds in steps, as follows: (i) detection of wires and cables; (ii) analysis of wires and cables; (iii) detection and classification of electric towers; (iv) analysis of tower components (insulators, hanging points).

There exist different methods to monitor the electric power lines. The most widely used method is the foot patrolling: a team of technicians inspects the lines by or with a ground vehicle. The team uses binoculars, and/or visible and infrared cameras, and this process may be tedious and long. Also, the inspection may be performed using a manned aerial vehicle, e.g. a helicopter equipped with visible or infrared cameras, acquiring images of conductors and insulators from high above the power lines, and quickly. In this case, the inspection is faster but usually more expensive. Then, the acquired data are either manually inspected by a skilled operator, or automatically processed for fault detection. When a fault is detected, skilled operators are required to climb in order to check and eventually fix the damage.

Recent advances in flight control techniques and image processing allow UAVs, equipped with a proper payload of acquisition (such as visible, thermal, and also multi-spectral cameras), to carry out fast inspection from some distance. Hence, in the last decade, UAVs are being used for a wide spectrum of applications, including the inspection and maintenance of power equipment. For example, there are algorithms able to perform automatic tracking of power lines using the GPS data of both the UAV and the electric towers. Compared with conventional inspection methods, UAV-based inspection has a number of advantages: it is more advanced, less expensive and safer. On the other hand, UAVs and manned aerial vehicles share some common problems: camera stabilization, pole tracking and automatic detection of anomalies. Monitoring an electrical infrastructure using UAVs requires to make the inspection fully automatic and almost real-time, and to get a reliable detection of defects and damages (such as hot spot in cables, or broken insulators).

In the following sections, firstly we present different computer vision methods used to analyze visible and infrared image data, for the detection and inspection of power lines and of the whole infrastructure. For sake of completeness we will present also other methods from machine learning, recently used to perform the task of recognition and classification of
electric towers and their components. Then we devote the remaining sections to the description of the ongoing work carried out in the framework of the SCIADRO project. The overall objectives of the project are: (i) achieving computer vision techniques and algorithms able to detect complex objects and extract information on local anomalies which might affect them; (ii) developing suitable policy and algorithms to effectively organize and guide the overall swarm motion and actions during a mission; (iii) studying, developing and demonstrating network architectures and protocols allowing communication among multiple drones within a swarm, possibly increasing the communication reliability towards the ground station.

II. STATE OF THE ART

The diagnosis of the electric infrastructure status is performed by analysing the visible or infrared data acquired during the inspection. To this aim, in the last two decades, thermal and infrared imaging have been increasingly studied and exploited both to test and inspect different electric parts, and to do preventive maintenance work. Recently, UAV-based inspection systems are used to comprehensively inspect power transmission line in urban and rural areas.

In this section we provide a review of the state of the art literature about the image processing methods used to analyse visible and infrared 2D images. Generally, such images may have several backgrounds: green fields, water, mountains or buildings. Therefore, in order to investigate the condition of the lines, the very first step is to identify them in the images. Detecting power lines from a cluttered background is one of the most important and challenging tasks. In general, all methods share this two-step process: i) Identify expected power lines and remove the background. ii) Connect the expected power lines and remove unwanted straight lines.

A. Processing of thermal-infrared data

Infrared thermography uses infrared sensors to capture images of thermal objects based on temperature variations. Also, thermal imaging is considered as a robust, non-destructive and contactless methodology to inspect electrical wires and connections (which are more subject to stress), as the inspection can be performed by keeping some distance, avoiding to halt or cut down electric supply during the inspection. Li et al. presented in [1] an image fusion algorithm (based on SIFT feature extraction) to support inspection robots in the automatic detection and identification of power equipments. Similarly, Larrauri et al. [2] identified areas of vegetation, trees and buildings close to power lines and calculated their distance from power lines. Simultaneously, the system processed the infrared images to detect hot spots in the power lines by estimating the threshold based on Otsu method [3] and later segment the lines from the background. Lages et al. in [4] describe a piece of software designed to acquire and process automatically two video streams, made of thermal and visible images. The processing results in the annotation of the video stream with faults. More recently, Luque-Vega et al. presented in [5] an inspection system based on a quadrotor helicopter, equipped with stereoscopic system made of a thermal-infrared and a color camera, to detect hot spots and other damages. The aerial video acquired are transmitted to the ground control station and then processed. Several of the above mentioned and similar methods require manual operations to adjust threshold values; measurement errors may occur, for example, due to the influence of the background, or to the lack of reference values for temperature. In this perspective Wronkowicz proposed in [6] the automatic detection of hot spots in power lines from IR images, without any reference temperature value. The thermal image is segmented by automatic thresholding.

B. Processing of RGB data

The recent boost of the UAV technology has increased the need of reliable automatic method of object tracking and detection from RGB images, supporting the UAV intelligence or improving the functionalities of a real-time monitoring system based on UAVs. Focusing on the detection and tracking of the power lines, Oberweger et al. presented in [7] a method to detect the insulators in aerial images, and to automatically analyze them for possible faults. The method is based on discriminative training of local gradient-based descriptors and a subsequent voting scheme for localization. In [8], Candamo describe a method to detect power lines from low quality videos, combining the motion estimation at the pixel level with the edge detection, followed by a windowed Hough transform. Yan et al. extracted straight line segments by Radon transform, followed by a Kalman filtering to connect segments into whole lines [9]. Zhang et al. [10] extracted the power lines by applying the Otsu thresholding to the gradient image. Hence, the straight lines are clustered and filtered; and Kalman filtering to track smoothly the power lines in the video sequence. Similarly, Li et al. in [11] proposed a more complex filter based on a simplified pulse coupled neural network model. This filter is used to remove the background noise as well as to generate edge maps, while the power lines are detected using an improved Hough transform. More recently, Li et al. [12] modeled the human attention mechanism model and the binocular vision to detect and track power lines in image sequences. The authors showed that the system is effective even in complex backgrounds and under different conditions. Song and Li described in [13] a method to detect not only the straight lines but also the curve ones. The method is based on a line segment pooling followed by a graph-cut model, to group together the lines corresponding to the same power line. We refer to [14], [15] for a more detailed summary of other methods.

Actually, monitoring the electric infrastructure means not only to understand the status of wires and cables, but also to detect and analyse electric towers and insulators.

Regarding the detection of electric towers and pylons, interesting results are reported by Sampedro et al. in [16] and by Dutta et al. in [17].

As regards the electric insulators, very important components of the electric grid, much research have been carried out. Recently, Zhao et al. proposed a method for the ac-
curate and real-time localization of multiple insulators with different angles [18]; while in [19] classical methods of image processing are used to separate the insulators from the complex background, and template matching to recognize the insulators. Most of classical methods are usually badly affected by changes in illumination and background; this may result in a poor generalization ability. Hence, the problem of detection, classification and location of insulators benefited from the recent advances in machine learning. In this line, Zhao et al. in [20] designed a multi patch convolutional neural network to extract deep features from the images representing insulators, and used such features to classify (via SVM) the status of the insulators; Liu et al. in [21] proposed to use a three-class dataset (insulator, tower, background) to train a convolution neural networks (CNN) used to predict the candidate insulator position. More recently, Tiantian et al. in [22] applied feature fusion techniques to the histogram of oriented gradients (HOG) feature and local binary pattern (LBP). Results reported are promising, but improvable, as the location is strongly dependent on the recognition of the insulator.

### III. The SCIADRO Project

This paper deals with the ongoing work being done in SCIADRO [23], a research project. The project aims at developing the enabling technologies, which are key to accomplishing a rather rich and diverse span of missions through the use of a swarm of drones within a civilian environment. The main idea of our work in the SCIADRO project is to provide a tool to support simultaneously the detection of the infrastructure components; and the diagnosis of the maintenance status of the detected insulators. Also, such algorithms should be specifically designed for the collaborative setting of an UAV swarm. As shown in Fig. 2, the image processing aims at the detection and analysis of the main components of the electrical infrastructure: electric towers, insulators, and conductors.

Traditional image processing is currently used to track power wires, both in thermal and visible images (possibly registered); while a multi-layer perceptron neural network will be used both to predict whether the region inside an image is a tower (or not), and to distinguish the tower type on the basis of a dataset of training. A rich dataset of images is needed to train the neural networks. A feature fusion step will be probably added in order to reduce the dependence on the training set. The correct classification of the tower (by adding a context knowledge given by the closeness of the tower and its components) would improve the performance of the automatic detection and analysis of the tower sub-components (i.e., insulators and hanging points), which will use a region-based segmentation and template matching.

Thermal data and images in the visible spectrum have been acquired by a drone flying at a distance of approximately 10 m from the power lines, with different cameras, near Parma in December, 2017. Data include also a small number of images containing common defects. These data have been used to test our methods for the detection of the infrastructure and the diagnosis of its status.

At this stage, two tasks have been implemented and partially tested on real data: i) classification of insulators as normal or rusty, using a convolutional neural network trained on our data; ii) detection of power lines by image processing applied to RGB images. In the following section we provide a description about the second task.

### IV. Methods

In this work, keeping in mind the typical linear characteristics of power lines, we applied Hough transform on visible images to identify power lines. Several methods based on Hough transform had been proposed in the past to identify power lines as in [8], [11]. The images were processed following the steps listed here:

i) Preprocessing to improve contrast in the image.

ii) Detection of edges by using Canny edge detector.

iii) Hough transform to detect all lines in the images.

iv) Extraction of power lines.

As explained before, we had analyzed images acquired by a camera mounted on the drone flying close to the electric power lines. By way of example, an image is shown in Fig. 3a. After the contrast enhancement, we applied the Canny edge detector to identify edges and remove unwanted objects from the background of interest area. Canny edge detection algorithm [24] consists of the following steps:

i) In order to smooth the image, Gaussian filtering is applied to reduce noise effects by convolving image with Gaussian filter.
Fig. 4. Detected peaks with Hough transform, where peaks correspond to the length of the line. $\rho$ is the perpendicular distance of the peak to the origin and $\theta$ correspond to the angle. Occurrence of all positive angled peaks correspond to power lines.

ii Image gradient magnitude and direction are computed

iii Non-maxima suppression, according to the gradient direction, to get unilateral edge response and to preserve local maxima as these maxima correspond to the edges. (The output of maxima suppression contains some local maxima which correspond to noise elements)

iv Double threshold method, in order to detect and connect edges.

The results obtained by using the Canny edge detector is shown in Fig. 3: power lines along with sharp edges of background were detected. The next step is to highlight only those edges which correspond to power lines. Hough transform is used to detect parameterized shapes through mapping each point to a new parameter space in which the location and orientation of certain shapes could be identified [25]. In this work we applied Hough transform to identify power lines, as the method identifies all straight lines in the image, maybe including roads, buildings etc. Therefore, in order to discriminate power lines from other linear object we applied clustering in the Hough space. The method usually parametrizes a line in the Cartesian coordinate to a point in the polar coordinate using the point-line duality equation:

$$x \cos \theta + y \sin \theta = \rho \quad \rho \geq 0 \quad 0 \leq \theta \leq \pi$$

(1)

Where $(x, y)$ is the point in image in Cartesian coordinates. $\rho$ is the perpendicular distance of the peak to the origin and $\theta$ correspond to the angle to the origin. Before detecting power lines in the Hough space, we applied the Canny edge detector to identify all edges in the images. Fig. 4 highlights the detected peaks: here we filtered the three close perpendicular $\theta$ peaks corresponding to power lines.

V. RESULTS AND ONGOING WORKS

We applied the method explained in the previous section to the video sequence acquired during the acquisition campaign. The method is applied on 44 frames of 2 mega pixel size and it took 117 seconds to process the complete video sequence. The detection obtained from a single frame is shown in Fig. 5. All detected lines are marked in the visible image; at the same time we generate the mask by segmenting power lines for better visualization and inspection. In future, we aim at improving the accuracy of the power line tracking by Kalman filtering [10], and make the processing faster (approximately real time) by integrating the prior knowledge of the drone motion. We had acquired simultaneously visible and infrared images with drones; in this work we used only visible image to detect power lines. The next step is to analyze IR images to identify defects and broken wires. In order to do so, a key point is to register the RGB and IR images, as cameras have different focal length and position. Therefore, in order to closely identify defected area it is of extreme importance to perform an efficient sensor fusion.

VI. CONCLUSIONS

In this paper, we have studied different inspection methods based on visible and infrared images designed to detect and inspect power transmission lines. Infrared imaging applied to power lines monitoring account for the differences of temperature at the joints; hence it is used for the fault diagnosis. On the other hand, image processing applied to RGB images and videos is able to provide quite an accurate localization of the inspected power lines, by applying standard computer vision techniques (e.g. denoising, filtering, segmentation, line detection). At first, We used vision based methods to identify power lines in visible images only. Canny edge detector has highlighted significant transition in the images and later by utilizing linear property of power lines, we applied Hough transform to identify power lines. We applied the described method on a short video sequence, and obtained encouraging
results. We plan to further extend the method to thermal images too, possibly improving the robustness of the method exploiting the registration of both data (i.e., thermal infrared and visible). The correct registration between data, together with a proper data fusion, is a key point to have not only an accurate detection and tracking of the power lines, but also to get a reliable and robust assessment of the maintenance status of the whole infrastructure.

VII. ACKNOWLEDGEMENT

This work is being carried out in the framework of the Tuscany Regional Project SCIADRO (FAR-FAS 2014).

REFERENCES


Big Data Application for Smart Features Formation in Medical Diagnostic Tasks

Nataly Ilyasova
Samara National Research University
IPSI RAS – Branch of the FSRC
“Crystallography and Photonics”
RAS
Samara, Russia
ilyasova.nata@gmail.com

Alexandr Kupriyanov
Samara National Research University
IPSI RAS – Branch of the FSRC
“Crystallography and Photonics”
RAS
Samara, Russia
alexkupr@gmail.com

Rustam Paringer
Samara National Research University
IPSI RAS – Branch of the FSRC
“Crystallography and Photonics”
RAS
Samara, Russia
rusparinger@gmail.com

Dmitriy Kirsh
Samara National Research University
IPSI RAS – Branch of the FSRC
“Crystallography and Photonics”
RAS
Samara, Russia
kirshdv@gmail.com

Abstract—The paper presents the main research results in the area of data mining application in medicine. We propose a single approach to effective features formation for the identification of regions of interest in an image. The technique is based on clustering methods with the use of texture features. The feature space is composed of features which are the best in terms of the discriminant analysis' separability criterion. The effectiveness of the set of features was estimated using a clustering procedure based on the K-means method. The Euclidean and Mahalanobis distances were utilized as similarity measures. Comparative experimental studies intended to identify informative texture features of different color subspaces were conducted. The pairwise feature selection approach has enabled the clustering error to be two times reduced and the number of features to be reduced more than twice. The use of Big Data technology in developed medical diagnostic systems has made it possible, due to involving more amounts of data, to improve the training sample and reduce classification errors that ensured an increase of diagnosis accuracy up to 95%.

Keywords—Big Data, medical diagnostics, data mining, feature formation, image segmentation

I. INTRODUCTION

The BigData mining for identification of relationships between changing diagnostic information on medical images with different kinds of diseases is a key problem of modern information technologies. Data Mining technique is used for identifying and analysis of relationships in arrays of semi-structured information and for build-up of models which describe behavior of complicated systems. Data Mining means the research and detection by a "mechanism" (algorithms, artificial intelligence tools) in raw data which were formerly unknown and are practically useful and acceptable for interpretation by a human being [1].

It moves beyond evidence-based medicine to personalized medicine involving a close integration of information technology, science, and clinical therapy to achieve the best clinical or preventive results. The problem of data-arrays-to-knowledge conversion is of prior focus for the world's leading scientists working in medicine, public health, and application informatics.

BigData has been characterized in terms of its volume, variety, velocity, veracity, and value [2,3]. A data volume is considered to be big when difficulties stem during its processing and storing by applying traditional methods which require new approaches and more refined tools. The internal reason for new data-processing techniques is the necessity to parallelize the processing and distribute it into a large number of independent data-processing flows. In regard to the variety, such data volumes are rarely homogeneous. In the vast majority of cases, the overall data array includes both structured and unstructured data. The velocity shows both an increasing data accumulation rate (90% of information has been gathered in the last two years) and a data processing rate - on-line data processing techniques have recently become more sought-after. Under big data work conditions, the particular significance is taken on by data veracity, separation of true data from information noise and junk information, and sifting of this noise and junk information [4]. There is the following thing in biomedicine: the larger sample size, the more accurate estimates. However, large sample sizes with poor-quality data may seriously disorient. In public health service, either accuracy or reliability qualities are equally important. Information value predetermines reasonability of its

Copyright © 2018 by CENPARMI, Concordia University
processing. Currently collected data should provide answers to preliminarily articulated and reappeared questions. The effects, resulting from data collection and processing, should justify the costs required for these operations. Currently collected data should bring positive results.

We offer to use a single approach to effective features formation for the identification of regions of interest (ROIs) in fundus images (Fig. 1) for further diagnostic analysis [5,6].

![Fig. 1. The example of a fundus diagnostic image without pathology (left) and an image with pathology detected (right).](image)

In order to identify the type of ROI based on the big data mining using methods of the discriminative analysis, a technique of efficient feature space formation has been developed [7-9]. The methods being developed are aimed at improvement of the medical diagnostics quality due to obtaining objective numerical estimates of biomimic image parameters using large volumes of accessible information.

II. INFORMATION TECHNOLOGY OF DIAGNOSTIC IMAGE DATA MINING

The main objective of currently conducted research at the Image Processing Systems Institute - Branch of the Federal Scientific Research Centre “Crystallography and Photonics” of Russian Academy of Sciences (RAS) under the leadership of the Academician of the RAS V.A. Soifer is the development of computer techniques for high-performance processing, analysis, and interpretation of medical and diagnostic images. The relevance of conducted research is also stipulated by the significance of early diagnostics, prediction of the course and selection of an optimal therapeutic approach to the treatment of human diseases. Late diagnostics or interpretation of changes often results in a significant decrease of treatment efficiency and disease prevention. Currently used methods of status accounting and formalized description do not always give an aggregate factor pattern required for proper diagnostics.

Research of diagnostic images is comprised of three stages of data processing given in Fig. 2: processing of biomedical signals, data analysis, and results visualization.

Big data mining of specified image classes performed at IPSI RAS to solve tasks of identification of regular changes in diagnostic information in medical images with various types of diseases includes the use of new mathematical methods and algorithms of distributed processing and recognition of biomedical images for remote diagnostic systems. A common approach has been proposed to the analysis of different classes of images based on evaluation of aggregate geometric and texture parameters of allocated regions of interest which are supposed to be a basic feature set for further diagnostic analysis.

The information technology of diagnostic image mining includes the following advanced techniques and algorithms:

- the technique and the algorithm of increasing a degree of the feature informativity based on the discriminative analysis and formation of an optimal learning sample to learn a disease diagnosis expert system;
- an estimation method of a class separability which is not influenced by distribution of objects through classes and is independent from a used classifier;
- the algorithm of decrease of feature space dimensions and formation of new informative features that maximize a separability criterion based on methods of the discriminative analysis and allow to increase a diagnosis accuracy of a pathology degree;
- a technique of optimum learning sample formation to learn a diagnostic system based on exemption of anomalous observations that will also enable to increase a disease diagnosis accuracy.

The efficient feature space formation technique has been developed to analyze diagnostic images based on big data mining of unstructured information using the methods of statistical analysis [10-15]. Informational feature analysis is performed by discriminative analysis using separability criteria that depends neither on distribution of objects per classes nor on the used classifier. This approach to generation of diagnostic features, which, unlike traditional abstract spectral-and-correlation features, are supposed to be customary and understandable for medical professionals, as well as clearly evident and having regard to specifics of the object, allows to eventually improve the value of diagnosis.

III. THE TECHNIQUE OF DEFINING RANGES OF INTEREST BASED ON THE TEXTURE ANALYSIS OF BIOMEDICAL IMAGES

The regions of interest were extracted by making the decision whether an image fragment belonged to one of the four above-described object classes: exudates, thick vessels, thin vessels, and intact areas. For fragmentation, the image was divided into square-shaped blocks, which were then classified based on the technique proposed in [5, 6].

The technique is based on clustering methods with the use of texture features. Analysis of the regions of interest (ROI) of the source images has shown their texture properties to be fairly different. It is worth noting that texture features were earlier shown to provide good results in biomedical image recognition and their subsequent use in diagnostics [5,6,7,8]. To calculate the texture features, we made use of a well-known library MaZda [16, 17]. Presented work is an attempt to improve the earlier proposed technique via forming a fragment sampling. Comparative experimental studies intended to identify informative texture features of different color subspaces were conducted. Thus, we formed five initial samplings of feature vectors, which were calculated in the images in different combinations. The technique for selecting effective features for recognition is presented in Fig. 3.
With the general approach to feature selection, the feature space is composed of features which are the best in terms of the discriminant analysis’ separability criterion \([10,11,14]\), which is calculated for all four classes of the objects under study. Alongside the general approach, we utilized a pairwise feature selection in which features characterized by the largest
separability criterion were selected, while separating each pair of classes and subsequently uniting them in an integrated set. To estimate the quality of the resulting sets, the clustering error was calculated for all fragmentation block sizes under study. Clustering was conducted using a K-means method, with the Euclidean and Mahalanobis distances used as similarity measures [18].

As we mentioned above, texture features have shown themselves to be well-suited for biomedical image recognition and subsequent diagnostics. There have been a large number of texture features. The software “MaZda” utilized in this work is able to calculate the following groups of texture features (see MaZda user manual, for further details):

- statistical characteristics (histogram-based features);
- gradient parameters of image;
- the co-occurrence matrix features;
- run-length matrix features.

In the discriminative analysis the class separability criteria are formed using scatter matrices inside classes and scatter matrixes between classes. The scatter matrix inside classes demonstrates a variety of objects with respect to mean vectors of classes: \( W = \sum_{k=1}^{g} \frac{1}{n_k} \sum_{i=1}^{n_k} (\mathbf{x}_i - \bar{x}_k)(\mathbf{x}_i - \bar{x}_k)' \), where \( k \) – is a class data corresponding to the mean vector \( \bar{x}_k = [x_{k1}, x_{k2}, \ldots, x_{km}] \), and \( g \) – is a total number of classes.

Elements of the scatter matrix between classes \( B \) is calculated by the formula:

\[
b_{ij} = \sum_{k=1}^{g} n_k (\bar{x}_i - \bar{x}_j)(\bar{x}_i - \bar{x}_j)' \]

where \( \bar{x}_i = (1/n) \sum_{k=1}^{g} n_k \bar{x}_k \) – is a mean feature value of \( i \)-feature in all classes, \( n_k \) – is a number of objects in \( k \)-class, \( \bar{x}_k = (1/n_k) \sum_{i=1}^{n_k} \bar{x}_{ik} \) – is the mean feature value in \( k \)-class, and \( x_{im} \) – is a value of \( i \)-feature for \( m \)-object in \( k \)-class. The matrices \( W \) and \( B \) contain all basic information about interrelationships inside and between classes. In order to obtain the class separability criterion some number is to be associated to these matrixes. This number should be increased with the class separability criterion some number is to be associated to these matrixes. This number should be increased with the

\( \text{SEPARABILITY CRITERION VS. A PARTICULAR FEATURE SELECTION TECHNIQUE} \)

**TABLE I. FEATURE SETS CHARACTERIZED BY THE MAXIMAL SEPARABILITY CRITERION VS. A PARTICULAR FEATURE SELECTION TECHNIQUE**

<table>
<thead>
<tr>
<th>Feature number</th>
<th>Comparison approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>General comparison</td>
</tr>
<tr>
<td>1</td>
<td>B Perc.99%</td>
</tr>
<tr>
<td>2</td>
<td>B Perc.99%</td>
</tr>
<tr>
<td>3</td>
<td>G S(1,0)Entrophy</td>
</tr>
<tr>
<td>4</td>
<td>G S(0,1)Entrophy</td>
</tr>
<tr>
<td>5</td>
<td>H Perc.99%</td>
</tr>
</tbody>
</table>

From Fig. 4, which illustrates the interrelationship of maximum, average and minimum clusterization errors depending on the type of features the regions of interest containing four different types of features: a) original features; b) original features; c) Euclidean distance and set of newly formed features; d) Mahalanobis distance and set of newly formed features. It is thereby recommended to use the fragmentation window 46 pixels in size. This shall provide at least 95% of identification certainty of ranges of interest.
V. CONCLUSION

In this work, we have proposed a technique for selecting effective features by the usage of BigData approach intended for fundus image clustering. The technique has enabled the intelligent feature analysis to be conducted when extracting the regions of interest containing four classes of objects: exudates, intact areas, thick vessels, and thin vessels. The effectiveness of the set of features was estimated using a clustering procedure based on the K-means method. The Euclidean and Mahalanobis distances were utilized as similarity measures. The analysis of informative value of obtained feature space and the selection of the most effective features is performed using the data discriminative analysis. The best result for the image segmentation with the feature sets providing the precise identification required for regions of interest are presented in Fig. 5.

The use of Big Data technology in developed medical diagnostic systems has made it possible, due to involving more amounts of data, to improve the training sample and reduce classification errors that ensured an increase of diagnosis accuracy up to 95%. The developed efficient feature-space generation technique used to analyze diagnostic images based on big data mining of unstructured information by applying the discriminative analysis methods and the methods proposed hereby, enables to improve the quality of medical diagnosis due to obtaining numerical objective estimations of biomedical image parameters using large volumes of arrays of available data. Further researches shall be aimed at the improvement of individual stages of the technology presented herein, particularly, shape modifications of the segmentation window, at the use of the image preprocessing procedure, which enables to focus on fundus image elements required for the analysis, and the development of an alternative feature selection method and the use of a more sophisticated clusterization algorithm. It also should be noted that the proposed technique has made it possible to extract the informative features. The follow-up study will focus on the improvement of separate stages of the proposed technique, in particular, the introduction of an image preprocessing procedure, which would allow fundus image elements essential in terms of analysis to be singled out.

VI. ACKNOWLEDGMENT

This work was partially supported by the Ministry of education and science of the Russian Federation in the framework of the implementation of the Program of increasing the competitiveness of Samara university among the world’s leading scientific and educational centers for 2013-2020 years; by the Russian Foundation for Basic Research grants (# 15-29-03823, # 15-29-07077, # 16-41-630761; # 16-29-11698, # 17-01-00972); by the ONIT RAS program # 6 “Bioinformatics, modern information technologies and mathematical methods in medicine” 2017, in the framework of the state task #0026-2018-0104 "Optoinformation technologies for obtaining and processing hyperspectral data".

VII. REFERENCES

A post-processing method for 3D fluorescence microscopy images

Andrey Nasonov, Yakov Pchelintsev, Alexandra Nasonova and Andrey Krylov

Laboratory of Mathematical Methods of Image Processing
Faculty of Computational Mathematics and Cybernetics
Lomonosov Moscow State University
Moscow, Russia
kryl@cs.msu.ru, nasonov@cs.msu.ru

Abstract—Three-dimensional (3D) deconvolution microscopy is very effective in improving the quality of fluorescence microscopy images. However, due to ill-posed nature of the deconvolution, many images still remain blurry after deblurring algorithms. We use an edge sharpening algorithm based on pixel grid warping to further improve the quality of blurry images in edge areas. Its main idea is to move pixels toward the nearest image edge to make them sharper without noise amplification. In this paper, we improve the results of recently developed 3D post-processing algorithm by considering the blur kernel that correspond to real optic blur and by optimizing the pixel displacement function. We illustrate its effectiveness on real data with modeled blur.

Index Terms—edge sharpening, image deblurring, grid warping, optical blur, 3D image sharpening

I. INTRODUCTION

Image blur occurs in numerous types of 2D and 3D images, e.g. photographs, medical images of different modalities, telescopes, microscopes, satellite sensors, etc. As a consequence, the deblurring problem (also called deconvolution) is widely investigated for simpler 2D case and then extended to 3D case. Deblurring methods require explicit knowledge or accurate estimation of the blurring kernel — Point Spread Function (PSF).

Image deblurring is a challenging ill-posed problem of finding a sharp image $I_0$ from the given blurred image $I_B$ using the blur model

$$I_B = I_0 * H + n.$$ 

If the blur kernel $H$ and noise $n$ are known exactly, the deconvolution problem can be effectively solved by regularization-based algorithms [1].

Typically, there is no or few information about $H$ and $n$. In that case, the blur kernel is to be estimated. There are some fairly powerful techniques for blind image deblurring [2], [3]. Non-uniformity of image blur, noise and blur kernel estimation errors may significantly degrade the result. It is not easy to find optimal parameters for a compromise between smooth result with blurry edges and sharp result with artifacts like ringing or noise amplification when blur kernel is estimated with errors.

Blurred image enhancement methods that are not based on the PSF concept can be referred to as image sharpening methods. A commonly used method of image sharpening is the unsharp masking method [4], [5]. The main problem of the existing sharpening methods is unwanted overshooting artifact and noise amplification that may appear in the output image [6].

The aim of the paper is to develop a post-processing method to enhance the results of existing image deblurring algorithms. We present an image sharpening method that performs the enhancement of a blurred image in the neighborhood of image edges. The idea is to transform the neighborhood of the blurred edge so that the neighboring pixels move closer to the edge, and then resample the image from the warped grid to the original uniform grid.

The warping approach is related to the morphology-based sharpening [7] and shock filters [8]–[10]. But these methods make the image appear piecewise constant which is effective mostly for cartoon-like images. The proposed method is applied to edges locally so the textures are preserved a priori. Also warping compresses the edge neighborhood at fixed rate and does not make the image piecewise constant.

The warping approach for image enhancement was initially introduced as the solution of a differential equation derived from the warping process constraints [11]. The solution of the equation is used to move the edge neighborhood closer to the edge, and the areas between edges are stretched. The method has several parameters, and the choice of optimal values for the best result is not easy. Due to the global nature of the method the resulting shapes of the edges are often distorted. In another work [12], the warping map is computed directly using the values of left and right derivatives. In both these methods [11], [12] the pixel shifts are proportional to the gradient values. It results in oversharpening of already sharp and high contrast edges and insufficient sharpening of blurry and low contrast edges. Both methods also introduce small local changes in the direction of edges and produce aliasing effect due to calculation of horizontal and vertical warping components separately.

Application of grid warping method for the enhancement of 2D image deblurring methods has been already considered [13], [14]. In this work, we improve the results of recently developed 3D post-processing algorithm [15] by considering the blur kernel that correspond to real optic blur and by...
optimizing the pixel displacement function.

II. VOLUME IMAGE BLUR MODEL

Although 2D PSF of a microscope can be approximated by a Gaussian kernel, 3D PSF does not have an accurate Gaussian approximation [16]. There is a variety of PSF models, but the most popular is the Gibson-Lanni model [17], [18]. This model takes into consideration the difference between the design conditions and experimental conditions, such as the object thickness and coverslips. An example of 3D microscopy PSF based on the Gibson-Lanni model is shown in Fig. 1.

![Fig. 1: An example of the 3D microscopy PSF (256x256x75) based on the Gibson-Lanni model.](image)

The idea of the proposed 3D image sharpening algorithm is to move pixels towards edge centerline [19]. Consider one-dimensional edge profile $g(x)$ centered at the point $x = 0$ (see Fig. 2), $d(x) — \text{displacement function and } h(x) = g(x + d(x)) — \text{warped edge profile.}$

![Fig. 2: The idea of edge sharpening using grid warping.](image)

Simple scaling $f(x) = kx$ will give sharper edge but it will also shrink the entire image. To make the edge sharper without changing image size, only the area near the edge center should be shrunk while the area outside the edge should be stretched proportionally.

The warped grid should remain monotonic (i.e. for any $x_1 < x_2$ new coordinates should be $x_1 + d(x_1) \leq x_2 + d(x_2)$), so the displacement function should match the following constraint:

$$d'(x) \geq -1.$$  \hspace{1cm} (1)

Another constraint localizes the area of warping effect near the edge center:

$$d(x) \rightarrow 0, \quad |x| \rightarrow \infty.$$  \hspace{1cm} (2)

The displacement function $d(x)$ greatly influences the result of the edge warping. On the one hand, the edge slope should become steeper. On the other hand, the area near the edge should not be stretched over some predefined limit to avoid wide gaps between adjacent pixels in the discrete case.

The work [19] constructs the displacement function $d(x)$ using the proximity function

$$p(x) = 1 + d'(x)$$

$$d(x) = \int_{-\infty}^{x} (p(y) - 1)dy.$$  \hspace{1cm} (3)

The proximity is the distance between adjacent pixels after image warping. This value is inverse to the density value. If the proximity function $p(x)$ is less than 1, then the image area is shrunk at the coordinate $x$. If the proximity is greater than 1, then the image is stretched. The identity transform has the value $p(x) \equiv 1$.

The constraint (1) leads to non-negativity of the proximity function. Also high values of the proximity function should be avoided to preserve image textures.

IV. VOLUMETRIC IMAGE SHARPENING

To apply the warping algorithm to 3D images, we generalize the two-dimensional extension [19] to the three-dimensional case. The displacement is a 3D vector field $\vec{d}(x, y, z)$ with the following constraints similar to the 2D case:

1) The shapes of the edges cannot be warped, so $\vec{d}(x_c, y_c, z_c) = 0$ for each edge point $(x_c, y_c, z_c)$.

2) There cannot be any turbulence: $\text{rot } \vec{d} = 0$. Since $\text{rot} \vec{d} \equiv 0$, the displacement field is assumed to be gradient of some scalar function $u(x, y, z)$: $d(x, y, z) = \nabla u(x, y, z)$.

3) The constraint (1) takes the form

$$\text{div } \vec{d} \geq -1$$ \hspace{1cm} (4)

and the proximity function is

$$p(x, y, z) = 1 + \text{div } \vec{d}(x, y, z).$$  \hspace{1cm} (5)

Since $\text{div } \nabla \equiv \Delta$, where $\Delta$ is a Laplacian, the warping problem is posed as a Dirichlet problem for the Poisson equation in the area of the image:

$$\begin{cases} 
\Delta u = p(x, y, z) - 1, \\
 u(x, y, z) = 0 \text{ at image borders}. 
\end{cases}$$ \hspace{1cm} (6)

The second constraint here is the boundary condition: the displacements at image borders should be equal to zero.

In order to get the same results as in the 1D case and to keep the edge pixels unwarped, the proximity value should be equal to the 1D proximity function depending on the distance to the edge. However, the distance to the closest edge $p$ as an argument of the proximity function $p(p)$ is not efficient as
it may produce gaps between close edges. Also it blurs edge ends.

We suggest the following method for calculating the proximity function:

\[
p(x_0, y_0, z_0) = \frac{\sum_{(x,y,z) \in E(x_0, y_0, z_0)} p(x_n) G_\sigma(x_t) |\vec{g}(x, y, z)|}{\sum_{(x,y,z) \in E(x_0, y_0, z_0)} |\vec{g}(x, y, z)|}
\]

(7)

where \(E(x_0, y_0, z_0)\) is the set of edge points in the neighborhood of \((x_0, y_0, z_0)\). The 3D edge point set is obtained using 3D Canny edge detector which is similar to 2D algorithm [20].

The value \(x_n\) is the projection and the value \(x_t\) is the length of the rejection of the vector \((x_0 - x, y_0 - y, z_0 - z)\) on the edge gradient vector \(\vec{g}(x, y, z)\).

The function \(p(x_n)\) is the 1D proximity function, weighting function \(G_\sigma(x_t)\) is Gauss function with standard deviation equal to the edge’s blur \(\sigma\).

We solve the partial differential equation (6) using Fourier transform technique.

V. PROXIMITY FUNCTION MODELS

Two models have been considered for the choice of the proximity function.

A. Difference of Gaussian functions

For volumetric Gaussian blur with parameter \(\sigma\), the difference of two Gaussian functions is used as a proximity function [15]:

\[
p_1(x) = 1 + \kappa (G_{\sigma_1}(x) - G_{\sigma_2}(x)), \quad \sigma_2 > \sigma_1, \\
\kappa = 1 / (G_{\sigma_1}(0) - G_{\sigma_2}(0)).
\]

(8)

It allows to control the areas of shrinkage and stretching independently [19]. Parameter \(\sigma_1\) controls the width of the densification area while parameter \(\sigma_2\) controls the width of the rarefaction area. We use \(\sigma_1 = \sigma\) and \(\sigma_2 = 2\sigma\). Fig. 3 illustrates grid warping for edge sharpening using this proximity function.

![Fig. 3: Edge sharpening by grid warping using the proximity function (8). Blue function is the blurred edge with some noise. Red function is the result of warping. Green line is the proximity function.](image)

B. Piecewise constant

In piecewise constant model (PWC), the area of densification with constant parameter is followed by the area of rarefaction with another constant parameter [21]:

\[
p_2(x)[a, b, c] = \begin{cases} 
1 + \frac{c}{a}, & |x| \leq a, \\
1 - \frac{c}{b-a}, & a < |x| \leq b, \\
1, & |x| > b.
\end{cases}
\]

The corresponding 1D displacement function looks as:

\[
d_2(x)[a, b, c] = \begin{cases} 
\frac{a}{a} x, & |x| \leq a, \\
\frac{b-|x|}{b-a} \text{sign} x, & a < |x| \leq b, \\
0, & |x| > b.
\end{cases}
\]

Despite the discontinuities, this model has greater sharpening effect than DoG model (8).

![Fig. 4: Piecewise linear displacement function.](image)

The parameters \(a, b\) and \(c\) define the width of densification and rarefaction areas and the steepness of the displacement function. The plot of this displacement function can be seen at Fig. 4. The strongest warping effect which meets the condition (1) is achieved when \(c = -a\). Therefore, we use the proximity function with two parameters:

\[
p_2(x)[a, b] = p_2(x)[a, b, -a] = \begin{cases} 
0, & |x| \leq a, \\
\frac{b}{b-a}, & a < |x| \leq b, \\
1, & |x| > b.
\end{cases}
\]

(9)

During the experiments we analyzed the choice of parameters \(a\) and \(b\) in order to maximize objective image quality. It was found that the ratio \(b/a\) of optimal parameters \(a\) and \(b\) was different for different images. At the same time, variations of the parameter \(b\) did not significantly influenced the image quality. To reduce the number of parameters at the cost of insignificant decrease of image quality, we set \(b = \frac{3}{2}a\):

\[
p_2(x)[a] = p_2(x)[a, \frac{3}{2}a, -a] = \begin{cases} 
0, & |x| \leq a, \\
3, & a < |x| \leq \frac{3}{2}a, \\
1, & |x| > \frac{3}{2}a.
\end{cases}
\]

(10)
VI. RESULTS

Test 3D images were generated using Pollen image, which is a thin optical section through the center of the desiccated stage of the mature pollen, collected from http://www.cellimagelibrary.org/images/35532. 16-bit reference image was convolved with the Gibson-Lanny blur kernel and corrupted by Poisson noise with different noise levels \( \lambda \). Then Richardson-Lucy algorithm [22], [23] with Total-Variation regularization [24] and 200 iterations was applied to restore the original image from the convolution result. ImageJ software with plugin DeconvolutionLab2 was used. The examples of images are at Fig. 5.

Then we applied image warping with two considered models as a post-processing algorithm after image deblurring. The results are shown in Table I and Figures 6, 7.

The execution time of image warping algorithm for \( 256 \times 256 \times 75 \) image is about 50 seconds for Intel Core i5 Skylake processor, and 5 minutes for \( 1024 \times 1024 \times 75 \) image. RL-TV algorithm from DeconvolutionLab2 plugin has taken 8 minutes and 10 hours respectively. Fast GPU implementation of the proposed algorithm is possible [25].

<table>
<thead>
<tr>
<th>Method</th>
<th>No noise</th>
<th>( \lambda = 15 )</th>
<th>( \lambda = 50 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blurred and noisy images</td>
<td>27.05</td>
<td>27.05</td>
<td>27.04</td>
</tr>
<tr>
<td>RL-TV [24]</td>
<td>32.16</td>
<td>32.08</td>
<td>29.99</td>
</tr>
<tr>
<td>RL-TV + DoG warping (8)</td>
<td>32.20</td>
<td>32.13</td>
<td>30.21</td>
</tr>
<tr>
<td>RL-TV + PWC warping (10)</td>
<td>32.30</td>
<td>32.24</td>
<td>30.37</td>
</tr>
</tbody>
</table>

TABLE I: PSNR values for test image deblurring with different noise levels and different warping proximity function.

VII. CONCLUSION

The proposed method sharpens 3D images in edge areas. It has the following advantages and features:

1. The best results are achieved when grid warping is used as a post-processing method after deconvolution-based image deblurring methods. Traditional image deblurring methods improve overall image quality while grid warping pays special attention to image edges.

2. No artifacts like ringing effect or noise amplification are introduced because pixel values are not changed.

3. Unlike morphological methods and shock filters, the resulting images look natural and do not inevitably become piecewise constant.
REFERENCES


Automatic Video Mining in Animal Behavior Study using Statistical Shape Models

Yaoguang Zhong
Dept of Computer Science
Northern Illinois University
DeKalb, IL, USA
yaoguangzhong@yahoo.com

Shawn B. Horvatic
Dept of Computer Science
Northern Illinois University
DeKalb, IL, USA
sbhorvatic@gmail.com

Takuya Kaneko
Life Sciences Institute, Dept of Cell & Developmental Biology
University of Michigan
Ann Arbor, MI, USA
tkaneko@umich.edu

Bing Ye
Life Sciences Institute, Dept of Cell and Developmental Biology
University of Michigan
Ann Arbor, MI, USA
bingye@umich.edu

Jie Zhou
Department of Computer Science
Northern Illinois University
DeKalb, IL, USA
jzhou@niu.edu

Abstract—Study of animal behavior is an important field in biology and neuroscience. While video recording has often been used to assist the research, most current studies still rely on manual observation which can be labor intensive and error-prone. Automatic analysis of images and videos for animal behavior can thus improve the state-of-the-art of related scientific research by reducing subjectiveness and promoting large scale study. Here we present a case study for model organism fruit fly larvae, where the behavior of the animal under nociceptive stimulation is automatically analyzed. In particular, we focus on the automatic separation of adjoining animals in a behavior recording. The method is based on a succinct shape representation via shape decomposition, feature alignment and feature reflection. Data reduction is then performed using Eigen analysis. For each animal in the video, a statistical shape model is constructed in order to separate it from adjoining animals efficiently and effectively based on candidate filtering. Once tracked and separated, the animals can then be individually analyzed for their behavior responses under external stimulation.

I. INTRODUCTION

Study of animal behavior has always been an important task in biology, neuroscience and psychology. It many cases, it is natural to use video to record and analyze the animal behavior, especially for small model organisms such as Drosophila m. (fruit fly). However, most such analyses still depend on manual observation of the recordings, which not only risks being error-prone and subjective, but also prevents large scale quantitative analysis in this era of unprecedentedly increasing amount of image and video data.

In this paper, we present an application of automatic video analysis for behavior study. Specifically, we automatically analyze the response behavior of fruitfly larvae under nociceptive (pain) stimulation. Fruit fly is an essential model organism. Although it seems to be a simple organism, there remain a lot of unknowns in fruit fly’s behaviors at their various life stages, and more so the underlying mechanisms of these behaviors – the study of which can thus potentially shed lights on many questions in neuroscience. For example, behavior analysis of fruit fly larvae under nociceptive stimulation provides insights into nociceptive neural circuits of how animals react to pain.

While automatic methods can provide efficient analysis and objective statistics about the animals of interest in the video, many computational challenges exist, especially in the presence of multiple individuals when they can adjoin and overlap in the recording. Efforts can be made at different stages of the application for separating, extracting and tracking of the animals. Here we focus on the algorithm for separation of adjoining animals while briefly covering other aspects.

Two special challenges need to be addressed in our application when separating the adjoining animals. First, fruitfly larvae in a behavior recording can change their poses dramatically under external stimulation (e.g., fast curling, rolling or extending). Second, noise and artifacts in the low- to-average quality video recording cause issues for locating and separating the animal(s). One example shown in Fig. 1d illustrates how image noise such as glare can deform an animal’s skeleton and lead to unexpected processing results. Because methods such as threshold-based segmentation or edge detection [4][5][6] can fall short in such cases while models that capture the statistical properties of the shape were seen to be more robust in the presence of similar noise [7][8][9][10], we choose to base our method on shape prior. Our task, in addition to handling noise in the recording, is to devise the approach that is effective with changing poses under stimuli.

In the remainder of this paper, we will discuss the procedures of our proposed method, demonstrate our experiments, and show the relevant results. Lastly, we will discuss the results.

II. METHODS

The purpose of our experiment is to separate the extracted and tracked animal (fruitfly larva) when it is in contact
The larvae in the video recordings are first detected and extracted using the following automated methods: We first detect the video sessions when the animal(s) are under the nociceptive stimulation. With our specific opto-genetics setup, it means that we are looking for video sessions when animals are under blue light. Then for each video frame in the session, we extract candidate objects using moving windows to look for the animal(s) of interest that will be tracked from frame to frame. Examples of cropped window with animal are shown in Fig. 3b and 4b. The cropped image then goes through several (standard and effective) image processing steps for denoising and binarization, specifically, rolling ball background subtraction, dynamic thresholding segmentation, opening operation, and then hole filling. Examples of the processed binary images are shown in Fig. 3c and 4c. Since the window may consist of multiple animals, the extracted object needs to go through the separation module, to be explained in detail in the rest of this section.

A. Procedures

Our method for separating adjoining animals consists of four procedures: shape decomposition, data reduction, statistical shape model construction, and probabilities of test instances calculation. Fig. 2 shows the flow of the method.

B. Shape Decomposition

1) Data Representing the Shape: Our method decomposes a larva shape in a few steps. The method first retrieves the object’s skeleton by carrying out a thinning operation on the binary image. The result of this operation is shown as Ske. 1 in Fig. 3d.
It then uses landmark points from the skeleton to approximate the shape of the object. We select nine of the skeleton points to represent the pose. We do so for two reasons. First, we want a number that is a power of 2 because we virtually cut the skeleton \( n \) times when we repeatedly divide the array of skeleton pixels into two halves and pick the middle element as the landmark point, and we would like to avoid rounding error. Second, the spacing between two selected adjacent skeleton points is critical for comparing corresponding points of two different larvae. Too little or too much space between two adjacent selected points can make the method compare irrelevant landmark points of two larvae. Selecting 9 points turns out a good choice heuristically (as opposed to 5, 17 or 33). Larvae have narrower ends and wider waists. They are rotation invariant when we repeatedly divide the array of skeleton pixels into two halves and pick the middle element because we virtually cut the skeleton \( n \) times when we repeatedly divide the array of skeleton pixels into two halves and pick the middle element as the landmark point, and we would like to avoid rounding error.

2) Point Alignment: We try to minimize the distances between the corresponding skeleton points of two larvae. To compare the equivalent points of two larvae, the method aligns the skeleton points of each larva. The center of the larva’s skeleton is translated to the origin of the coordinate. Suppose the translation vector is \( t \). All the larva’s skeleton points are rotated by an angle of \( \theta \) clockwise around the origin so that the end-point line (the line passes both end points of the skeleton) is parallel to the y-axis. Eq. 2 is used to align skeleton point \( S_0 \) to its new coordinate \( S_1 \). In Fig. 3d, Ske. 2 is the skeleton after translation, while Ske. 3 is the skeleton after transformation.

\[
S_1 = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix} (S_0 + t)
\] (2)

3) Points Reflection: After completing translation and transformation processes, skeleton points can fall in three places: to the left of the y-axis, to the right of the y-axis, or on the y-axis. We arrive at a simple algorithm for comparing two larvae’s skeleton points. Our goal is to minimize the distance between the corresponding skeleton points of two larvae but maintain their poses if possible. If more points fall to the left of the y-axis, the sign of the y coordinate of all skeleton points will be reversed, i.e., point \((x, y)\) is reflected to \((-x, y)\). The skeleton after reflection is shown as Ske. 4 in Fig. 3d.

4) Data Reduction: We assume that the dimensions of data with most variability encode the most important information about a larva’s shape. We apply Principal Component Analysis (PCA) to reduce dimensions of data. The larva’s 9 skeleton points and thickness are projected to a lower-dimensional space, the eigenspace. The eigenvector and the eigenvalue are calculated using Eq. 3.

The process keeps \( k \) dimensions of data that account for 99.5% of the overall variation of the data in the eigenspace, i.e., the first \( k \) principal components of PCA. In our experiment, the first 9 principal components already cover 99.5% of the overall variation.

\[
S_x v_i = \lambda_i v_i, i = 1, 2, 3, ..., n
\] (3)

In Eq. 3, \( S_x \) is the mean-adjusted covariance matrix; \( \lambda_i \) is the \( i^{th} \) eigenvalue; \( v_i \) is the eigenvector corresponding to the \( i^{th} \) eigenvalue; and \( n \) is the total number of principal components of PCA.

\[
x = \mu + \sum_{b=1}^{k} v_b w_b, b = 1, 2, 3, ..., k
\] (4)

Eq. 4 is used to restore a larva from the eigenspace to its original feature space. A larva can be reproduced by adding together the average larva and proportions of the eigenvectors. In Eq. 4, \( x \) is the restored larva; \( \mu \) is the mean larva; \( v_b \) represents the eigenvector corresponding to the \( b^{th} \) largest eigenvalue; and \( w_b \) is the weight corresponding to eigenvector \( v_b \).

The shape of a larva in the eigenspace is represented as follows:

\[
[w_1 \ w_2 \ w_3 \ ... \ w_i \ ... \ w_k]
\] (5)

where \( w_i \) refers to the weight of the eigenvector corresponding to the \( i^{th} \) largest eigenvalue.
5) Statistical Shape Model Construction: We observed that the weights of the eigenvectors are in Gaussian distribution. Hence, a larva’s probability can be calculated using the probability density function. Eq. 6 is a probability density function derived from [11],

\[
p(w) = \frac{1}{\sqrt{(2\pi)^k |\Sigma_k|}} e^{-\frac{1}{2} w^T \Sigma_k^{-1} w}
\]  

Where \(w\) is a matrix of the weights and \(\Sigma\) is the diagonal matrix of corresponding eigenvalues; \(\Sigma_k\) is the first \(k\) rows and first \(k\) columns of the diagonal matrix.

In Eq. 6, \(\frac{1}{\sqrt{(2\pi)^k |\Sigma_k|}}\) is a constant. Supposing \(c = \frac{1}{\sqrt{(2\pi)^k |\Sigma_k|}}\), Eq. 6 can be written as Eq. 7.

\[
p(w) = ce^{-\frac{1}{2} w^T \Sigma_k^{-1} w}
\]  

III. EXPERIMENTS

A. Dataset

To report the performance of our method, we used 543 images extracted from the behavior recording. An example is shown in Fig. 4.

The images are extracted from the frames using methods as described in Section 2. Afterwards, the extracted objects’ body area and skeleton length are examined. The objects whose body areas are less than a threshold (0.6 times the average larva’s body area) or with a skeleton length less than a threshold (0.6 times of the average larva’s length) were removed from the data set. If the extracted and tracked object contains one isolated animal, it is put in the training set. Otherwise, it is added to the test set.

The training set is made up of 507 images where the tracked larva is isolated. Processing is done as demonstrated in Fig. 4 and described in Section 2. The shapes of the training larvae are represented by a 507 × 19 matrix in the original feature space.

The test set is composed of 34 images where the tracked larva is in contact with other larvae. An example of test image is shown in Fig. 5a. When converting the image to a binary image through the process explained in Section 2, the tracked larva and the larva it touches form a single object as shown in Fig. 5b.

B. Candidate Filtering and Results

We define a candidate larva as one or multiple segments that can form a larva in an image. Also, we define a candidate case as a case that divides the segments into candidate larvae.

Specifically, the extracted object’s shape is divided into parts by Watershed segmentation of the Euclidian Distance Map as shown in Fig. 5c. The graph representation of the image is shown in Fig. 5d. A segment in the image is represented by a node, and the connection between two segments is represented by an edge. The method represents the segments in the image with an undirected connected graph \(G = (V, E)\). \(V\) is a set of nodes, while \(E\) is a set of edges connecting the nodes. In Fig. 5d, nodes 1 and 2 can form a candidate larva, and nodes 3 and 4 can form another candidate larva. A candidate larva is essentially a sub-graph of the graph. A candidate case is a way to divide the graph into sub-graphs. Table 1 shows all the candidate cases along with the candidate larvae. In the table, \((1, 2)\) represents the candidate larva made up of nodes 1 and 2.

A candidate larva’s probability can be calculated using Eq. 7. The probability of a candidate case is the sum of all candidate larvae’s probabilities in the case. Fig. 6 shows the probabilities of all the candidate cases. In lieu of including the part of the equation that remains constant, we use \(c\) to symbolize this formula to simplify comparison and save computation.

Table 2 lists the probabilities of all the candidate cases and larvae. Among all the candidate cases, candidate case 2 has the highest probability. As the case with the highest probability, candidate case 2 is chosen as the solution for the separation case by the proposed method.

In our experiments, 94.1% of the test instances were sepa-
TABLE II

<table>
<thead>
<tr>
<th>Case</th>
<th>Larva 1</th>
<th>Larva 2</th>
<th>Larva 3</th>
<th>Larva 4</th>
<th>Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.9c * 10^{-22}</td>
<td>5.5c * 10^{-51}</td>
<td>2.9c * 10^{-22}</td>
<td>2.9c * 10^{-22}</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.2c * 10^{-6}</td>
<td>7.2c * 10^{-6}</td>
<td>8.4c * 10^{-6}</td>
<td>1.4c * 10^{-15}</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3.0c * 10^{-31}</td>
<td>1.4c * 10^{-15}</td>
<td>5.5c * 10^{-51}</td>
<td>1.2c * 10^{-6}</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1.2c * 10^{-6}</td>
<td>7.8c * 10^{-11}</td>
<td>7.2c * 10^{-6}</td>
<td>1.4c * 10^{-14}</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3.0c * 10^{-31}</td>
<td>1.4c * 10^{-14}</td>
<td>5.5c * 10^{-51}</td>
<td>1.2c * 10^{-6}</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>3.0c * 10^{-31}</td>
<td>9.0c * 10^{-33}</td>
<td>7.2c * 10^{-6}</td>
<td>7.2c * 10^{-6}</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>3.0c * 10^{-31}</td>
<td>9.0c * 10^{-33}</td>
<td>7.8c * 10^{-11}</td>
<td>7.8c * 10^{-11}</td>
<td></td>
</tr>
</tbody>
</table>

One advantage of this method is that it can separate adjoining larvae correctly in images with noise. Fig. 1e is an example of an image containing noise in which our method is able to separate the adjoining larvae. Fig. 5 also shows how the proposed method overcomes the over-segmentation of Watershed segmentation of the Euclidian Distance Map. Another advantage of the method is that it obtains knowledge about the the animal’s shapes under study. Such knowledge is learned before the separation process takes place. Such statistical shape knowledge is also adaptive when the animal’s shape varies due to different experimental conditions. For example, fruit fly larvae tend to curl under nociceptive stimulation. The proposed model captures such shapes when separating adjoining animals.

Among the test cases, 94.1% were separated correctly using statistical shape models alone, which demonstrates the effectiveness of the proposed method. A few reasons contribute to the incorrectly separated larvae. First, one thickness index may be insufficient for “portraying” the shape of the animal. Second, the training set likely does not capture all the characteristics of the shapes under external stimuli. This can affect the calculated probabilities of the test instances. The accuracy rate can be further increased when combining with other validation strategies and making the training instances more representative and comprehensive.

V. Conclusions

In this paper, we demonstrated an application of automatic video mining in study of animal behavior recording and we focused on separation of adjoining animals in the presence of multiple fruit fly larvae to enable the tracking of individual animals. Once automatically tracked and separated, the animals can then be individually analyzed for their behavior responses under external stimulation. The method has the potential of leading to behavior study of large population for fruit fly larvae or similar model organisms.

The particular method we used comprises two key parts: data reduction and statistical shape distribution construction. The data reduction allows the method to exploit the linear formulation of the larva’s shapes and reduces dimensions of data where the data vary very little. The statistical shape model provides a way to learn variation of larva’s shapes. With the trained model, the method can evaluate likelihood of a shape representation being a larva. We have demonstrated the ability of this method to isolate adjoining larvae with the constructed model. We have also described the capability of the method to separate larvae in images with the presence of image noise. This approach may be applied to a range of uses for similar object separation in video recording.

REFERENCES

Separation of arterial and venous vessels images from computer tomography

Alexei Baev  
RTMBS department  
VSTU  
Yoshkar-Ola, Russia  
BaevAA@volgatech.com

Vasilii Dubrovin  
RTMBS department  
VSTU  
Yoshkar-Ola, Russia  
DubrovinVN@volgatech.com

Alexei Rozhentcov  
RTMBS department  
VSTU  
Yoshkar-Ola, Russia  
RozhentcovAA@volgatech.com

Anna Eruslanova  
RTMBS department  
VSTU  
Yoshkar-Ola, Russia  
yanaavelahim@mail.ru

Abstract— The method of vessel selection on tomographic images based on tracking is considered. The method of preliminary image processing is shown. The algorithm of localization and selection of tomographic images of vessels is synthesized.

Keywords— segmentation, vessels, tracking, computed tomography, image processing, anisotropic diffusion

I. INTRODUCTION (HEADING I)

Any quantitative computerized tools of image processing of human arterial and venous vessels requires the localization and separation of arteries and veins on their images. Localization is performed during the segmentation process. Segmentation can be carried out by specialists in manual mode directly on the image or using automatic or semi-automated methods. The problem with manual segmentation of arterial and venous vessels is that it depends on the professional skills. The task is quite laborious. Thus is considered an expensive way of localizing vessels in images. This causes the urgency of the problem being solved, there is a need for reliable automatic methods for segmentation of arterial and venous vessels.

Today there are many different approaches and algorithms for automatic segmentation of different types of vessels in the human body. The low contrast on computer tomography (CT) images, their proximity to other larger vessels, the fact that they are very thin - objectivities which makes segmenting arteries and veins task to be a difficult task. The existing segmentation algorithms require additional adjustment and adjustment.

During the last decade, many different methods for coronary artery segmentation have been presented [3]. They can be divided into two main categories: processing initial data to improve the contrast of vessel images and intelligent vascular segmentation techniques.

Methods based on the processing of raw data tend to improve the contrast of vessel images in a variety of ways. For example, by subtracting background structures [4]. Another category, the methods of vessel segmentation are based on pattern recognition, tracking and dissemination of mathematical models [5,6,7], neural networks [10], methods based on artificial intelligence, etc. [8,9].

II. MATH MODELLING

A. Math model of vessels system

Blood vessels - elastic tubular formations in the body of animals and humans, through which the force of the rhythmically contracting heart or pulsating vessel carries the movement of blood through the body: to organs and tissues along arteries, arterioles, capillaries, and from them to the heart - through the veins and veins. Arterial and venous vessels are the largest elements of the human circulatory system, with a diameter of more than 0.1 mm.

At the same time, arterial and venous vessels can be described as flexible vessels that taper along their length and become stiffer with a decrease in radius. They are organized in the form of a binary forked tree, an example of which is shown in Figure 1.a, b, in which the cross-sectional area of the vessel expands from about 5 cm² in the aortic root to about 400 cm² in the arterioles. The expansion occurs because, in each bifurcation, the combined cross-sectional area of the child vessel is larger than the parent vessel, even though the cross-
sectional area of each child vessel is smaller than the region of the parent vessel [1].

Consequently, a mathematical model of the system of blood vessels is a binary tree.

A tree is a hierarchical data structure used to facilitate access and sorting of data. A tree node is an instance of one of the two types of graph elements corresponding to an object of some fixed nature. A node can contain a value, state, or view of a particular information structure or the tree itself.

Each tree node has zero or more descendant nodes that are located down the tree (by convention, the trees grow 'down' rather than up, as is the case with real trees). A node that has a child is called a parent node relative to its child (or a precursor node, or a parent node). Each node has no more than one ancestor. The height of the node is the maximum length of the downward path from this node to the lowest node (edge node), called the sheet. The height of the root node is equal to the height of the whole tree. The depth of the nesting node is equal to the path length to the root node.

The mathematical model of the blood vessel system, \( V = \text{Tree}(B, V, L) \), is a binary tree \( T \), whose vertices are the branches of the vessels \( B \), the branches of the tree are the vessels \( V \), the leaves of the tree \( L \) are the arterioles, capillaries and venules.

B. Math model of single vessel

The arterial and venous vessel is characterized by a number of properties:

- the contents of the vessel have a relatively constant physical and chemical composition, as a result of which it has a practically constant contrast, in terms of computed tomography;
- the vessel narrows and curves rather smoothly;
- the vessel has a section in the form of a circle.

Based on the observation data, the geometrical model of the vessel section can be represented in the form of an axisymmetric cylinder, (see Figure 2) with the axis:

\[ \mathbf{v} = (v_x; v_y; v_z) \]  

The restriction along its length has an exponential law [1]

\[ r_0(x) = r_e \frac{\log \left( \frac{h}{s} \right)}{L} \]  

where the radii of the initial and final portion of the vessel, \( L \), the total length of the vessel.

III. LOCALISATION AND SEPARATION

The separation of the vessel on the CT image can be divided into 2 subtasks. Firstly, it is the detection and localization on CT of images of fragments potentially representing the vessel. Secondly, the calculation of the measure of the similarity of the image of the vessel with its reference model and its separation.

Since the image contains noise, direct search for arterial and venous vessels is considered a difficult task. To facilitate the process of vascular isolation, techniques such as digital subtraction angiography and image processing are used to remove noise.

A. The digital subtraction angiography

The digital subtraction angiography (DSA) is a contrast study of blood vessels with subsequent computer processing. It allows you to obtain pictures with the isolation of individual vessels from the general picture, while the amount of contrast medium administered can be reduced and this substance can be administered intravenously without resorting to catheterization of the artery, which is less traumatic for the patient. The method is based on subtracting the contrastless X-ray image \( I_{\text{abdominal}} \) from the contrast image \( I_{\text{contrast}} \). This is allows eliminating signals from the bones and improves the clarity of the vessel image \( I' \).

\[ I' = I_{\text{contrast}} - I_{\text{abdominal}} \]  

Figure 3 shows images of a non-contrast image (a), a contrast image (b), and a subtraction result (c). After performing smoothing operations and threshold binarization, images of the form (d).

![Fig 2. a) Vessel geometric math model](image)

![Fig 3. a) A CT image of the low-contrast phase, b) a CT image of the contrast phase, c) a result of the cleaning, d) after applying the threshold binarization](image)
B. Noise reduction

Images of organs of the retroperitoneal space on CT images obtained as a result of the examination have a variable intensity. This is caused by the heterogeneity of the structure of tissues and organs, most of the tissues contain blood, which leads to "noisy" images. The random element is also embedded in the very nature of the radiation source. Thus, the mathematical model of the CT image is as follows:

\[ I = b \cdot J + Z, \]  
(6)

where \( J \) - the undistorted ("ideal") image, \( b \) - the component associated with the heterogeneity of the image, \( Z \) - additive noise, \( I \) - the resulting image produced by the tomography. We assume that within a single tissue the component varies monotonically and smoothly, and that the noise component obeys the normal distribution law.

Traditional methods of removing noise, improving the edges and reconstructing blurry images rely on methods of high-frequency filtering or back propagation of the diffusion equation in time. The problem in the current formulation can be solved using as a physical basis a nonlinear diffusion process. P. Perona and J. Malik described [12] that if the conductivity factor is selected in accordance with the image gradient function, the diffusion will simultaneously smooth out the image and strengthen the edges, stabilizing the image to a stepped view. The result is guaranteed by the principle of maximum diffusion equation.

The anisotropic diffusion filter works as follows. On the input filter are images of slices of tomograms, scale coefficients of the CT image describing the physical distances between the pixels, and taken into account in calculating the coefficients of the CT image describing the physical distances in 

\[ \| \nabla I(u) \| \cdot \exp\left( -\frac{\| \nabla I(u) \|}{\kappa} \right). \]  
(7)

If the diffusion \( \Delta I(u) \) in a certain fixed vicinity of the point is large, then this intensity at this point is noise and the intensity value is corrected towards the dominant intensity. If \( \kappa \) is large, then this intensity at this point is noise and the value is corrected towards the dominant intensity. If \( \kappa \) is small, then the point belongs to the boundary and the smearing does not occur. For a more detailed description of the mechanism of filter operation see P. Perona and J. Malik. Among the advantages of the filter is stability and parallelism.

A gradient \( \nabla I(u) \) is estimated for a 3x3x3 neighborhood of a point \( u \). The number of iteration steps is 16. The choice of the parameter responsible \( \nu \) for the smoothness of the diffusion process is considered in the article G. Gerig et al [13]. In this paper we use the value \( \nu = 0.142857 \). The parameter \( \kappa \) has the same role as the root-mean-square deviation in the model of the normal distribution law, and affects the smoothing process. The higher the value of the parameter \( \kappa \), the greater the intensity range is smoothed. The working range of the parameter values \( \kappa \) for CT images of organs of the retroperitoneal space was experimentally established: \( \kappa = 25 \pm 5 \).

Figure 4 shows the result of an anisotropic diffusion filter.

Since the anisotropic diffusion filter is used in the previous step, which minimizes the scatter of values in the homogeneous region, but preserves the boundaries, the function of the form of the boundary detection function can be used:

\[ g(u) = \frac{1}{1 + \sigma^2[I(u)]}. \]  
(8)

where \( \sigma^2[I(u)] \) – is the variance of the brightness values in the region of the pixel radius \( r = 1.5 \) around the point. This type of function detects boundaries with sufficient sensitivity to weak brightness differences.

C. Contour extracting and circle extraction

The processed image undergoes binarization. Then the area of interest is segmented in by a closed loop. Proceeding from the rules of carrying out the operation of selecting the contour of the vessel, the desired contour must have the form of a circle of some measure value.

Since the cross-section of the vessel has the shape of a circle, the main criterion is a measure of the similarity with the model of an ideal circle of radius \( R \) at the point \( (x_0, y_0) \).

\[ (x-x_0)^2 + (y-y_0)^2 = R^2. \]  
(9)

The similarity measure is calculated as follows:

1. The contour center is calculated:

\[ c = \frac{1}{N} \sum_{i=1}^{N} C_i, \]  
(10)

where \( C_i \) - the points of the selected contour on the image.

2. Calculates the set of distances \( R \) of contour \( C_i \) points to the center \( c \) of the figure, the mathematical expectation \( M[R] \) of the distance, the dispersion \( D[R] \) of distances:

\[ R = \{|r_i| : r_i = |C_i - c| \}, \]  
(11)

\[ M[R] = \frac{1}{N} \sum_{i=1}^{N} |r_i|, \]  
(12)

\[ D[R] = M[(R - M[R])^2]. \]  
(13)

3. The value of variance \( D[R] \) is a measure of similarity.
In Figure 5, the possible minimum value of the dispersion is shown for different values of the radius of the circle in the range \([1:100]\) pixels. It is seen from the graph that for CT images the value does not exceed 0.07.

On the actual image, contour distortion occurs due to segmentation errors, the presence of background noise on the CT, etc. Also, the vessel is not always perpendicular to the plane of the cut, the vessel is at an angle. In this case, as will be shown below, the image is projected onto the plane passing through the current point of the region of interest and having a normal along the guide vessel. In this case, a projection error occurs, so that the circle becomes an ellipsoid:

\[
\frac{x^2}{a^2} + \frac{y^2}{b^2} = \frac{x^2}{k^2a^2} = 1, \quad b = ka. \tag{14}
\]

In Figure 6, the dependence of the dispersion on the compression coefficient (ellipticity) \(k\) is shown. The graph was constructed for a fixed value of \(R = 10\), the coefficient varied in range \([1:3]\). The deviation of the mathematical expectation \(M[R]\) for each experiment due to the accumulated sampling error did not exceed 1.5.

By analyzing the graph, the following criterion for the contour similarity with the contour of the ideal circle model is established:

\[
f(\{C_i\}) = \arg \min \{c_i, M[r_i], D[r_{ij}] > D[r_j] < 2, \tag{15}\]

where \(\{C_i\}\) is the set of obtained contours.

The choice of value 2 is justified by the maximum permissible deviation of the vessel shape from the ideal circle, taking into account the effect of segmentation interference.

**D. Vessel tracing**

The task of tracking the channel of the vessel is solved from the positions of secondary processing of radar information, consisting in determining the parameters of the trajectory of the target along one or a number of previous local marks [2].

The process of secondary processing is divided into two separate stages: detection of trajectories and tracking trajectories. Automatic trajectory detection is the initial point of secondary processing. Let there was a single mark \(\#1\) from the goal. It is taken as the initial mark of the trajectory. In the next review, the second mark belonging to the same trajectory should be sought in a certain region \(S_1\) (Figure 7) enclosed within the ring and having an area

\[
S_1 = \pi (R_{max}^2 - R_{min}^2). \tag{16}
\]

In the region \(S_1\) can fall several marks and each of them should be considered as a possible continuation of the proposed trajectory. On two marks the speed and direction of each of the intended targets are calculated, and then the position of the marker is extrapolated to the next review. Around the extrapolated marks, circular areas \(S_2\) are formed. If a mark falls in any area of \(S_2\) in the third survey, it is considered to belong to the detected trajectory, the trajectory continues, and the mark is transmitted for escort.

The operations that are performed during autodiscover are reduced to extrapolating coordinates, smoothing them, and strobing the marks. The extrapolation process consists in calculating the coordinates of the future mark on the coordinates of the previously obtained marks. Extrapolation requires knowledge of the laws of the movement of the target, on the basis of which the trajectory is laid. When a trajectory of a maneuvering target is detected, the size of the gates must be calculated taking into account the possible maneuver. The size of the strobe directly affects the quality of the trajectory detection. Its increase leads to an increase in the number of false marks in the gate. Reducing the size of the strobe can lead to a loss of true mark in the strobe.

The length of each step is determined by the current radius of the vessel \(r\). The predicted spatial position of the center of the vessel is located at a distance of \(1.5r\) from the current spatial position \(x_i\). The size of the gate is assumed to be equal to an angle of \(\pi/3\) from the vessel direction vector \(v\).
where \( x_t \) - the current position of the center of the vessel, \( x_{t-1} \) - the previous position of the center of the vessel.

The next point is selected as follows:

- for each of the projections of the variants, a plane passing through the point and having a normal is formed;
- a cut of the CT image along the plane is formed;
- a contour of the vessel is allocated on the cut image;
- for each contour, the coordinate of the center of the vessel and the measure of similarity with the true circle of radius \( r \) are calculated. Then, using the maximum likelihood principle the next point of the center of the vessel is selected;
- the algorithm completes its work when branching is detected, as the criterion for a smooth change in the diameter of the vessel will be violated.

The algorithm is finite for the following reasons:

- under natural conditions it can not be looped, since the vessel has a beginning and an end, caused either by the boundaries of the CT image, or by the branching points of the vessel;
- the trace of the vessel can not jump to an adjacent vessel, since the search step is commensurable with the radius of the vessel.

The construction of the shear image along the plane is performed using trilinear interpolation at the intersection points of the CT slices and the plane.

E. Branch processing

As described previously, the vascular system has the form of a tree and it contains branchings. Branching is binary, i.e. the vessel bifurcates. In this case, the branching point is usually thicker than the incoming outgoing vessels. Adequate treatment of branches, their detection is a necessary stage of vascular tracking.

The consequence of branching is the formation of two independent vessels with different orientation and thickness. A sign signaling the possible branching is the presence of two potential centers of the vessel, in which the criterion of similarity with the image of an ideal circle is satisfied. At the same time, it is possible that two vessels are very close together.

Let the two candidates have the coordinates \( p_1 \) and \( p_2 \). Two distances \( d_1 \) and \( d_2 \) are calculated, the first is the projection of the point with the coordinates \( p_1 \) on the vector drawn from the point \( p_2 \) with the normal \( d_2 \). Similarly, the distance \( d_2 \) is calculated as the length of the perpendicular dropped from the point \( p_2 \) to the vector drawn from the point \( p_1 \) with the normal \( d_1 \).

If one of the distances \( d_1 \) and \( d_2 \) is less than the threshold value \( \Delta r \),

\[
(d_1 - \Delta r) \cdot (d_2 - \Delta r) < 0 ,
\]

where \( \Delta r \) is the current radius of the vessel, then it is considered that the branch point is detected.

Such a criterion makes it possible to circumvent the situation when two closely located vessels are found moving in approximately the same direction.

In the case of branching, each subtree is constructed independently.

F. 3D model building

In the simplest case, the segmentation result is represented as a binary CT image, where the value 0 corresponds to the environment, and to the value 1 - the segmented object. A binary CT image is formed as a result of applying the decision functional immediately for the entire area of the CT image. Since the CT image is discretizable across all three coordinate axes, the Marching cubes algorithm is used to construct the triangulated 3D model [11]. The result of the algorithm is a regular isosurface.

IV. The results

Virtual 3d model was formed using the original software “Volga-M” (cert. № 2015660462), based on the mathematical models of the structural elements of the body we accepted a law of probability distribution of the brightness and dispersions values. When using augmented reality technology, we combined the 3D model image with the video of kidney tumor.

The Figure 8 shows the image of the selected vessel.

Fig. 8. 3D image of vessel and kidney

Using a virtual 3D model, augmented reality helps to emphasize the contours of the body, the boundaries of the pathological process, allows us to see the internal structures in the "transparent" mode, which is especially valuable in partial nephrectomy. The separation of arterial and venous vessels, the construction of their model allows the surgeon to detect traumatic directions of surgical intervention and conduct a safe operation planning. Preliminary results of our clinical studies have shown the significance of 3D modeling to improve visualization of the affected organ during surgery for the surgeon and for the understanding of the nature of the pathological process of the patient.
ACKNOWLEDGMENT

The work is executed at financial support of the Ministry of education and science of Russian Federation, project RFMEFI577170254 "System intraoperative navigation technology to support augmented reality-based virtual 3D models of organs obtained from the results of CT diagnostics, minimally invasive surgeries".

REFERENCES


On Conditionality of Pairwise Comparisons in Machine Learning

Sergey Dvoenko
Institute of Applied Mathematics and Computer Sciences
Tula State University
Tula, Russian Federation
dsd@tsu.tula.ru

Denis Pshenichny
Institute of Applied Mathematics and Computer Sciences
Tula State University
Tula, Russian Federation
denis.pshenichny@yandex.ru

Abstract—Recently, experimental data in machine learning often appear as direct comparisons between objects. Different ways to evaluate difference or similarity of a pair of objects in image and data mining, image analysis, bioinformatics, etc., are usually used in practice. Nevertheless, such comparisons often are not distances or correlations (scalar products) as a correct function defined on a limited set of elements in machine learning. This problem is denoted as metric violations in ill-posed matrices. Therefore, it needs to recover violated metrics and provide optimal conditionality of corresponding matrices of pairwise comparisons for distances and similarities. This is the correct basis for using of modern machine learning algorithms.

Keywords—metrics, similarity, dissimilarity, distance, scalar product, condition number, determinant, principal minor, eigenvalue

I. INTRODUCTION

Let experimental data be presented by pairwise similarities between elements from a limited set as a square matrix. It is well known, that the positively definite square matrix has the positive determinant and the corresponding set of positive eigenvalues [1]. In this case, elements can be presented in some multidimensional coordinate space by vectors, where scalar products and distances based on the cosine theorem can be evaluated between them. For normalized scalar products, vector tips are arranged on the hypersphere of the unit radius. If all scalar products are positive, then all vectors are placed in the positive quadrant of a coordinate space.

In practical sense, it is very important that experimental data can be placed in some real or imaginary (for pairwise comparisons only) feature space to think about it like the usual, for example, multidimensional Euclidean one.

From the mathematical point of view, data need to be embedded (immersed) in some metric space. Nevertheless, pairwise comparisons usually are not mathematically correct distance or similarity functions, and the corresponding metrics appears to be violated. Therefore, it is necessary to recover the nonpositive definite similarity matrix for the limited set of elements.

Based on the well-known discrete Karhunen-Loève expansion [2] we can remove contributions, first, of negative eigenvalues and, second, of lowest ones (to define, for example, the smaller dimensionality of an eigenvector space in a standard case of a set of features). This transformation reduces the rank of the initial similarity matrix. It is natural for the feature-based approach to reduce a space dimensionality (we can re-calculate distances and scalar products), but it is not like this for pairwise measurements only, because the initial set of elements is reduced in fact. Indeed, we need to recognize correctly all elements from the given set by machine learning and clustering algorithms, for example, specially developed for the featureless case [3].

Therefore, we need optimal corrections to recover the violated metrics and get the positively definite similarity matrix for the limited set of elements under investigation (objects in some unknown imaginary feature space or features themselves). As a result, our approach differs, for example, from the well-known multidimensional scaling problem: we need to recover the metrics only, not the feature space itself.

Here we describe our previously developed approach to recover Euclidean metric based on the idea of minimal corrections of similarity matrices [4]. We develop here our approach to improve the conditionality of matrices of pairwise comparisons and discuss results of experiments.

II. RELATED WORKS

In general, the metric embedding is the known strongly mathematical problem [5]. Recently, this problem appears to be practical for modern data in machine learning, image analysis and data mining in the form of matrices of pairwise comparisons [6]. For example, such matrices represent similarity score between protein macromolecules in bioinformatics [7] or dissimilarity between images and shapes in image registration [8], etc.

If the matrix of pairwise similarities is positively (semi)definite, then the set of elements is immersed in some metric space, e.g. Euclidean, with dimensionality not more than the rank of the matrix. In practice, the matrix of pairwise similarities can appear to be nonpositive definite itself, because measurements are not usually scalar products. Moreover, standard transformations to get, for example, symmetric matrix can produce the nonpositive definite one.
In general, this problem needs to be solved before the data analysis and even before the measurement process itself. In particular, this is a problem of a multi-kernel similarity measurements and multimodal image registrations [9, 10].

From the other side, the idea to get the correct matrix \( S(n, n) \) of pairwise comparisons of \( n \) elements is well known too, for example, in expert questioning for ranking of alternatives (projects, enterprises, etc.). In this case, it does not need an expert individual opinion to be transitive, for example, like so-called Luce condition \( s_{ij} = s_{ik} s_{kj} \) about three alternatives \( i, j \) and \( k \). On the contrary, it needs to recover the so-called supertransitive matrix to comply with the Luce axiom [11] by some technique and evaluate the expert’s ranking.

Our case is more complicated, since measurements are presented in scales of higher level than ordinal and interval ones. As a result, our pairwise comparisons need to comply with the triangle inequality and, more closely, the cosine theorem.

In machine learning, the practical embedding problem has been developed in details, for example, in [6] for a set presented by a conditionally negative definite distance matrix \( D(n, n) \) to be immersed in Euclidean metric space.

Here, on the contrary of [6], we investigate a set presented by a positively definite similarity (or scalar products) matrix \( S(n, n) \) [4]. It needs to point the attention, the metric recovery problem falls into the problem of singular matrices [12, 13], since the recovering principle developed here gives the about to be zero determinant of the corrected similarity matrix in general [4]. Therefore, we repair ill-conditioned matrices of scalar products and improve the conditionality of them in the single correcting process.

The approach developed here differs from sophisticated computational methods, because we can correct data itself (not metrics) and regulate this process to get well-conditioned data.

### III. METRIC CORRECTIONS AND THE CONDITIONALITY

It is known, a condition number characterizes matrix degeneracy. If a square matrix \( A \) of coefficients in \( A x = b \) is about to be completely degenerated, then small deviations in \( A \) and \( b \) give large changes in a solution \( x \), and vice versa for a nondegenerated matrix. A well-conditioned matrix has a small condition number.

Let \( S \) be a square matrix. The matrix condition number can be defined as \( \text{Cond} (S) = ||S|| ||S^{-1}||. \) The matrix norm can be defined by different ways, e.g. \( ||S|| = \max ||x|| \). Hence, the norm of the inverse matrix is \( ||S^{-1}|| = 1/\min ||x|| \). The condition number of the positive definite matrix \( S(n, n) \) is

\[
\text{Cond} (S(n, n)) = \lambda_1/\lambda_n, \quad \lambda_{\text{max}} = \lambda_1 > \ldots > \lambda_n = \lambda_{\text{min}} > 0.
\]

Metric violations cause negative eigenvalues, whereas recovering of a metrics removes them. Hence, it is suitable definition of the condition number in our problem.

Let the sequence of principal minors \( S(1,1)=1, S(2,2), \ldots, S(k, k), \ldots S(n, n) \) be given, where the normalized matrix of scalar products is \( S(n, n) \). The corresponding sequence of decreased values is

\[
S_1 = 1, \ldots, S_2 = \det S(k, k), \ldots S_n = \det S(n, n).
\]

The value \( S_i \) is positive, if no metric violations in configuration of \( k \) elements, represented by mutual comparisons. In the other case, this is the alternating-sign sequence \( S_1, \ldots S_n \). If current \( S_i < 0 \), then the current element represented by \( s_{ij} = s_{ik}, i = 1, \ldots, k \), causes a violation to be eliminated and get \( S_i > 0 \).

Unfortunately, the idea of minimal corrections of similarity matrices [4] isn’t connected directly with the new value \( S_i \) of the corrected minor \( S(k, k) \) with its condition number \( \text{Cond} (S(k, k)) \). We know only that \( S_i = \prod_{i \leq j} \lambda_j \) and \( k = \sum_{i \leq j} \lambda_j, k = 1, \ldots n \). According to it, we can declare only that \( \lambda_{\text{max}} = \lambda_1 > 0 \) decreases, while \( \lambda_{\text{min}} \) increases to become positive \( \lambda_{\text{min}} = \lambda_k > 0 \). As a result, the condition number \( \text{Cond} (S(k, k)) = \lambda_i/\lambda_k \) decreases to improve the conditionality of \( S(k, k) \). The best correction of \( S_i < 0 \) gives \( S_i = 0 \) for minimal deviations of corrected elements in \( S(k, k) \). Unfortunately, this result isn’t suitable, since \( \lambda_{\text{max}} = \lambda_k = 0 \) with \( \text{Cond} (S(k, k)) = \infty \).

Such contradiction can be eliminated based on stronger corrected matrix elements of the last row and column of the current minor \( S(k, k) \).

From the other side, \( S_i \leq S_{k-1} \) after correction. If \( S_k = S_{k-1} \), we show, that corrected elements of the last row and column become zeros \( s_{ij} = s_{ik} = 0, i = 1, \ldots k \). In this case, we can get the minimal \( \text{Cond} (S(k, k)) = \text{Cond} (S(k-1, k-1)) \). Nevertheless, such correction isn’t suitable for new values of pairwise comparisons.

Hence, the corrected value \( S_k = C \) is the optimization parameter, which only indirectly connected with the condition number of the minor \( S(k, k) \). We need to coordinate two contradicting conditions: minimum of deviations and minimum of a condition number.

Here we propose a heuristic way based on mutual analysis of diagrams of two functions \( \text{Cond} (S(k, k)) \) and

\[
D_\alpha = \sum_{i=1}^{k-1} (s_{ik} - x_i)^2 \quad \text{dependent on } 0 < \alpha < 1, \text{ where } \alpha \text{ is a part of value } S_{k-1}, C = \alpha S_{k-1} \text{ is the optimization parameter, } x_i \text{ is a new value of corrected } s_{ik}, \text{ where } s_{ik} = s_{ik}.
\]

The proper optimization problem for the predefined \( 0 < \alpha < 1 \) with \( C = \alpha S_{k-1} \) under constraints is

\[
D_\alpha \rightarrow \min, \quad \sum_{i=1}^{k-1} \sum_{j=1}^{k-1} c_{ij} x_i r_j = C
\]

where \( r_j \) is the element of the matrix \( R = S^{-1}(k-1, k-1) \), is solved by the method of Lagrange multipliers.
IV. EXPERIMENTS

Experimental data are represented by the similarity matrix with elements in the range $0 \leq s_{ij} < 1$. This is the matrix of normalized similarity scores between 418 protein sequences. The set of proteins was investigated in Lawrence National Lab., Berkley, USA by Dr. San-Ho Kim [7]. Protein sequences were compared by the Fasta program to investigate in Molecular biology the separability problem of protein classes represented in the form of amino-acid sequences. The initial 420 protein sequences from 51 classes were selected based on minimal similarity between them.

Later, separability of this set was investigated in a featureless problem [14] and the initial set was reduced to 418 proteins, because two pairs of proteins were represented by the identical set of pairwise scores with other proteins. This similarity matrix has 415 eigenvalues in the range from 77.767 to 0.0103, and 5 negative eigenvalues: -0.002479, -0.008042, -0.016319, -0.053135, and -0.077528 (Fig. 1).

According to it, the sequence of principal minors has five sign changes (Fig. 2, 3) and the last determinant is negative.

We can build different principal minor sequences with corresponding decreased values. Nevertheless, only some of them in a case of the alternating-sign sequence appear to be optimal, since all minors alter signs at the end of the sequence.

We can show, that the number of corrections can be slightly more than the number of negative eigenvalues, and in the ideal case equals to it. Hence, the sequence in Fig. 2 is optimal and the first negative minor appears to be not closer to the end of the principal minor sequence than the number of negative eigenvalues. Fig. 3 shows the last fragment of the sequence with sign changes for principal minors 411, 413, 416, 417, and 418.

In this case, seven minors were corrected: 411, 412, 413, 414, 416, 417, and 418 (Table 1). In general, correction of a current negative principal minor in the nonoptimal sequence causes the additional chain of negative minors to be corrected too. Therefore, 411th and 413th minors generate two chains of...
disturbances: 411, 412 and 413, 414. The optimal sequence of principal minors minimizes additional disturbances after corrections.

Fig. 4 shows in the single scale several diagrams of a condition number \(\text{Cond}(\alpha)\), minimal deviations \(\text{D}^\alpha\), maximal \(\lambda_{\text{max}}\) and minimal \(\lambda_{\text{min}}\) eigenvalues of the corrected minor \(S(411,411)\) in the interval \(0 < \alpha < 1\) (horizontal axis), while \(0 < S_{411} \leq S_{410}, S_{411} = \alpha S_{410}\).

The intersection point of diagrams (1) and (2) for the minor \(S(411,411)\) corresponds to the condition number \(\text{Cond}_{\alpha}(S) = 23036.74\) and the minimal deviation \(D_{\alpha} = 0.08\).

It is clear, the heuristic point \(\alpha = 0.26\) can be shifted at least to \(\alpha = 0.3\) with \(\text{Cond}_{\alpha}(S) = 22110.24, D_{\alpha} = 0.087\), or even to \(\alpha = 0.4\) with \(\text{Cond}_{\alpha}(S) = 21204.03, D_{\alpha} = 0.109\).

These diagrams show that the suboptimal correction of \(S_{411}\) to be positive and near the zero, for example, for \(\alpha = 0.01\) gives \(\text{Cond}_{\alpha}(S) = 383759.995, D_{\alpha} = 0.054\). As a result, the optimal condition number is up to 20 times less than nonoptimal one.

Correction of other minors gives the same type of diagrams and corresponding parameters (Table 1). Generally, the optimal condition number can be significantly reduced relative to the nonoptimal one.

**Correction of other minors** gives the same type of diagrams and corresponding parameters (Table 1). Generally, the optimal condition number can be significantly reduced relative to the nonoptimal one.

| Table 1. Conditionality of Minors |
|-----------------|-----------------|-----------------|-----------------|
| Size | \(\alpha=0.01\) | \(\alpha=0.3\) | \(\lambda_{\text{min}}\) | \(\lambda_{\text{max}}\) |
| 411  | 383759.995 | 22110.24 | 0.087 | 0.0034 | 75.698 |
| 412  | 83425.308  | 25926.104 | 0.121 | 0.0029 | 75.894 |
| 413  | 279173.092 | 26320.001 | 0.091 | 0.0029 | 76.108 |
| 414  | 113028.505 | 27741.608 | 0.159 | 0.0028 | 76.319 |
| 416  | 49729.213  | 44612.865 | 0.177 | 0.0017 | 76.785 |
| 417  | 295262.916 | 44746.375 | 0.043 | 0.0017 | 77.001 |
| 418  | 79583.506  | 44914.965 | 0.124 | 0.0017 | 77.226 |

**CONCLUSION**

Recently, experimental data are usually represented by mutual pairwise comparisons of similarity and dissimilarity of set elements. This set needs to be embedded into some metric space for correct using by machine learning algorithms. The correct embedding is subjected to a condition of a nonnegative definite matrix of pairwise similarities of set elements.

Various similarity and dissimilarity measures are used in practice. Nevertheless, many of them are empiric and not correct as metric functions. Hence, it needs to correct real matrices of pairwise comparisons to guarantee the positive definiteness of corresponding matrices of scalar products.

Unfortunately, the natural limit of minimal deviations of corrected values from initial ones leads to ill-conditioned matrices of scalar products with large condition numbers.

In this paper, we propose a way to improve the conditionality of matrices of scalar products and, accordingly, corresponding matrices of pairwise comparisons.

The novelty of the proposed method consists in detecting objects, which make disturbances in metrics. Pairwise comparisons of such objects with others are corrected to provide minimal deviation of new values from initial ones. Generally, we can correct only some of the matrix elements. From the other side, the proposed method allows to get the suitable condition number of the corrected matrix.

Unfortunately, today we can't obtain the strong mathematical result for optimal condition numbers based only on diagrams of condition numbers and deviations. The more strong idea consists in some standard statistical hypothesis testing, e.g. of correlation coefficients relative to a significance level, since scalar products can be used as statistical variations.

**REFERENCES**


Dataless Black-Box Model Comparison

1st Christoph Theiß  
Computer Vision Group  
Friedrich Schiller University Jena  
Jena, Germany  
christoph.theiss@uni-jena.de

2nd Clemens-Alexander Brust  
Computer Vision Group  
Friedrich Schiller University Jena  
Jena, Germany  
clemens-alexander.brust@uni-jena.de

3rd Joachim Denzler  
Computer Vision Group  
Friedrich Schiller University Jena  
Jena, Germany  
joachim.denzler@uni-jena.de

Abstract—In this paper, we present a method for estimating the similarity of two black-box models that does not depend on the knowledge about specific training data. This method can be used to identify copies of or “stolen” machine learning models. It can also be applied to detect instances of license violations regarding the use of datasets.

We validate our proposed method experimentally on the CIFAR-10 and MNIST datasets using convolutional neural networks, generative adversarial networks and support vector machines. We show that it can clearly distinguish between models trained on different datasets. Theoretical foundations of our work are also offered.

Index Terms—model comparison, function space, black-box

I. INTRODUCTION

In the near future, machine learning model theft can become a problem either through the abuse of prediction APIs [16] or by stealing data directly from cloud services [13]. Issues like the bypassing of licenses and terms of service or copying of models trained on confidential data can arise. Proving or testing for the theft of models or training data is an important task given these issues.

In this paper, we present a method for estimating the similarity of two black-box models that does not depend on the knowledge about specific training data. Our contributions include a formula for estimating the distance between two models represented by elements of a function space as well as a proof that this formula is indeed a distance function.

A. Stolen Data

The ImageNet dataset [4], specifically its variant for the annual visual recognition challenge [15] has enabled the successful training of very large models and represents an important benchmark even today. To acquire the images of the dataset, one has to agree to a number of terms regarding the usage of the images. Commercial use is explicitly forbidden because the images are compiled from a variety of sources.

Microsoft’s COCO dataset [10] uses images from Flickr1, which is governed by the Yahoo terms of services. These explicitly state that commercial use of the content is not allowed.

However, one needs datasets of this size for the initial training of large models such as convolutional neural networks (CNN). To the best of our knowledge, there are no large-scale datasets that allow for commercial use. Open Images2 attempts to fill this gap, but offers no warranty w.r.t. the actual license status of the images.

On the other hand, there is no straightforward way for the copyright owners to find out whether a given model has “seen” an image or not. In this paper, we offer a tool towards this difficult task by showing that similar models trained on different datasets can be clearly distinguished.

B. Stolen Models

Since training a CNN on a large dataset from scratch can be time- and resource-intensive, a reasonable choice would be to adapt or fine-tune an existing model. Models can be acquired even in cases where reuse is not explicitly permitted, e.g. using a technique such as presented in [16].

First approaches for watermarking neural networks are proposed in [11], [12]. However, they focus specifically on neural networks and the task of identifying one network. The method proposed in this paper can be applied to a variety of different machine learning methods without any knowledge about the weights, parameters or dataset.

Further, we not only explore the comparison of black-box models but also whether it is possible to determine if a machine learning algorithm or a convolutional neural network specifically is trained on a particular dataset.

II. MODEL COMPARISON

We propose a method to compare two models based on their output given random noise samples. The distance between the two models is represented by the distance

\[ d(f,g) := \|f - g\| = \frac{1}{V(C)} \int_{C} \|f(x) - g(x)\|_{2} dx \]  

(1)

between two functions \( f, g \) in the same linear normed function space \( T \) of continuous functions mapping a compact subset \( C \) of a metric space with volume \( V(C) \) onto a real valued vector space \( D \).

This distance is stochastically approximated by calculating

\[ d_z(f,g) := \frac{1}{n} \sum_{i=1}^{n} \|f(z_i) - g(z_i)\|_{2} dx \]  

(2)

with \( n \) random samples \( z_i \) uniformly distributed in \( C \).

1http://www.flickr.com

2https://github.com/openimages/dataset
The independence from any dataset and the comparison as
functions allow the method to be applied to a variety of dif-
gerent machine learning methods. These include convolutional
neural networks (CNNs) [9], generative adversarial networks
(GANs) [5] and support vector machines (SVMs) [2].

We can calculate the distance between two functions because they
can be represented by continuous functions with compact
support. Additionally, the analysis of black-box models is possible
without the knowledge of parameters, relying only
on predictions and confidence scores.

In the following section, we first prove that the function
d(f,g) in (1) is a distance function, i.e. it defines a metric in
the aforementioned function space. Since the direct calculation
of this function over the whole set C is computationally
impossible in most cases, we further show that d(f,g) can
be approximated stochastically using uniform noise over C.

**Theorem 1.** The distance
\[ d(f,g) = \| f - g \| = \frac{1}{V(C)} \int_C \| f(x) - g(x) \|_2 dx \]  
(between two functions f, g in the same linear function space
T of continuous functions mapping a compact subset C of a
metric space with volume V(C) onto a real valued vector
space D is a metric on T.

**Proof.** It is sufficient to show that
\[ \| f \| := c \int_C \| f(x) \|_2 dx \]  
is a norm on T, because we can then use (4) to induce the
metric (1).

For readability let from now on \( c = \frac{1}{V(C)} > 0 \). Both the
triangle inequality
\[ \| f + g \| = c \int \| f(x) + g(x) \|_2 dx \]
\[ \leq c \int \| f(x) \|_2 + \| g(x) \|_2 dx \]
\[ = c \int \| f(x) \|_2 dx + c \int \| g(x) \|_2 dx \]
\[ = \| f \| + \| g \| \]  
and the absolute homogeneity
\[ \| af \| = c \int \| af(x) \|_2 dx = c \int |a| \| f(x) \|_2 dx \]
\[ = |a| c \int \| f(x) \|_2 dx = |a| \| f \| \]  
of \( \| f \| \) can be derived from the linearity of integration, \( c > 0 \)
and the properties of the Euclidean norm. Further, \( \| \cdot \| : T \to \mathbb{R} \)
is positive semi definite. Especially, \( \| f \| = 0 \) if and only if
\( f \equiv 0 \) is the zero function. Since
\[ f \equiv 0 \Rightarrow \| f \| = c \int \| 0 \|_2 dt = c \int 0 dt = c \cdot 0 = 0 \]  
holds, \( f \equiv 0 \) implies \( \| f \| = 0 \).

We prove \( \| f \| = 0 \implies f \equiv 0 \) by contradiction. Assume
\( f \not\equiv 0 \) and \( \| f \| = 0 \). If \( f \not\equiv 0 \) then there exists an \( \epsilon > 0 \)
and \( x \in C \) such that \( \| f(x) \|_2 > \epsilon \). Further, because f is
continuous we know that there exists an open ball B around
\( x \) such that for every \( y \in B \) the inequality \( \| f(y) \|_2 > \epsilon \) holds.

Then
\[ \| f \| = c \int_C \| f(t) \|_2 dt \]
\[ \geq c \int_U \| f(t) \|_2 dt \]
\[ \geq c \epsilon dt \]
\[ \geq cV(U) \epsilon \]
\[ > 0 \]
shows the existence of a positive lower bound on \( \| f \| \), which
contradicts the initial assumption. We conclude that \( \| f \| = 0 \implies f \equiv 0 \).

That \( \| f \| \geq 0 \) follows from (5) and (6). However, it is
also important to note that there exists no \( f \in T \) such that
\( \| f \| = \infty \). In the following, let \( \Delta := \max_{\| \cdot \|_2} \| \cdot \|_2 \). It holds that
\[ \| f \| = c \int_C \| f(t) \|_2 dt \]
\[ \leq cV(C) \Delta \]
\[ < \infty \]
because the images of all \( f \in T \) are compact which follows
from the fact that C is compact and all \( f \in T \) continuous
(proof [14]).

Therefore, \( \| f \| \) is indeed a norm on T and it follows directly
that the function \( d : T \times T \to \mathbb{R} \) is a metric on T. \( \square \)

**Theorem 2.** The distance between two functions \( f, g \in T \) as
defined in Theorem 1 can be approximated by calculating
\[ d_z(f,g) = \frac{1}{n} \sum_{i=1}^{n} \| f(z_i) - g(z_i) \|_2 dx. \]  
with n random samples \( z_i \) uniformly distributed in C.

**Proof.** Since \( z_i \) are independent and identically distributed,
the strong law of large numbers holds and the sample average
converges almost surely towards the expected value for a large
enough number of samples n. The \( \lim_{n \to \infty} d_z(f,g) \) being equal
to the distance function as defined in (1) follows directly from
the definition of the expected value.

\[ \lim_{n \to \infty} d_z(f,g) = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} \| f(z_i) - g(z_i) \|_2 dx \]
\[ \overset{a.s.}{=} E \left[ \| f(x) - g(x) \|_2 \right] \]
\[ = \frac{1}{V(C)} \int_C \| f(x) - g(x) \|_2 dx \]
III. EXPERIMENTS

In this section, we validate our proposed method experimentally. To assess its general applicability, we test not only the popular convolutional neural networks (CNNs) [9], but also generative adversarial networks (GANs) [5] as well as support vector machines (SVMs) [2].

Additionally, we investigate for CNNs how the number of trainable parameters – i.e., the number of layers and feature maps per layer – influence the proposed method.

A. Setup

Unless stated otherwise, we use the same architecture for all trained CNNs. The same holds true for GANs. For the CNN architecture, 200 different weight initializations as well as different batches are chosen resulting in 200 different models. These models are divided into two subsets, where one half of the models is trained on the MNIST dataset [8] and the other half on a grayscale version of the CIFAR-10 dataset [7]. In the same manner 20 different GANs are trained.

All models are trained with a mini-batch size of 128 and the Adam optimizer [6]. We find that the standard hyperparameters of $\beta_1$ and $\beta_2$ with values 0.9 and 0.999 respectively are sufficient to train the models on both MNIST and CIFAR-10. As learning rate, we choose 0.0001 to ensure a stable training. All CNNs are trained for 10000 iterations whereas the GANs are trained for 25000 because of their slower convergence.

To obtain 200 different SVMs the datasets are shuffled differently for each model. All SVMs are trained for 1000 iterations or until convergence using the implementation provided by [17].

We can compare all models of one type at once by visualizing the distance matrix between them. This matrix results from the pairwise calculation of (2) for all models. The distances in all figures are scaled with $\ln(d(M, N) + 1)$ for a better visualization. The matrices can be split into four submatrices. The upper left submatrix displays the distances between models trained on the MNIST dataset. Likewise, the lower right submatrix indicates the distance between the CIFAR-10 models. The other two submatrices give identical information about the distance between models of the two types.

As a reference point on how well the models perform on the initial classification task, Table I shows the overall recognition rate averaged overall all models on a test set for their respective dataset. The test sets consist of 10000 samples for both MNIST and CIFAR-10.

TABLE I

<table>
<thead>
<tr>
<th>Method</th>
<th>MNIST</th>
<th>CIFAR-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>99.6%</td>
<td>66.5%</td>
</tr>
<tr>
<td>SVM</td>
<td>90.3%</td>
<td>36.3%</td>
</tr>
</tbody>
</table>

B. Convolutional Neural Networks

First, we evaluate our method on convolutional neural networks, all with an architecture similar to LeNet [8]. The CNNs represent functions within the same function space. They share the same architecture and therefore have the same input and output dimensions. For two such functions represented by two models $M$ and $N$, we can define their distance as the mean over the Euclidean distances from the $n$ outputs generated by these models with noise as input in a high-dimensional space as in (2). Consequently, a distance matrix $\mathbb{R}^{200 \times 200}$ (representing the 200 models of our experimental setup) can be calculated using (2). By feeding random noise into the models instead of samples from training data, a dataless comparison between the models is possible.

Fig. 1. Distance matrix for the CNN models

a) Sample Size Influence: Initially, it is important to determine how many random samples are needed to be able to recognize a difference between two models at all. For that we choose $n = 100, 1000, 10000$ as different sample sizes. We report the mean and standard deviation for the different submatrices (Table II). The results show that a small amount of samples is already enough to clearly distinguish between the two types of models. However, we will continue to use $n = 10000$ for all further experiments to increase the probability of meaningful results.

TABLE II

<table>
<thead>
<tr>
<th>$n$</th>
<th>MNIST/MNIST</th>
<th>CIFAR/CIFAR</th>
<th>MNIST/CIFAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.640 ± 0.267</td>
<td>0.059 ± 0.107</td>
<td>1.236 ± 0.064</td>
</tr>
<tr>
<td>1000</td>
<td>0.632 ± 0.264</td>
<td>0.063 ± 0.108</td>
<td>1.236 ± 0.062</td>
</tr>
<tr>
<td>10000</td>
<td>0.634 ± 0.263</td>
<td>0.061 ± 0.108</td>
<td>1.236 ± 0.062</td>
</tr>
</tbody>
</table>

b) Model Comparison: Fig. 1 shows the distance matrix resulting from the pairwise calculation of (2) for all 200 networks. In combination with Table III one can make two major observations.
c) Model Size Influence: To assess the effect of the model size we train 48 models - 24 on each dataset - varying in depth and width by adding \( l = 0, 1, 2, 3 \) additional layers and modifying the number of feature maps \( f = 2, 4, 8, 16, 32, 64 \). Every model has at least two convolutional layers and two fully connected layers. The first convolutional layer has \( f \) feature maps, whereas the second, as well as every additional layer, compute \( 2f \) feature maps. The two fully connected layers consist of \( 32f \) and 10 neurons respectively. The additional convolutional layers are inserted before the first fully connected.

From Fig. 2 it is evident, that the model size greatly influences the distance between two models. One can see that more complex models - those with more trainable parameters - can be differentiated more easily whereas it is generally harder with smaller models. This can be beneficial when trying to identify if a model is trained on one dataset rather than if it has a specific architecture because modern networks usually have a high number of parameters.

<table>
<thead>
<tr>
<th>Method</th>
<th>M vs. M</th>
<th>C vs. C</th>
<th>M vs. C</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>0.64 ± 0.26</td>
<td>0.06 ± 0.10</td>
<td>1.24 ± 0.06</td>
</tr>
<tr>
<td>GAN</td>
<td>44.96 ± 17.52</td>
<td>252.59 ± 85.854</td>
<td>202.21 ± 18.06</td>
</tr>
<tr>
<td>SVM</td>
<td>0.21 ± 0.06</td>
<td>0.65 ± 0.17</td>
<td>44.44 ± 0.17</td>
</tr>
</tbody>
</table>

### C. Generative Adversarial Networks

In the following experiment, we apply our methods on generative adversarial networks. We try to distinguish the discriminators of 20 downsized Cramer GANs [1] trained in the same manner as the CNNs in the model comparison experiment previously. We use the identical setup and calculate the distance between the discriminators with \( n = 10000 \) random samples. They differ from the networks in the previous experiment insofar as that they are a part of a model that is trained in an unsupervised manner and a higher output dimension. The compared outputs of the discriminators are of size \( \mathbb{R}^{256} \).

In Fig. 3 one can see that it is harder to tell what kind of dataset a GAN is trained on solely based on the discriminator even so the models have more trainable parameters which should help to distinguish them as we have seen in the earlier experiment. However, whereas the input size remains the same, the dimension of the outputs of the discriminators is much larger than the ones in our experiments with CNNs. From that, we can conclude that the number of output dimensions is another influencing factor for our proposed method.

![Fig. 3. Distance matrix for the GAN models](image-url)
minor if they are trained on the same dataset compared to two models trained to classify different dataset.

In the case of SVMs, the proposed distance function can be used to identify on what dataset a model is trained on but it is nearly impossible two differentiate between two SVMs trained on the same dataset.

![Distance matrix for the SVM models](image)

Fig. 4. Distance matrix for the SVM models

IV. CONCLUSION

We introduce a new method to compare two black-box models solely based on their outputs given random noise samples and successfully apply it to distinguish between the functions defined by different machine learning methods including convolutional neural networks, generative adversarial networks and support vector machines trained on different datasets.

We show that in the case of CNNs, distinctions are possible with as little as 100 noise samples, resulting in a very economic method of model comparison. The notion of distance between models is intuitive and our formula behaves as one would expect from a distance function. This is confirmed by our experiments and proven mathematically.

REFERENCES

In Defense of Active Part Selection for Fine-Grained Classification

1st Dimitri Korsch
Computer Vision Group
Friedrich Schiller University Jena
Jena, Germany
dimitri.korsch@uni-jena.de

2nd Joachim Denzler
Computer Vision Group
Friedrich Schiller University Jena
Jena, Germany
joachim.denzler@uni-jena.de

Abstract—Fine-grained classification is a recognition task where subtle differences distinguish between different classes. To tackle this classification problem, part-based classification methods are mostly used. Part-based methods learn an algorithm to detect parts of the observed object and extract local part features for the detected part regions.

In this paper we show that not all extracted part features are always useful for the classification. Furthermore, given a part selection algorithm that actively selects parts for the classification we estimate the upper bound for the fine-grained recognition performance. This upper bound lies way above the current state-of-the-art recognition performances which shows the need for such an active part selection method.

Though we do not present such an active part selection algorithm in this work, we propose a novel method that is required by active part selection and enables sequential part-based classification. This method uses a support vector machine (SVM) ensemble and allows to classify an image based on arbitrary number of part features. Additionally, the training time of our method does not increase with the amount of possible part features. This fact allows to extend the SVM ensemble with an active part selection component that operates on a large amount of part feature proposals without suffering from increasing training time.

Index Terms—Fine-Grained Classification, Support Vector Machine, Ensemble, Bagging

I. INTRODUCTION

With the growing digitalization the amount of digital images rapidly increases. Consequentially, the need for precise automatic classification of these images was never so present. The newest hardware and software developments in the field of computer vision solve challenges from the last decades with ease and new challenges with more sophisticated tasks are created. It can be noticed on the latest benchmark datasets for object recognition like CUB-200-2011 [1]. This dataset consists of bird images from 200 species. The large number of different species and the fact that some species are only characterized by subtle differences in their appearance create a need for novel classification approaches. Therefore, fine-grained algorithms get more and more attention.

The classical way of extracting features from the entire image achieves remarkable results [2], but most related works [3]–[6] on fine-grained image classification use a part-based approach. The main idea is to combine the global feature with additional part features extracted from different locations of the image. Based on the extracted part features and the global feature the image is classified.

One of the drawbacks of considering all existing part features is that in some cases the features extracted to distinguish different classes do not contribute to the classification decision. Furthermore, the additional information in form of these part features often leads to an increased training complexity, since all features are considered regardless of their impact on the classification result. A possible solution is a part selection component which actively selects the part features for the classification process with the objective to increase the classification performance. This is similar to former research on optimal sensor data selection with the help of a sequential decision process [7].

An active part selection component requires a classifier that is capable of processing an arbitrary number of part features. Hence, we suggest in this work a sequential classification method that observes the part features one after another and
classifies them. Additionally, since the classifier takes only one part feature as input, its training time does not increase with the number of existing part features. This allows to train a classifier on a large amount of part features. Since the active part selection component is not the focus of this work, we simulate the part selection by computing the classification results for all possible part feature combinations. With this simulation we show that given such a selection component the upper bound of the suggested method’s recognition performance lies far above the current state-of-the-art part-based classification approaches.

The sequential classification is realized with an SVM ensemble, more specifically SVM bagging [8]. Works like [9], [10] have already used SVM ensembles to either improve the classification performance or decrease the training complexity. These approaches as well as the original bagging algorithm train the single classifiers on a random subset of samples. In contrast, we suggest to train a single SVM classifier on all training samples, but randomly select a single part feature for each sample, as shown in Figure 1.

The resulting ensemble of weak classifiers performs a classification on a single part feature, creating a classification decision for each classifier in the ensemble. In the end, the ensemble decisions for one part feature are aggregated to a single decision. Furthermore, the classification based on a part feature combination can be performed iteratively by observing one part feature after each other. This also allows each combination to have a different and also arbitrary number of part features.

II. RELATED WORK

A. Part-Based Fine-Grained Recognition

Most of the part based fine-grained recognition methods [3]–[6] fit a part detection model, which identifies part locations. At these locations, part features are extracted and the object is classified based on these features. Works like [3], [6] use a constellation or pose based approach. The authors estimate relative positions of some part proposals and find best groupings based on the training images. Based on these groupings, which defines the part model, the positions are estimated and used for feature extraction.


All of these methods perform a passive part selection since the parts are either selected based on the location information [3], [6] or totally independent from each other [3], [5], [11]. In this work we suggest a simulation of an active part feature selection component that computes the decisions based on prior knowledge about the observed image. Additionally, all of the mentioned methods have not investigated the upper bound recognition performance of their part-based approaches.

An active part selection component could be realized with recurrent attention models (RAMs) [12]–[14]. RAMs are iterative attention models that predict positions where the system should extract features next based on previously seen information. This information can be seen as prior knowledge, but all our experiments with RAMs so far could not outperform the previously mentioned state-of-the-art methods.

B. SVM Bagging

Bagging predictors were introduced by Breiman [8]. It is a “method for generating multiple versions of a predictor and using these to get an aggregated predictor”. In the original paper a single predictor was trained on a random subset of the training data, as shown in Figure 1a. The performance of such resulting classifier is mediocre, but if the prediction results are aggregated, for instance with a majority voting, then such an classifier ensemble is able to outperform a single classifier trained on the whole training dataset.

Wang et al. [9] illustrate also some empirical results on using SVMs as base classifier. In their work the authors were able to show some minor advantages of SVM bagging compared to regular SVMs and other classifiers. Linghu et al. [10] have used in their work SVM ensembles to reduce the training complexity and improve the recognition performance of their system. Additionally, they have evaluated different aggregation methods for the ensemble decisions.

III. METHOD

A. Part Feature Extraction

Our part feature extraction builds on the work of Simon et al. [3]. As shown in Figure 2, the authors estimate a part constellation model in unsupervised manner with the help of gradient maps of a pre-trained CNN. Then, they sort the part locations by their fitting score to the constellation model. Finally, the best ten part locations are selected and another pre-trained CNN extracts the part features on these locations. The extraction is performed on two different scales, resulting in 20 part features. These part features are concatenated together with the global feature to one single feature vector and a single SVM is trained.

B. SVM Ensemble Training

In this work, we consider the part features as distinct features, since we assume that not all present part features impact the classification result in a positive way. In order to allow an active part selection, the fine-grained classifier has to be able to handle changing number of part features as input. Hence, we propose an ensemble of SVMs as a sequential classifier.

We utilize the bagging algorithm [8], but instead of randomly selecting a subset of training samples, we select for each sample a random part feature, as illustrated in Figure 1. After an arbitrary amount of SVMs has been trained, the aggregated ensemble decision for a single part feature is computed by a majority voting over the ensemble classifiers.
C. Part Feature Combination Analysis

Next, we define how to compute a prediction for a given part feature combination \( \langle f \rangle_t = \langle f_1, f_2, \ldots, f_t \rangle \). Given such part feature combination and an classifier \( k \) we perform a maximum likelihood estimation under the uniform prior assumption:

\[
c_{ext} = \arg \max_c \prod_{x=1}^{t} p_k(c \mid f_x)
\]

This way of class estimation for a specific part feature combination allows compute the decisions in the form of class probabilities for each part feature separately and fuse these decisions. It also allows the usage of the SVM ensemble presented in this work as a classifier \( k \), because the input of the SVM ensemble is a single part feature.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th># of parts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simon et al. [3]</td>
<td>81.0%</td>
<td>21</td>
</tr>
<tr>
<td>Krause et al. [6]</td>
<td>82.0%</td>
<td>31</td>
</tr>
<tr>
<td>Jaderberg et al. [4]</td>
<td>84.1%</td>
<td>4</td>
</tr>
<tr>
<td>Liu et al. [5]</td>
<td>84.3%</td>
<td>3</td>
</tr>
<tr>
<td>Zheng et al. [11]</td>
<td>86.5%</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ours</th>
<th>Accuracy</th>
<th># of parts</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Parts (single scale)</td>
<td>77.4% ± 0.3%</td>
<td>11</td>
</tr>
</tbody>
</table>

This table shows the comparison of our work with other part-based methods on the CUB-200-2011 dataset. The best performance is achieved by our method, which uses all parts (single scale) with an accuracy of 77.4% ± 0.3% and 11 parts.

IV. EXPERIMENTS AND RESULTS

In our experiments we have used the CUB-200-2011 dataset. The SVM ensemble consists of 48 classifiers. The number of the classifiers can be chosen arbitrary. However, in our case, 48 SVMs could be trained in parallel most efficiently on our testing machine, a dual-socket server with two Intel Xeon E5-2650 v4 processors. Further investigation may also be needed to check the impact of this hyperparameter on the classification performance.

The experiments were repeated ten times to observe the variance of the approach. Every test run took seven hours for training the classifier ensemble and computing predictions for all possible part feature combinations.

As mentioned before, we are using the ten part locations matching best the constellation model of Simon et al. [3]. On these locations we extract the part features with a VGG19 network from the last pre-classification fully connected layer \((f_7)\). This results in a single feature having the dimensionality of 4096. Like Simon et al. [3], we extract the features from forms when it classifies the images based on all part features as well as only on a specific subset of the part features. As already mentioned before, we simulate the active part selection component by computing the classification results for all possible part feature combinations. Since we use 20 part features, we compute the predictions for \(2^{20} = 1,048,576\) part feature combinations. With these combinations we are able to estimate the upper bound for the recognition performance for these part features. First, we select the best combination for all training samples, then best combinations for each class and finally for each single sample separately. This way we simulate three different active part selection methods. The first one can be constructed without any prior knowledge about the observed image. The later two methods would use a prior information about the given image in form of a class distribution or any other sample related information.
the locations on two different scales, which yields 20 part features. In our experiments we also consider a setup with a single scale, namely only ten part features.

The results of the mentioned setups can be seen in Table 1. First, we compare our baseline results where all parts were selected with the reference work of Simon et al. [3]. Here, the original work clearly yields better performance. Same observation can be seen if we select one combination that performed best for all samples or if we randomly select four arbitrary parts. The original strategy of using all part features still performs slightly better. However, the selected combination contains less than half of the additional information, namely only nine part features instead of all 21 (20 part and one global feature).

If we go further and select the best combination for each class or for each sample separately, then we can clearly outperform not only the reference work of Simon et al., but also other part-based approaches. The best combinations achieve state-of-the-art performances not only on the more comprehensive two-scale setup, but also in the single scale scenario. Thereby, at most only four parts are selected from the given 21 or 11. It is important to notice that the additional scale does not impact the mean number of selected parts, but the performance. Hence, we conclude that the extraction scale is as important as the active part selection.

Additionally, we have visualized how extracted parts may distract the classifier. In Figure 3 a sample from the CUB-200-2011 dataset is shown on the left (Figure 3a). On the right, Figure 3b illustrates the parts that were extracted with the constellation model of Simon et al. [3]. The first two rows are the ten parts extracted with the one scale and the last two rows are extracted with the second scale. In this particular example, if all of the parts are selected, the image is classified incorrectly. The prediction becomes correct if only the green parts are selected.

This qualitative result shows how active part selection can improve the classification by leaving out non-informative parts. It illustrates that the unsupervised part extraction algorithm used in this work is not perfect. Some of the extracted parts do not even cover the observed object (second row, first and fourth column) or only cover a the object marginally (second column). On the other hand, the parts leading to the correct prediction mostly cover the object. The final understanding why these parts lead to a correct decision and how to select them automatically is an open research objective.

V. Conclusion

In this work we presented a novel approach for sequential fine-grained classification. The sequential classifier is an ensemble of SVMs implementing the bagging algorithm. In contrast to previous works on bagging algorithm, we used random part features instead of a random sample subset for the training of the ensemble. The advantage of the presented algorithm is the independence of the SVM ensemble from the number of the observed part features.

To compare our approach with other part-based fine-grained classification methods, we simulated an active part feature selection component. The simulation was achieved by computing the classification result for all possible part feature combinations. Afterwards, best combinations were estimated for all samples, for each class and for each sample separately. These best combinations represent an upper bound recognition performance given specific part features.

Based on the part features of Simon et al. [3] we showed that the upper bound recognition performance is way above the current state-of-the-art results. This leads to the conclusion that an active part feature selection component is the best way to improve the fine-grained classification.

References


The features of the recognition of the region of efficiency of microwave devices

Galina M. Antonova
Laboratory № 33
Trapeznikov Institute of Control Sciences Russian Academy of Sciences
ul. Profsoyuznaya 65, Moscow, 117997 Russia
gmant@ipu.ru

Abstract—In this article statement of the problem for optimization of microwave devices is considered from the standpoint of recognition of region of efficiency in which necessary values of given output characteristics are appeared. The application for optimization of meshes that are uniform in the multidimensional space of parameters is discussed. The solution of the approximate optimization problem using a specially created mathematical model of the device is debated. (Abstract)

Keywords—Modeling, Multiple-criterion optimization, Region of efficiency, Recognition, Parameter space investigation (key words)

I. INTRODUCTION

Microwave devices are widely used in various industries. They include accelerator equipment, powerful and super powerful colliders, microwave energy, radiolocation, radio equipment, long-haul communication and many others. In addition, microwave devices play important role in modern medicine.

The nature of the electron flow through resonators and the algorithm for controlling this movement play an important role in the description of microwave devices. The parameters of successive elements of the structure (resonators, etc.) serve as levers for control. The scheme and control algorithm for each device are unique and depend on the skills of experts developers.

The process of construction of microwave devices contains several stages. There is a very responsible pilot project stage. It is devoted to the selection of the physical structure and associated geometrical parameters. A physical structure and numeric values of constructions parameters of device define the features and quality of its work. It is known that dimension between resonators gap (floating-drift length), own frequencies and quality factors of resonators are key parameters for power klystrons. So necessary output characteristics of device will depend on the correct choice of these parameters.

This problem was considered in papers [1-4]. The required values of input parameters can be successfully determined after the solution of optimization problems. New results for approximate optimization of span multiple-cavity klystrons were obtained by application of the aggregate-decomposition analysis. This approach allowed performing a significant simplification of the mathematical description of the microwave device without loss of adequacy.

After creating an adequate model of the device it is necessary to solve the problem of choosing the method of finding such regions of values of the input parameters, in which maximum efficiency factor in set frequency band for set input power level is achieved. This will be the second design stage.

For complex objects it is advisable to implement a search procedure of approximate optimization. For the algorithmically-defined indices of quality and multidimensional multivariable region of changing of input parameters values most simple and convenient method is a simple search of indices of quality with greatest values among existent by means of viewing the space of input parameters at the points of uniform mesh. This method is called the method of uniform probe and discussed in detail in the book [5]. If the grid is uniform in the multidimensional space of parameters it helps to implement search in a reasonable timeframe. The viewing area in this case will have maximum volume. Now $LP_{r}$ - sequence or Sobol’- sequence has good properties of uniformity and it is the best among known uniform sequences [6-9].

A remarkable property of the $LP_{r}$ - sequence is that by doubling the number of points in the space of input parameters for the experiment, the new points will be placed evenly among the old points. The density of the points to view the values of the objective functions will increase. The probability of finding the approximate extreme values of the indices of quality will also increase.

In [5] approximate solution for formulated problem of searching of the region of efficiency in input parameters space is described. The main issue is to determine such previously unknown region in which values of indices of quality are better than in other regions. The name region of efficiency is proposed for this area and name $LP_{r}$ - search with averaging shows the changes in the widely known search procedure suggested in [6]. This procedure can be performed in acceptable time with the help of modern computer
technology. However for implementation in full it demands still more time for imitation experiments with computer model than for simple probing of the parameter space to validate the values of indices of quality in various combinations of values of input parameters as in [2]. This implies that adequate computer model with high performance is required for the successful approximate optimization. The number of experiments can be very large.

There are several examples of successful implementation of the proposed approach. In [5, 10] typical applications of this procedure are considered. In a number of papers the difficulties in realization of $LP_{\tau}$ - search with averaging are discussed and possible techniques to overcoming of these difficulties are presented.

Principle of adaptive control of a complex system, operating in difficult conditions when it is need to save resources without losing the high quality of the system control is long and widely known. A vivid example of such systems is data communication systems. One application connects with such system functioning according algorithm with adaptive control of transmitter power through short-wave radio channels with fading [5]. The $LP_{\tau}$ - search with averaging approach is used to seeking a region of efficiency of the data communication systems in space of parameters defining the characteristics of the fading.

Procedures for selection of noise - resistant correcting code are investigated in another application [5, 10]. These procedures involve two stages. Problem under consideration deals with Viterbi decoder i.e. special algorithm Viterbi for decoding is considered. The solution must be determined for systems operating in short-wave radio channel with fading. The study is organized for check a few codes to select the best among them. At first stage solution of this problem allows to define regions of efficiency for several noise-resistant correcting codes. At second stage the existence of common region of efficiency for several given codes is tested. In advance it is unknown whether it will be possible to build an intersection of individual regions of efficiency. If common area among regions of efficiency for several various codes is detected then discrete problem is decided for determination of the best code in the group from several noise-resistant correcting codes.

A useful application of this approach is related to the design of the navigation system. The definition of coefficients for skip checking block in Net Satellite Radio-Navigation System by using of $LP_{\tau}$ - search with averaging algorithm is demonstrated in [1, 5, 10].

After rational parameter values are selected, it is necessary to perform a series of experiments on the model to verify the obtained solution. This will be the third design stage.

The next stages will solve the problem of creating the device, checking the quality of its operation and transfer the results of the project to the customer.

II. RECOGNITION OF REGION OF EFFICIENCY IN THE $LP_{\tau}$ - SEARCH WITH AVERAGING

A. Statement of the Problem in the $LP_{\tau}$ - Search with Averaging

Let's choose a set of indices of quality among output variables of system under investigation for statement of a multi-parameter multi-criteria optimization problem

$$K_j, j = 1, J$$

in $LP_{\tau}$ -search with averaging approach in the form of multidimensional integrals:

$$K_j = \int_0^\infty \int_{\Omega} \int_{\mathcal{G}} \int_{\mathcal{K}} \int_{\mathcal{T}} f_j(\alpha(t), \omega) w_{\Omega}(\alpha(t), \omega) d\omega \, d\alpha \, dt$$

Here $J$ is the total number of indices of quality characterizing the considered object, $\mathcal{G}$ - the region of efficiency which is estimated in the process of searching a solution of optimization problem, $\Omega$ - the domain of variation of values of stochastic parameters, $f_j(\alpha(t), \omega)$ - the function describing the $j$-th quality index, $\alpha(t)$ - input parameters vector with dimensionality $n_1$, $\omega$ - external and internal stochastic noise vector with dimensionality $n_2$, $t$ - time.

$$w_{\Omega}(\alpha, \omega) = w(\alpha, \omega) / \int_{\Omega} w(\alpha, \omega) d\omega$$

the distribution density, normalized relatively the region $\mathcal{G}$, $w(\alpha, \omega)$ - the distribution density satisfying the condition:

$$\int_{\Omega} w(\alpha, \omega) d\omega = 1.$$

Such statement of the problem can be extended to a more complex case with index of quality in form of continues curve. The corresponding function is transformed into a vector of point’s indices of quality by means of application of discretization procedure. So index of quality in form of continuous curve will be replaced by means of a set of point’s criteria. As an optimal or rational value for such criterion we will consider such vector, all coordinates of which fall into the area bounded from below and from above (tube). After discretization a finite set of criteria will be formulated. If there are a few indices of quality in form of continues curve special vector criterion will appear for every. Naturally, the procedures for calculating all components of every individual vector for index of quality in form of continues curve should be identical. An example of index of quality in form of continues curve from [4] is shown on Fig. 1. The case of continuous index of quality and procedure of discretization considered in detail in [11].
dimensionality of space of indices of quality after insertion of indices of quality in form of continuous curve will increase greatly. Sobol’ and others were defined space of coordinates of \( L_{P_{\tau}} \) - sequence for dimensionality, equal 50. For high-complexity problems, all necessary calculations can be performed with parallel computational structures application in an acceptable time and with acceptable accuracy.

![Amplitude frequency characteristic of optimal variant of device. Input power is equal 0.5, 1, 2, 4, 6, 8, 10 Wt](image)

**B. Search of Approximate Solution of the Optimization Problem**

The numerical evaluation of the region of the Euclidean space or the set of numerical evaluations from the space of evaluations with given metric according [5] should be determined as a result of solving the problem. Moreover, the point’s quality indices must constitute the Pareto set. For indices of quality in form of continuous curve components of every vector’s criterion must change in preset bounds. They should provide a pre-specified graphic image of characteristics. For the characteristics of a klystron in the form of a continuous curve, the coordinate values must be inside a predetermined tube.

If the region of efficiency is defined the solution of approximate solution will look like region of joint extremum of indices of quality of system under investigation. All preset conditions must be fulfilled:

\[
K_j(G) \geq K_{j_z}, j = 1, q. \quad (1)
\]

\[
K_j(G) \leq K_{j_z}, j = q + 1, J. \quad (2)
\]

\[
CK_{by} \leq CK_j \leq CK_{y}, j = J + 1, J + n_c. \quad (3)
\]

Here \( K_{j_z}, j = 1, J \) - boundary values of point indices of quality \( K_j(G) \), \( CK_j, j = J + 1, J + n_c \) - indices of quality in form of continuous curve, \( CK_{by} \) - bottom limit, \( CK_y \) - top limit of values of indices of quality \( CK_j, j = J + 1, J + n_c \).

Conditions (1) and (2) refer to point indices of quality and conditions (3) refer to indices of quality in form of continuous curve. The procedure for finding approximate solution brings together the methods of statistical modeling, the Monte Carlo method and approach named by \( L_{P_{\tau}} \) - search with averaging [5].

The averaged values of indices of quality in the region of efficiency will be better then in other regions from the space of parameters changing. Pareto set consisting from point’s indices of quality will contains incomparable variants of criteria. The main advantage of this search procedure consists in absence of the need to calculate derivatives.

If the number of simulation experiments is large enough, decision rule can be formulated. It will involve procedures of selection of suitable combinations of parameters for which point’s criteria will have satisfactory values and continuous criterion will has necessary shape [11]. For a large number of experiments the decision rule will be steady, not a random improvement among close variants of prototypes with close values of parameters.

Thus, the solution of the above problems showed that proposed approach is effective for the search of approximate solution when problem of optimization of dynamic stochastic systems is formulated for insufficiently studied complex systems.

**III. DESIGN OF HIGH-POWER KLYSTRONS**

Non-formalized statement of the problem similar to that proposed in the \( L_{P_{\tau}} \) - search with averaging, A.Yu. Baikov has offered in [12-14] for designing a klystron with high efficiency. He did a great job on creating discrete-analytical model of the span multiple-cavity klystron. By means of the aggregate-decomposition analysis He identified 5 complexes of input parameters to clarify the problem of creating a device with high value of coefficient of efficiency in a wide frequency range. The basic parameters preset the underlying structure of the device. They are selected in advance on the basis of the requirements to the device and General physical considerations. These include the fundamental frequency, accelerating voltage, total current, the number of rays (electron beams), the number of resonators, the working harmonics, the magnitude of the gaps, the characteristic impedance of resonators, self quality factors of the resonators, diameters of the beams, diameters of tubes, etc.
Another feature of the model associated with its approximate nature is the presence of incorrect points. They appear when it is necessary to take into account the phenomena poorly described in the model. These include such a phenomenon as the reflection of electrons. Sometimes iterative procedures performed in the modeling process do not converge well. Such points named as incorrect are provided with special labels and excluded from further consideration. Such situations lead to a more complex view of change space of model output variables. This space is multi-connected and incorrect points further complicate its shape.

The algorithm of approximate optimization is based on two methods and contains two major stages. At the first step preview of large area of the parameter change space is fulfilled with purpose to find points suspicious in the global extremum.

At the second step method macro step method is performed to verify the presence and location of a global extremum.

The original search method by performing calculations at the nodes of the composite mesh and method of move by means of macro steps was used in the modeling process. A composite mesh was done in the following way. The first thirteen variables are selected as the coordinates of the sequence. Further, seven coordinates was selected by the method of Holton. The remaining input variables with ordinal numbers larger than 21 are chosen as uniformly distributed pseudo-random numbers.

The mixing of points of uniform in multidimensional space grid with the best properties of uniformity among the existing grids with points of other grids, especially with points of the pseudo-random grid will lead to loss of uniformity in multidimensional space of parameters. In composite grid the effects that Sobol’ sequence was designed to eliminate will appear again. Some coordinates of the meshes will be repeated and some coordinates will be equal. I. M. Sobol proposed feature, named RANGE [15], as a quantitative measure of irregularity of distribution. Should be checked with the characteristic named range the properties of composite grid before using it as a tool for uniform probe of the parameter space.

The main achievements of A.Yu. Baikov are associated with the solving of two problems.

Firstly, the author has successfully realized the goal of achieving a simplified description of such a complicated object as the klystron without loss of adequacy to a real object.

Secondly, the author wanted to increase the speed of calculation compared to standard software for simulation such object.

Comparison of result of calculation for the dependence of efficiency factor from the power according to the programs KlipWin and MAGIC [16] is shown in Fig. 2.

$$K_{rel} = \frac{P_0}{\lambda^2} \left( \frac{U_0^{1/2}}{1 + \frac{U_0}{2}} \right)^{3/2} N_h.$$ \hfill (4)

$$\Theta = \omega_0 c \sqrt{1 - \left(1 + \frac{U_0}{2}\right)^2}.$$ \hfill (5)

$$\mu = I_b / 2r_e.$$ \hfill (6)

$$\alpha = \kappa / \lambda.$$ \hfill (7)

$$v = \rho P_0 / U_0^2.$$ \hfill (8)

Here $P_0$ - power of beam, $N_h$ - the number of rays, $\omega_0$ - the circular microwave frequency, $r_e$ - radius of channel, $r_b$ - radius of the beam, $\rho$ - the characteristic impedance of the resonator. The fundamental normalization done in advance is shown by underlining. Fundamentally normalized dimensionless quantities are obtained from the ordinary physical quantities by dividing them to their respective fundamental values, which are formed with using of fundamental constants, such as electron charge, electron mass, speed of light, dielectric constant, etc.

According to the General principle of equivalence [13], klystrons are completely equivalent to each other if all their dimensionless parameters coincide. The parameters $K_{rel}$ and $v$ are decisive. Within the same class of devices according to $K_{rel}$ and $v$ parameters, the dependence of the efficiency factor on the relative frequency and on the normalized input power does not change. This feature reduces the cost of modeling and optimizing for design of new devices.

Finally total combined discrete-analytical model of the device of the klystron type was built. This model has included a model of the drift tube combined with the model of distribution density of the microwave field in the gap and model of microwave clearance. Various iterative procedures for performing the calculations using this model have been successfully implemented in a special software complex KlypWin.

Significant amount of studies conducted with using the model and complex KlypWin showed the presence of a huge number of local extremums in the change space of such index of quality as efficiency factor. According to [14] their number reaches tens of thousands for the two-resonator klystron. For multiple-cavity klystron, the number of local extremums increases to $10^{13} - 10^{14}$.
IV. Conclusion

It is possible not agree with the second purpose of the research of A.Yu. Baikov connected with increasing the speed of extreme search. For a new product design in full amount is carried out once. It is possible to spend considerable effort by qualified experts and it is difficult to determine the customer who will be limited only by preliminary calculations in such a careful preliminary study.

Choosing a composite grid made up of fragments with different properties at the second step of algorithm of approximate optimization raises many questions. It is not known what properties the composite grid has in reality. The dense grid with the best properties of uniformity in multidimensional space of parameters covers space of parameters with uniform density both for area includes points with the correct calculation of the output variables of the model and for area consisting of incorrect points.

Complicated image of region of the change of output parameters and the region of efficiency of the klystron will increase the number of experiments independently of the used mesh. However, for the grid with the best properties of uniformity in multidimensional space of parameters that increase may be less significant because of the accuracy of viewing increases. In addition, there are known methods for generating Sobol'-sequence points in an area whose boundaries are precisely predefined.

Sobol'-sequence is convenient for parallelization because the coordinates of points of experiments can be calculated in advance. If the viewing density is not large enough, the number of points can always be doubled. As was said above the new points will be selected in the empty spaces between old points. According to the properties of the Sobol'-sequence, the coordinates of the points will be different and the property of uniformity of viewing will remain. Previously performed calculations will be used together with new calculations. If designer agree with increasing of investment of time of calculation, it is possible to choose a sufficiently dense grid.

ACKNOWLEDGMENT

Author thanks A.Yu. Baikov and I. M. Sobol' for the opportunity to read the publications covering the issues discussed in the article.

REFERENCES


Fig. 2. Results of comparision for programs KlifpWin and MAGIC


A. Yu. Baikov, O. A. Grushina, Model of electron beam transformation in a narrow tube and its application to the design of high-power klystrons. Moscow: Moscow university of finances and law (MFUA). 2013.


Approximation-based transformation of color signal for heart rate estimation with a webcam

Mikhail Kopeliovich  
I.I. Vorovich Institute for Mathematics, Mechanics and Computer Science  
Southern Federal University  
Rostov-on-Don, Russia  
kop@km.ru

Mikhail Petrushan  
Center of Neurotechnologies  
Southern Federal University  
Rostov-on-Don, Russia  
drn@bk.ru

Abstract—Photoplethysmography (PPG) is a method for contactless heart rate estimation through the analysis of slight variations of skin color caused by changes in the blood volume in vessels. Skin color variation registered by a camera is called color signal. Recent studies proved that some PPG methods are accurate enough if applied on videodata recorded by common web-cameras that makes them potentially applicable for long-term health monitoring in house or office conditions. In this study, we developed novel Approximation-based transformation method of signal processing and combine it with popular pre-processing and postprocessing algorithms. Approximation-based transformation is the procedure of computing an approximation signal that consists of leading coefficients of the local square polynomial approximation of the color signal.

Index Terms—Remote photoplethysmography, Color signal, Signal processing, Heart rate evaluation

I. INTRODUCTION

Photoplethysmography (PPG) is one of the effective methods of optical evaluation of blood volume changes in microvascular system of skin.

PPG has been successfully used in different medical applications, for example, the measurement of oxygen saturation, blood pressure and cardiac output, and also the detection of peripheral vascular diseases [1]. Some kinds of PPG methods, for example, webcam-based PPG, could possibly be utilized in such long-term monitoring, as they are contactless.

A number of methods for heart rate evaluation by webcam-PPG were developed recently [2, 3, 4]. Some of them claim to be usable with common low-cost web-cameras in particular environmental conditions and motion scenarios [5, 6]. Most of these methods implement the following processing pipeline: a) extraction of so called “color signal” (CS) from certain skin areas; b) noise reduction and band-pass filtering of color signal; c) final processing of color signal and estimation of heart rate value. The effectiveness of the extraction of the color signal strongly depends both on the nature of the color signal and the positions of the areas for extraction. Usually this signal is some function of R - red, G - green, and B - blue color components [7, 8], averaged over some facial or body areas. The performance of the second and third stages of the processing pipeline depends on particular algorithms of signal denoising, filtering, and analysis that are usually based on Fourier transformation and spectral analysis.

In this work, we evaluate impact of the proposed method of Approximation-based transformation on accuracy of heart rate estimation.

II. RELATED WORKS

Early methods and devices for contact PPG were based on the analysis of variations in intensity of light passing through a finger [1, 9]. The alternative approach to photoplethysmography, which is based on the analysis of slight variations of skin color recorded on video was proposed by Wieringa et al. [10] and Verkruysse et al. [2], and is known as color based remote photoplethysmography (rPPG), or imaging photoplethysmography (iPPG) [11]. rPPG methods work properly with common webcams capable of capturing frames in the visible spectrum at 10-30 frames per second [12, 5, 6].

Typical steps of webcam-based photoplethysmography pipeline are as follow [7, 13]: 1) extraction of color signal from video, 2) extraction of PPG signal (that represents only changes of blood volume in vessels and is invariant to external conditions such as the characteristics of the camera and lighting) from color signal, and 3) heart rate calculation by PPG signal analysis.

A. Color Signal Extraction

rPPG methods are usually applied to data obtained from a video, where the person’s face is visible in each frame [13, 6]. Therefore, color signal is extracted from skin areas located on a face. The most frequent approach to detect such areas is to use an algorithm for face detection [13] such as the Viola–Jones [14] or Lienhart and Maydt [15] algorithms. These algorithms take a frame with the subject’s face in it and return coordinates of a bounding facial rectangle, which is expected to fit the area of a face in the image.

Areas of color signal extraction, which are also known as regions of interest (ROIs), can be chosen in different ways. In some cases, color signal was extracted from the large central part of the facial rectangle [13, 4], in others, certain small areas were selected within the facial rectangle. In the current work, ROI was determined in relative coordinates of the facial bounding rectangle, occupying 100% of its heights and 60% of its width and placed in the center of the rectangle. In spite
of its coverage of non-skin areas represented in the middle part of face, it covers large skin areas effectively.

Due to possible movements of the subject's head and body, ROI could move from its initial positions, and therefore should be tracked robustly. The tracking-by-detection approach based on the Viola–Jones algorithm was used in the current work.

B. PPG Signal Amplification

The color signal acquired by rPPG methods contains a PPG signal, which has relatively low amplitude [10, 2]. Besides the PPG signal, a number of factors influence color signal: pose, current physiological processes, lighting conditions and the characteristics of the capturing device.

Several approaches of noise reduction in color signal are known [13, 8, 12]. The most common technique of PPG signal amplification is to average the intensity values of red, green and blue color components independently within each ROI [2, 13]. The other method is a moving average filter, which removes noise by subtracting average signal values in a temporal window from the current signal value. The possible extension of a moving average filter is signal normalization in the temporal window.

Another way to reduce noise and amplify the PPG signal is based on finding correlations and dependences of color signals obtained synchronously from different ROIs [5, 12] or of RGB components of a color signal obtained within a single ROI [8, 13]. In particular, noise reduction was achieved by using a second ICA component as a purified color signal obtained by means of ICA analysis [16] applied to red, green and blue components [13]. One of the most effective approaches is the CHROM signal, which is obtained by weighted summarizing of filtered (with Fourier bandpass filter) mixtures of red, blue and green components, pre-scaled in a temporal window, of the color signal from a ROI [8].

Either the Fourier bandpass filter, Butterworth bandpass filter [2] or Hamming window-based finite impulse response bandpass filter [6] were applied to the signal to remove frequencies outside the relevant range.

C. Heart Rate Calculation

Heartbeat frequency is typically estimated by spectral or peak-detection analysis of the purified PPG signal after denoising. The estimation is typically performed using Fourier transformation of the PPG signal [2, 3]. The maximal power of the Fourier spectrum of filtered color signal is searched for in a certain range (of possible heartbeat frequencies). The frequency that corresponds to this power was considered as the average HR on the color signal. The typical range of frequencies to search in is [0.7, 3] Hz, which covers a normal HR range from 42 bpm to 180 bpm. Several methods [13, 3] also used peak detection in PPG-signal. Such peaks contain information about RR Intervals, which is widely used in medical applications for various diagnostics procedures [1, 17].

III. EXPERIMENTAL SETUP

The efficiency of the proposed Approximation-based method was evaluated on a self-collected dataset. It consists of videos of 60-80 seconds duration, containing the person's face in each frame.

RGB camera Logitech C920 webcam was used in video recording. Frame resolution was 1920 × 1080 and frame rate was 30 maximal frames per second.

8 healthy participants (7 male, 1 female, aged from 24 to 37, all europeoids) were involved in the experiments. From 2 to 14 videos were recorded for each subject. Each subject signed written consent to take part in the tests, which were performed in compliance with the bioethics regulations; experimental protocols were approved by the bioethics committee of the Southern Federal University.

The distance from the face to webcam was in the range of 0.5–0.7 m. The pixel size of the facial area was from 350 × 350 pixels to 550 × 550. Each video was recorded in daylight (300-1000 lux) at 15 frames per second.

The Choicemmed MD300C318 pulse oximeter (with declared accuracy ≈ 1-2 bpm) was used to register the HR values considered as ground truth.

Two types of experiments were conducted: a stationary scenario and a mixed motion scenario.

a) Stationary Scenario: The subject was asked to sit still in front of the webcams looking straight ahead and fixing their pose in order to prevent any movement. 12 videos were recorded for the stationary scenario.

b) Mixed Motion Scenario: In the mixed motion scenario, the subject moved their head looking from right to left (with 120° angle amplitude), from up to down (with 100° angle amplitude), speaking and changing facial expressions. As with the stationary scenario, the subject sat in front of the webcams. 6 videos were recorded for the mixed motion scenario.

IV. METHODS

The following pipeline combined of particular algorithms of color signal extraction, amplification and filtering was evaluated by estimating the accuracy of HR values obtained by applying these algorithms on two groups of video sequences, containing video records of the experiments of two types, the stationary and mixed motion scenarios.

A. Color Signal Extraction

We obtain color signals for red, green, and blue color components from two ROIs by averaging the intensity independently within each ROI in each frame. ROIs positions are defined relative to the facial bounding box obtained by the OpenCV implementation of the Viola–Jones face detector [18] applied to each frame. The last 20 coordinates of detected bounding boxes are averaged in order to minimize detection jitter.

The ROI lies over the central part of the facial area. It is defined as the rectangular area placed in the center of the facial bounding box with height equal to the height of the facial rectangle, and width at 60% of its width. As the ROI
covers a large skin area, it can be assumed that it will produce a clear and representative PPG signal.

B. PPG Signal Amplification

Color signal amplification is based on ICA analysis that was proposed by Poh et al. [13, 4]. The ICA method decomposes red, green, and blue traces obtained within the certain ROI. The second independent source is then used as the amplified signal. Signal amplification algorithm is applied to obtained color signals within a sliding window of 256 frames duration (that corresponds approximately to 17 seconds of video in a case of FPS rate being equal to 15) with 1-frame step. This particular size of sliding window is chosen for three reasons: first, it is assumed that the heart rate does not greatly vary within 17 seconds if the subjects are not doing physical exercise; second, a wider sliding window covers a greater amount of heart beats that may result in higher accuracy of applying methods; third, 256 is a power of two that leads to a faster calculation of Fourier spectrum, a frequently using procedure in our work. The following particular algorithms of PPG signal amplification are considered.

C. PPG Signal Filtering

We apply the PPG signal filtering algorithm to the obtained amplified PPG signals within a sliding window of the same size as at the PPG signal amplification stage (256 values) with a 1-value step.

a) No Filtering (No): No filtering is done; the filtered signal is equal to the obtained amplified signal. We consider pure amplified signal without filtering to compare it with the Approximation-based filtering.

b) Approximation-based Transformation (Appr): We propose a new signal filtering technique, Approximation-based filtering, that reduces the impact of high-amplitude noise and disturbances which could be caused by the subjects movements or changes in illumination. The intuition behind this method is as follows. We want to focus on pulse-caused waves in color signal and neglect linear trends and strong disturbances in color signal. Due to that, the following procedure looks reasonable. In order to equalize amplitudes of informative pulse-caused waves and noise-caused high-amplitude disturbances, we use normalization of color signal (see description below). After that, we perform square polynomial approximation of local pattern of color signal. We assume that duration of such local pattern should be near to the typical duration of pulse wave in color signal. Leading coefficient of the polynomial obtained in such manner characterizes form and strength of pulse wave. Formally, to calculate the Approximation signal, the original signal is firstly averaged within a sliding window:

$$\text{Aver}_i = \frac{1}{N_{\text{Aver}}} \sum_{j=i}^{i+N_{\text{Appr}}} \text{Orig}_j,$$

where $\text{Orig}_j$ is the $j$-th value of the original signal, $\text{Aver}_i$ is the $i$-th value of the averaged signal, $N_{\text{Aver}}$ is the size of an averaging sliding window equal to 5 (one third of a second); this value reduces high-frequency noise from the original signal without influencing the PPG signal. The segment of the averaged signal is then normalized:

$$\text{Norm}_i = \frac{\text{Aver}_i - \frac{1}{N_{\text{Appr}}} \sum_{j=i}^{i+N_{\text{Appr}}} \text{Aver}_j}{\text{stddev}(\text{Aver}_{i-N_{\text{Appr}}}, \cdots, \text{Aver}_{i+N_{\text{Appr}}})},$$

where $\text{stddev}(\text{array})$ is a function that returns the standard deviation of the array, $N_{\text{Appr}}$ is the size of the approximation sliding window equal to 3. This value was selected for the following reasons. Firstly, it cannot be smaller than 3 because the square polynomial approximation can only be constructed taking at least 3 points. Secondly, we have found that setting this value at 3 in our experiments leads to the most clear approximation signal.

The normalized segment is approximated by the square polynomial using the Ordinary Least Squares method. Finally, the inverted value of the leading coefficient of the square polynomial is determined as the Approximation signal value corresponding to the normalized segment of the averaged original signal (Fig. 1):

$$\text{Appr}_i = -\text{LC}_{\text{OLS}}(\text{Norm}_{i-N_{\text{Appr}}}, \cdots, \text{Norm}_{i+N_{\text{Appr}}});$$

where $\text{LC}_{\text{OLS}}(\text{array})$ is a function that returns the value of the leading coefficient of the ordinary least squares approximation of the array.

Fig. 2 illustrates the Approximation signal calculated from Green signal. The Approximation signal amplitude changes slightly in contrast to the Green. This signal filtering method does not filter high-frequency noise less than 4 Hz because of its window size. However, it could enhance the wave peaks of the original signal and suppress motion-caused high-amplitude disturbances.

D. HR Calculation

As HR calculation method, we use Fourier analysis (FA) with slight modifications. The method takes the amplified (and optionally filtered) pattern of color signal with a size of 256 elements and calculates the HR estimate over it. Fourier

![Fig. 1: Approximation signal calculation scheme](image-url)

$$\text{y} = Ax^2 + Bx + C$$
analysis is a classic approach of color signal processing. It is based on Fourier transformation that is applied to the input signal. To reduce noise, we perform per-element averaging of the last 50 Fourier spectra obtained from sequential patterns of the amplified (and filtered) color signal with a size of 256 and step equal to 1. The maximum amplitude value is searched for in the averaged spectrum within the frequency domain limited by the expectable HR range: from 45 bpm to 120 bpm (Fig. 3a). The result is the HR value that corresponds to the maximum amplitude. We evaluated an extension of Fourier analysis that is commonly used in signal processing but not typical in rPPG works. FA' is similar to the classical FA, except that before finding an average, each Fourier spectrum is smoothed by Gaussian with window size equal to 30 bpm and sigma equal to 5 bpm (Fig. 3b). The spectrum obtained by Extended Fourier Analysis is smoothed expecting to lead to more robust HR estimates. The subject’s HR could vary in short-term periods. It may cause the presence of multiple low-amplitude peaks in the Fourier spectrum instead of one high-amplitude peak (see Fig. 3a where small peak was occurred near the main peak). Spectrum smoothing allows aggregating several low-amplitude peaks into one high peak (Fig. 3b).

E. HR Estimates Filtering

After calculating HR values, we filter it by averaging \( \text{Aver} \) them within a sliding window with size \( N_{\text{filt}} \) equal to 2 \( \cdot \) FPS (which corresponds to 2 seconds); the sliding window moves with step equal to 1:

\[
\text{HRAver}_i = \frac{1}{N_{\text{filt}}} \sum_{j=i}^{i+\left\lfloor \frac{N_{\text{filt}}}{2} \right\rfloor} \text{HR}_j,
\]

where \( \text{HR}_j \) is the \( j \)-th value of estimated HR values, \( \text{HRAver}_i \) is the \( i \)-th value of HR values filtered by Averaging.

V. RESULTS

In this section, we present and evaluate the results for the methods described in Sect. IV, applied to the dataset collected as described in Sect. III: pipeline \( P1 \), ICA-No-FA'-Aver (without signal amplification method), and pipeline \( P2 \), ICA-Appr-FA'-Aver (with Approximation-based filtering as signal amplification method).

<table>
<thead>
<tr>
<th>Pipeline</th>
<th>Average RMSE (bpm)</th>
<th>RMSE standard deviation (bpm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P1 )</td>
<td>15.8</td>
<td>12.8</td>
</tr>
<tr>
<td>( P2 )</td>
<td>3.0</td>
<td>3.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pipeline</th>
<th>Average RMSE (bpm)</th>
<th>RMSE standard deviation (bpm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P1 )</td>
<td>26.8</td>
<td>15.0</td>
</tr>
<tr>
<td>( P2 )</td>
<td>16.2</td>
<td>14.5</td>
</tr>
</tbody>
</table>

Tables I, II illustrate that average RMSE for the pipeline \( P2 \) is significantly lower than for pipeline \( P1 \). However, average RMSE in mixed motion scenario is still high for both pipelines. High error value can be explained by presence of outliers among videos where RMSEs were much higher than average RMSE value.

VI. CONCLUSION

In this paper, we proposed a new preprocessing method for rPPG pipeline to enhance heart rate estimation. The method amplifies PPG signal resulting in better accuracy in case of stationary scenario when subject is asked sit motionless. However, even slight non-special motions during an rPPG experiment may cause high-amplitude disturbances of color signal. Approximation-based transformation suppresses such disturbances.

According to the obtained results, subject’s motion affects reliability of the considered methods. In the case of the stationary scenario, pipeline with Approximation-based transformation generate HR estimates that are close to real values. However, in a case of mixed motion, the same pipeline produce inaccurate estimates. One possible way to avoid this problem when using rPPG methods in real-life application is to detect patterns of motion (for example, by the optical flow estimation algorithms) and to exclude video fragments that contain these patterns from analysis. We believe that this technique merits further research.

ACKNOWLEDGMENT

The work is supported by the Russian Ministry for Education and Science, project no. 2.955.2017/4.6 “Development of the hardware and software system for monitoring the attention level and psychoemotional state of pilots and dispatching personnel to improve flight safety”.
REFERENCES


Development Of The Filtering Algorithm For Doubly Stochastic Images Based On Models With Multiple Roots Of Characteristic Equations

Nikita Andriyanov  
Telecommunications Department  
Ulyanovsk State Technical University  
Ulyanovsk, Russia  
nikita-and-nov@mail.ru

Vitaly Dement'ev  
Telecommunications Department  
Ulyanovsk State Technical University  
Ulyanovsk, Russia  
dve@ulntc.ru

Konstantin Vasiliev  
Telecommunications Department  
Ulyanovsk State Technical University  
Ulyanovsk, Russia  
vkk@ulstu.ru

Abstract— A filtering algorithm is developed for doubly stochastic autoregressive random fields based on the application of models with multiple roots of characteristic equations. Cross sections of the correlation functions of such models with an average parameter are obtained, the variances of filtering errors.

Keywords— doubly stochastic models, image modeling, image processing, filtering, model with multiple roots.

I. INTRODUCTION

Recently, satellite systems for recording areas of the Earth's surface have been widely used. This is due to the wide possibilities that remote sensing of the Earth (RS) provides. Indeed, satellite data has a number of significant advantages over conventional cartographic information. The key ones are the accuracy and efficiency of remote sensing data, as well as the possibility of monitoring in different spectral ranges. Therefore, space images are the best means of operational control and monitoring. Unlike maps, they allow you to quickly identify new phenomena and their development processes.

One of the key features of satellite imagery is their pronounced spatial heterogeneity. It is associated with the variety of shapes and textures of different objects observed from space. Indeed, any satellite image of the Earth's surface contains images of different objects, for example, rivers, forests, urban buildings, agricultural lands, etc. The visual characteristics of these objects are significantly different. Attempt to describe the whole picture with the help of known homogeneous models (for example, autoregressive, wave, etc.) leads to errors caused by incorrect averaging of information. In this regard, to describe satellite images, it is necessary to apply more complex mathematical models capable of simulating inhomogeneous random fields (RF). In the present work, a study is made of doubly stochastic autoregressive RF models [3-8] that can serve as an adequate mechanism for describing satellite heterogeneous images.

II. MODEL WITH MULTIPLE ROOTS AND ITS PROPERTIES

Consider the following formula for autoregressive (AR) models of different multiplicities:

\[ x_{i,j} = \beta_{\alpha} z_{i,j} - \sum_{l=0}^{N_1} \sum_{j=0}^{N_2} \alpha_{l,j} x_{i-l,j-j} \]  

(1)

where \( N_1 \) and \( N_2 \) characterize the multiplicity of the model; coefficients \( \alpha_{l,j} (\alpha_{0,0} = 0) \) are products of the corresponding coefficients of one-dimensional AR along the axes \( x \) and \( y \):

\[ \alpha_{l,j} = \alpha_{l} \alpha_{j} \]  

(2)

The coefficients of one-dimensional AR (2) can be obtained from expression

\[ \alpha_{l} (\rho_{s}, N_{l}) = (-1)^{l} \frac{n!}{m!(n-m)!} C_{n}^{m} \rho_{s}^{n-m} \]  

(3)

where \( C_{n}^{m} = \frac{n!}{m!(n-m)!} \) is number of combinations from \( n \) to \( m \); \( \rho_{s}, \rho_{y} \) are model parameters.

Finally, the coefficient \( \beta \) of two-dimensional model is the normalized product of the corresponding coefficients of one-dimensional AR along the axes \( x \) and \( y \):

\[ \beta = \frac{\sigma_{\beta}}{\sigma_{\alpha}} \beta_{x} \beta_{y} \]  

(4)

And these coefficients can be found by the following formulas

\[ \beta_{x} = \frac{1}{\sum_{j=0}^{N_{2}+1} (C_{N_{2}+1}^{j} \rho_{y}^{j})^{2}} \]  

(5)
In order to obtain covariance functions (CF) of models of arbitrary orders, one can use the expressions for one-dimensional QF AP with multiple roots of the characteristic equations

\[ B_x(k) = \sigma_i^2 \sum_{l=0}^{m-1} g(m, l, k) \rho^{2(m-l-1)} \frac{1}{(1 - \rho^2)^{\frac{1}{2}l^{-1}-1}}, \]

where \( g(m, l, k) = \frac{(m + k - 1)! (2m - l - 2)!}{l! (m - 1)! (m - l - 1)! (m + k - l - 1)!} \). The variance of independent random variables \( \xi_i, i = 1, 2, ..., n \), we find from a given variance \( B_x(0) = \sigma_i^2 \):

\[ \sigma_i^2 = \sigma_x^2 (1 - \rho^2)^{2m-1} \sum_{l=0}^{m-1} (C_{m-l}^l \rho^l)^2. \]

Thus, for a spatial AR with characteristic roots of multiplicity \((m_1, m_2)\) the expression for the CF can be written as follows:

\[ B_{ij}(k, l) = \sigma_i^2 \sum_{m=0}^{m-1} g(m_i, l, k) \rho_i \frac{2(m_i-l-1)}{(1 - \rho_i^2)^{\frac{1}{2}l^{-1}-1}} \sum_{m=0}^{m-1} g(m_j, l, k) \rho_j \frac{2(m_j-l-1)}{(1 - \rho_j^2)^{\frac{1}{2}l^{-1}-1}} \]

Fig. 1 shows the CF cross sections for the models of the 1st (solid line) and the 2nd (dashed line) orders with the correlation radius \( k_0 = 15 \). Fig. 2 shows cross sections for models of different orders with the same value of the parameter \( \rho = 0.8615 \).

As can be seen from Fig. 1, the second order CF model has a "bell-shaped" vertex, and the correlation links between the RF elements generated by such models are stronger within the correlation radius.

Fig. 2 shows that the CF cross sections with increasing multiplicities AR tend to ellipsoids. However, with a significant increase in the orders, the CF decreases much more slowly than when using the first and second order AR.

Thus, the correlation properties of AR with multiple roots of the characteristic equations ensuring equal correlation radii are investigated. Dependences of the correlation parameter of such AR on the correlation radius are obtained.

III. TWO-DIMENSIONAL DOUBLE STOCHASTIC RANDOM FIELD MODEL

Let us consider the case where a two-dimensional image can be described by a \( \{x_{ij}\} \), given on a rectangular grid \( \{(i, j); i = 1, M_i; j = 1, M_j\} \), where \( i, j \) are spatial coordinates. Consider the space AR model

\[ x_{ij} = \sum_{i, j D_{ij}} \sum \rho_{ij} x_{ij} \], where \( D_{ij} \) is a region of local states containing elements \( \{x_{ij}\} \), defining \( x_{ij} \). Suppose that the coefficients \( \rho_{ij} \) are not permanent, but random. Let these RV also be determined in a similar way \( \rho_{ij} = \sum \sum r_{ij} \rho_{ij} \). In the last expression \( r_{ij} \) are constant coefficients that determine the correlation characteristics of the auxiliary RF \( \{\rho_{ij}\} \).

The mathematical model defined in this way is nonlinear, and its direct analysis is associated with considerable mathematical difficulties. Therefore, in order to reduce the material presented below and simplify its understanding, we use a simpler version of the doubly stochastic model:
\[ x_{ij} = \rho_{xj}x_{i-1,j} + \rho_{yj}x_{i,j-1} - \rho_{xj}\rho_{yj}x_{i-1,j-1} + \xi_{ij} \]  

where \( x_{ij} \) is simulated RF with normal distribution 
\[ M\{x_{ij}\} = 0 , \quad M\{x_{ij}^2\} = \sigma^2 \]  
and \( \xi_{ij} \) is RF of independent Gaussian RV with 
\[ M\{\xi_{ij}\} = 0 , \quad M\{\xi_{ij}^2\} = \sigma^2 = 1 \]  
\( \rho_{xj} \) and \( \rho_{yj} \) are coefficients of correlation in a row and in a column, respectively, at the point \((i,j)\).

RV \( \rho_{xj} \) and \( \rho_{yj} \) with a Gaussian probability distribution density can be described by the following autoregressive equations:
\[
\tilde{\rho}_{xj} = \tilde{\rho}_{xj}(t-j-1) + r_x(t) - r_x(t-j)\tilde{\rho}_{xj}(t-j) + \sigma_x \sqrt{(1-r_x^2)(1-r_y^2)}\xi_{ij},
\]
\[
\tilde{\rho}_{yj} = \tilde{\rho}_{yj}(t-j-1) + r_y(t) - r_y(t-j)\tilde{\rho}_{yj}(t-j) + \sigma_y \sqrt{(1-r_x^2)(1-r_y^2)}\xi_{ij},
\]

where \( r_x = M\{\tilde{\rho}_{xj}\tilde{\rho}_{x(j-1)}\} \) , \( r_x = M\{\tilde{\rho}_{xj}\tilde{\rho}_{x(j-1)}\} \) are correlation coefficients of a random parameter \( \tilde{\rho}_{xj} \) ;
\[
\tilde{\rho}_{yj} = \tilde{\rho}_{yj}(t-j-1) + r_y(t) - r_y(t-j)\tilde{\rho}_{yj}(t-j) + \sigma_y \sqrt{(1-r_x^2)(1-r_y^2)}\xi_{ij},
\]
and \( \xi_{ij} \) are normally distributed RV with 
\[ M\{\xi_{ij}\} = M\{\xi_{ij}^2\} = 0 , \quad M\{\xi_{ij}^2\} = \sigma^2 = 1. \]

Let the available observations be a mixture of the useful signal and the additive white Gaussian noise \( \{n_{ij}\} \) with 
\[ m_n = 0 \]  
and \( \sigma_n^2 \) such as follows \( z_{ij} = x_{ij} + n_{ij} \).

Under the proposed conditions, we will try to construct the best algorithm for filtering the SP in terms of the dispersion variance minimum of the estimation.

IV. FILTER FOR A DOUBLY STOCHASTIC FIELD BASED ON A MODEL WITH MULTIPLE ROOTS

Instead of the model (10), it is better to use a two-dimensional AP model with multiple roots of the characteristic equations, since it provides smooth fields:
\[
x_{ij} = 2\rho_{xj}x_{i-1,j} + 2\rho_{yj}x_{i,j-1} - 4\rho_{xj}\rho_{yj}x_{i-1,j-1} - \rho_{xj}\rho_{yj}x_{i-2,j} - \rho_{xj}\rho_{yj}x_{i-2,j-1} + 2\rho_{xj}\rho_{yj}x_{i-2,j-1} + 2\rho_{yj}\rho_{yj}x_{i,j-2} - \rho_{xj}\rho_{yj}x_{i-2,j-2} + b_i\xi_{ij}
\]

where \( \xi_{ij} \) is RF of independent Gaussian RV with 
\[ M\{\xi_{ij}\} = 0 , \quad M\{\xi_{ij}^2\} = \sigma^2 = 1 \]  
\( \rho_{xj} \) and \( \rho_{yj} \) are RV, determined by the AR equations (10).

In this case, the region of local states for each of the \( x_{ij} \) is a set of eight points, and for sequential filtering it is required to use \( 2M_1 + 3 \) elements. Fig. 3 shows the logic of the algorithm.

![Fig. 3. The region of local states and support elements for the AR model with the roots of the characteristic equation of multiplicity (2,2)](image)

If we compose the following vector of elements 
\[ \hat{x}_{ij} = (x_{ij}, \hat{\rho}_{xj}, \hat{\rho}_{yj})^T \]  
with length \( 4M_1 + 5 \) , where 
\[ \hat{x}_{ij} = (x_{ij}, x_{i-1,j}, ..., x_{i-1,M_1}, ..., x_{i-1,j-1}, x_{i-2,j-2}) \]

\[ \hat{\rho}_{xj} = (\rho_{xj}, \rho_{xj-1}, ..., \rho_{xj-1,M_1}, ..., \rho_{xj-1-j}) \]

\[ \hat{\rho}_{yj} = (\rho_{yj}, \rho_{yj-1}, ..., \rho_{yj-1,M_1}, ..., \rho_{yj-1-j}) \]

and write model (11) as 
\[ \hat{x}_{ij} = \rho_{ij}(\hat{x}_{ij-1}) + \xi_{ij}, \]  
then the process of filtering the image corresponding to such a model will be identical to the nonlinear filter [5], taking into account the corresponding changes in the matrix transformation \( \phi_{ij}(\hat{x}_{ij-1}) \).

Obviously, the considered variant of vector filtration can also be refined due to the use of other mathematical models for correlation parameters or basic RF. Model parameters will define the transformation \( \phi_{ij}(\tilde{x}_{ij-1}) \) and the size of the estimated vector \( \hat{X}_{ij} \), and the structure of the nonlinear filter is preserved. Varying the complexity of these models, one can obtain the necessary estimation accuracy.

To further refine the results obtained, it is possible to use the remaining observations within the interpolation procedure. To this end, we will determine and store, during the forward filtering process, the matrices \( A_j = P_{ij}P_{ij}^{-1}P_{j+1}^{-1} \) and the extrapolated estimate vector \( \hat{x}_{M_1} \). Upon completion of the forward motion of the filter, we define the interpolated evaluation vector in the lower right-hand point of the image as 
\[ \hat{x}_{M_1,M_2} = \hat{x}_{M_1,M_2} \]  
Indeed, the estimate at the point \( (M_1, M_2) \) is based on the processing of all points of the original image and, in this connection, can be considered as interpolated. Let us find the interpolated estimates at the previous point with the coordinates \( (M_1, M_2 - 1) \). Then
Continuing the return stroke of the filter to a point with an arbitrary coordinate \((i,j)\), we finally write down the expression for the interpolated estimate \(\hat{x}_{ij}^\lambda = \hat{x}_{ij} + A_j(\hat{x}_{ij}^\lambda - \hat{x}_{ij+1}^\lambda)\). \(\text{(13)}\)

As before, it should be taken into account that the preceding one for the vector \(\hat{x}_{i1}^\lambda\) is a vector \(\hat{x}_{i-1M_1}^\lambda\).

The proposed nonlinear filtering algorithms can be applied to real images for both noise suppression and for creating a field of correlation parameters of the image. Fig. 4 shows the results for the three satellite images.

![Fig. 4. Filtering images. From left to right: a real image, a noisy image, its evaluation, an estimation of correlation parameters on a line, an estimation of correlation parameters by a column.](image)

It should be noted that for assessing each next pixel of the image, a region of local states was used such that the previous scan was determined in Fig. 3. Moreover, simultaneous estimation of the model parameters is carried out, which are subsequently used to estimate the luminance signal.

Fig. 5 shows the dependences for the dispersion error of the first image in Fig. 4 on the noise variance, namely: A1 is Kalman filter without interpolation; A2 is discrete Wiener filter, A3 is Kalman filter with interpolation, A4 is non-linear recurrent filter based on model (11) with interpolation.

![Fig. 5 Satellite image filtering efficiency](image)

Note that due to the peculiarities of matrix operations, the solutions found require considerable computational costs. To illustrate, we present Table 1, obtained from the processing of the satellite image 256x256 shown in Fig. 4. The calculation was performed on an Intel Core i7 computer, 2.7 Ghz, 8GB of RAM in MatLab 2014. A preliminary estimation of the parameters was performed for all images.

<table>
<thead>
<tr>
<th>Method of processing</th>
<th>The accuracy of the variance estimation</th>
<th>Processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pointed Kalman filter. Estimation by the autoregressive model with multiple roots of the characteristic equation. The multiplicity is (1.1).</td>
<td>0.4981</td>
<td>0.3 seconds</td>
</tr>
<tr>
<td>Pointed Kalman filter. Estimation by the autoregressive model with multiple roots of the characteristic equation. The multiplicity is (2.2).</td>
<td>0.4745</td>
<td>0.5 seconds</td>
</tr>
<tr>
<td>Point Kalman filter with interpolation. Estimation by the autoregressive model with multiple roots of the characteristic equation. The multiplicity is (2.2).</td>
<td>0.4698</td>
<td>1.8 seconds</td>
</tr>
<tr>
<td>Discrete Wiener filtering</td>
<td>0.4978</td>
<td>47 seconds</td>
</tr>
<tr>
<td>Nonlinear vector filter. Estimation by a combination of autoregressive models with multiple roots of the characteristic equation. The multiplicity is (1.1).</td>
<td>0.1359</td>
<td>31 seconds</td>
</tr>
<tr>
<td>Nonlinear vector filter. Estimation by a combination of autoregressive models with multiple roots of the characteristic equation. The multiplicity is (2.2) for the main simulated field; the multiplicity is (1.1) for correlation parameter fields</td>
<td>0.1289</td>
<td>2 minutes 17 seconds</td>
</tr>
<tr>
<td>Nonlinear vector filter with interpolation. Estimation by a combination of autoregressive models with multiple roots of the characteristic equation. The multiplicity is (2.2) for the main simulated field; the multiplicity is (1.1) for correlation parameter fields</td>
<td>0.1254</td>
<td>1 hour 13 minutes 17 seconds</td>
</tr>
</tbody>
</table>

Copyright © 2018 by CENPARMI, Concordia University
The analysis of the presented table allows to draw a conclusion about the best results of filtration for a nonlinear vector filter with interpolation. This is due to the fact that in such a filter the possible heterogeneity of the image is taken into account, and the nature of the CF. The reverse side of this result is the considerable computational resources required for its implementation. The processing time of the image by the specified filter exceeds by several orders of magnitude the time for other filters. This is due to the need to address the matrix of extrapolation errors.

V. CONCLUSION

A filter is proposed that provides 3-4 times due to application of doubly stochastic models based on AR with multiple roots. For the latter, studies of CF cross sections have been performed, showing the possibility of describing smooth images with their help.

ACKNOWLEDGMENT

The study was supported by RFBR, project 17-01-00179 A.

REFERENCES

The solution of the problem of simplifying the images for the subsequent minimization of the image bit depth

Evgenii Semenishchev  
Don State Technical University  
Rostov-on-Don, Russia  
sea.sea@mail.ru  

Viacheslav Voronin  
Don State Technical University  
Rostov-on-Don, Russia  
voroninisl@mail.ru  

Igor Shraifel  
Don State Technical University  
Rostov-on-Don, Russia  
shraifel17@mail.ru  

Dmitrii Chernishov  
Don State Technical University  
Rostov-on-Don, Russia  
dimcher@inbox.ru  

Abstract—In this paper, an algorithm for adaptive changing the brightness of image pixels is proposed. To perform the operation of changing the bit depth of the image, it is suggested to minimize the error on three parameters. As a result of the algorithm, the problem of finding the minimum of a certain numerical characteristic of the partition is performed - the number of clusters, the minimum size, and the sum of the maximums of the cluster size. The smaller the value of any of them, the better will be the division of areas in the image. In the paper, a mathematical description of the algorithm is given, and also block diagrams of the interrelationships of the data processing steps are presented. Recommendations on the choice of parameters of the algorithm for optimal minimization of parameters are presented.

Keywords—images, bit depth, simplifying images, selection of areas, the replace brightness’s.

I. INTRODUCTION

Analysis of streaming video, as well as big data of requires the initial preparation. Performing the preliminary image analysis operation, as well as searching for objects on them, will be performed faster in the case of a reduction in the dimension of the frame. To reduce the depth of data (bits), the operation of changing the bit depth is suggested in the work. The change in bit depth is not done by combining the data of the nearest values, but by applying the procedure for forming blocks and averaging the values in them. The application of this approach allows us to introduce maximum changes in stationary areas and leave unchanged areas of interest to user.

The application of the bit change operation, it is possible: to the image area, the block near a certain section of the histogram or the entire image. The application of this approach is relevant when combining data obtained in different electromagnetic ranges. Also when forming a border layer on a low-detailed image. An example is the integration of IR and optical image data. In this task it is necessary to perform the operation of reducing the bit depth of data to the level of the IR sensor.

This direction is relevant for solving the problem of combining data obtained by sensors operating in different electromagnetic ranges [1]. An example is a problem of obtaining an image with the boundaries of objects obtained by a video camera [2]. Obtaining the boundaries of objects is done by analyzing the data obtained by the optical camera. When obtaining the boundaries of objects obtained by IR and optical sensors in conditions of poor visibility [3]. When solving the problem of obtaining depth maps for a set of data obtained from a group of cameras [4]. With the preliminary simplification of the data, the task is used to improve the performance of systems operating with big data [5]. To preliminary analysis in the case of solving the problem of stitching images into a single composition [6] and others problem.

II. STATEMENT OF THE PROBLEM

Let some (first) image be divided into small fragments of the $K_1, K_2, \ldots, K_n$, and the $i$-th fragment consists of $l_i$ pixels. Next, the operation of inducing the brightness of its fragments by the first image will be performed. In this paper we will call them - big pixels. For the case of $l_i = 1$, the large pixel $K_i$ is an ordinary pixel. We believe that all the pixels in the image are equal. The brightness of a large pixel $K_i$, it is advisable to calculate as the arithmetic mean of its usual pixels, $i = 1, 2, \ldots, n$. The image thus obtained will be called the second image, and the set of its larger pixels will be denoted by the symbol $\gamma_i$. We put the problem of partitioning this set into clusters, in order to give all the large pixels of each cluster $T_j$ the same new lumiance value. The averaging of their brightness is performed by one of the following two formulas:

$$\beta_j = \frac{\max_{K_i \in T_j} a_i + \min_{K_i \in T_j} a_i}{2}, \quad (1)$$

$$\gamma_j = \frac{\sum_{K_i \in T_j} a_i l_i}{\sum_{K_i \in T_j} l_i} \quad (2)$$

In the future, we will shorten the phrase "splitting into clusters" to one word "splitting". If as a result of the method, all $n$ of large pixels are replaced by the brightnesses $\beta_j$ in the first case or $\gamma_j$ in the second, two more images are obtained, which are called the third and fourth, respectively. Obviously, the necessary condition for the qualitativeness of the above
partition is the possibility of enumerating clusters in such a way that the inequalities

\[ \min_{\mathbf{K} \in T_j} a_j \geq \max_{\mathbf{K} \in T_{j-1}} a_j, \quad j = 1, 2, ..., m - 1; \quad (3) \]

here: \( m \) – is the number of clusters, \( T_j \) – is the cluster with the number \( j \). Further by partitioning the set \( \mathcal{R} \) into clusters, we mean those decompositions that correspond to the requirement (3). We believe that for any decomposition the implication is valid

\[ 1 \leq \tau_i < \tau_j \leq n, \quad a_\tau = a_j, \quad K_\tau \in T_p, K_j \in T_q \Rightarrow p \leq q \quad (4) \]

In paper we propose an algorithm for partitioning \( \mathcal{R} \). Its use provides a predetermined degree of proximity of the second and third or (and) second and fourth images. Replacing the first image with the subsequent one allows significantly to reduce the processing time. We solve the problem of choosing between bit depth and image quality.

III. NOTATION AND DEFINITIONS

Let \( x = (x_1, x_2, ..., x_m) \) – be any vector with real coordinates. We arrange the values of the vector in decreasing order, we obtain a series of numbers \( x_1 \geq x_2 \geq ... \geq x_m \) and, respectively, the vector \( \tilde{x} = (\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_m) \) of the same dimension \( m \).

The calculation procedure of the algorithm presented in the work begins with the construction of the brightness vector of all large pixels \( \tilde{a} = (\tilde{a}_1, \tilde{a}_2, ..., \tilde{a}_m) \), which are ordered by in descending order. For any pair of large pixels \( K_\tau, K_j (\tau < j) \) with the same luminosity, we assume that in the coordinate recording of the vector \( \tilde{a} \), the brightness \( K_j \) precedes the brightness \( K_\tau \). This requirement, together with the monotonicity of the sequence \( \tilde{a} \), will ensure, by (3), (4), the one-to-one correspondence between all partitions of the set \( \mathcal{R} \) and all possible separations of the vector \( a \) on subvectors (vectors of the form \( (\tilde{a}_1, \tilde{a}_{i+1}, ..., \tilde{a}_m) \) when \( 1 \leq \tau_i < j \leq n \). To solve the problem, it is convenient to use formulas containing the function \( \text{sign}(x) \). We use the notation: \( r_j \) – is the number of coordinates of the vector \( a = (a_1, a_2, ..., a_\tau) \), large \( j \)-th coordinates or equal to it; \( j \)– is the number of the \( j \)-th coordinate of the vector \( a \), after the ordering of the coordinates \( a \) in descending order. We use relations

\[ r_j = \sum_{\tau=1}^{\tau_i} \text{sign}(1 + \text{sign}(a_j - a_j)), \quad j = 1, 2, ..., n. \]

By the requirement of ordering, the numbers \( \eta_1, \eta_2, ..., \eta_{\tau_i} \) are uniquely determined and can be found by formulas

\[ \eta_j = 1 + r_j - \sum_{\tau=1}^{\tau_i} (1 - (\text{sign}(r_1 - r_1))^2 \text{sign}(1 + + \text{sign}(i - j)), \quad j = 1, 2, ..., n. \]

Accept the expression \( \tilde{a}_j = \sum_{\tau=1}^{\tau_i} a_\tau (1 - (\text{sign}(\eta_\tau - j))^2), \quad j = 1, ..., n \).

We obtain the desired vector \( \tilde{a} = (\tilde{a}_1, \tilde{a}_2, ..., \tilde{a}_n) \). Next, we carry out the relocation operation \( \Theta_1, \Theta_2, ..., \Theta_n \) of the elements of the set \{1, 2, ..., n\} which corresponds to the change relocation \( n_1, n_2, ..., n_n \). We find this relocation by taking \( \Theta_1 = i \) for each \( 1 \leq i \leq n \). Any partition of the set \( \mathcal{R} \) into \( m \) clusters has the form \( T_j = (\Theta_1, \Theta_2, ..., \Theta_n) \) \( 1 \leq j \leq m \) which allows you to specify it \( n_1, n_2, ..., n_m \) (when \( n_0 = 0 < n_1 < n_2 < ... < n_m = n \)). We call the cluster size \( T_j \) the difference \( \Delta_j \) the between distance it’s largest and the smallest brightness of its large pixels \( \Delta_j = \tilde{a}_{n_{j+1}} - \tilde{a}_{n_{j}} = 1, 2, ..., m \). The maximum \( \Delta \) from the distances of the clusters will be called the minimal step of the partition: \( \Delta = \max_{1 \leq j \leq m} \Delta_j \).

Fix an arbitrary nonnegative number \( \varepsilon \). We denote by \( m(\varepsilon) \) the smallest number of clusters onto which we can partition the set \( \mathcal{R} \) under the condition that the smallness of the dimension of the resulting partition does not exceed \( \varepsilon \). We introduce the value \( \Delta(\varepsilon) \) the smallest possible size of partitioning the set \( \mathcal{R} \) into \( m(\varepsilon) \) clusters (obviously, \( \Delta(\varepsilon) \leq \varepsilon \)). Let us introduce \( \sigma(\varepsilon) \) the smallest possible sum of cluster sizes among all partitions of \( \mathcal{R} \) into \( m(\varepsilon) \) clusters with minimum size \( \Delta(\varepsilon) \). This is true, since there exists a partition of the set \( \mathcal{R} \) whose dimension does not exceed \( \varepsilon \). Under the condition that the set of all partitions of \( \mathcal{R} \) is finite.

In this paper, we solve the problem of finding a partition of the set \( \mathcal{R} \) into \( m(\varepsilon) \) of minimal clusters \( \Delta(\varepsilon) \) with the sum of the ranges of the clusters \( \sigma(\varepsilon) \). The triple minimization algorithm will be mathematically described and justified. It consists of three stages, each of which minimizes a certain numerical characteristic - the number of clusters, the minimum and the sum of the dimensions of the clusters. At least any of them is an acceptable result. All three minimizations are performed for \( \varepsilon \leq \varepsilon \) and each subsequent minimization preserves the result of the previous minimization.

IV. MATHEMATICAL JUSTIFICATION OF THE ALGORITHM

For any nonnegative number \( \delta \), we define \( \delta \) the partition of the set \( \mathcal{R} \) as follows. Since the sequences increasing \( \tilde{a}_1, \tilde{a}_2, ..., \tilde{a}_n \), there is a unique number \( 1 \leq n_1 \leq n \) such that \( \tilde{a}_1 - \tilde{a}_{n_1} < \delta < \tilde{a}_1 - \tilde{a}_{n_{m+1}} \) (for convenience we will assume that \( \tilde{a}_{n_{m+1}} = -\infty \)). Suppose that the next number \( n_1 \) is found, where \( j \geq 1 \). If \( n_j = n \) the required \( \delta \) partition \( (n_1, n_2, ..., n_{m-1}) \) is already constructed; it remains to take...
In the case \( n_j < n \), we take as \( n_{j+1} \) the only number corresponding to the condition \( \tilde{a}_{n_j+1} - \tilde{a}_{n_j} \leq \delta < \tilde{a}_{n_{j+1}} - \tilde{a}_{n_{j+1}+1} \). After a finite number of steps, we obtain a \( \delta \) break down of \((n_1, n_2, ..., n_{m-1})\), and with it the number of its clusters \( m \).

The algorithm that realizes the execution of partitioning into local areas is realized in three steps.

**I. Calculating the value \( m(\varepsilon) \).** We construct an \( \varepsilon \)- partition of the set \( \mathcal{R} \) and fix the number of its clusters. This value coincides with the value of \( m(\varepsilon) \), at the first stage of the algorithm.

**II. Calculating the value \( \Delta(\varepsilon) \).** The algorithm is derived from the construction of the 0-partition, the \( \varepsilon \)-partition of the set \( \mathcal{R} \), and the calculation of the number of clusters of each of them. The values of \( m(0) \) and \( m(\varepsilon) \) are equal. The minimum of the 0-partition and the size of all its clusters are zero, that is, they assume the values that are minimal possible for these values. Hence, for \( m(0) = m(\varepsilon) \), the equalities \( \Delta(\varepsilon) = \sigma(\varepsilon) = 0 \), holds, that is, 0-partition is a finite result of applying the algorithm. For condition \( m(0) \neq m(\varepsilon) \), \( \varepsilon > 0 \) and \( m(0) = m(\varepsilon) \). Having found the minimum of the \( \varepsilon \)- partition \( \mathcal{R} \), we assign its value to the variable \( \Theta \).

We denote by \( \rho \) the smallest of the numbers \((\tilde{a}_i - \tilde{a}_{i+1})\), where \( 1 \leq i \leq n - 1 \), \( \tilde{a}_i > \tilde{a}_{i+1} \) (the inequality \( \Theta > 0 \) implies the existence of at least one number of the specified type). Obviously \( \rho > 0 \). The algorithm for determining the magnitude of the \( \Delta(\varepsilon) \) is shown in Figure 1.

The algorithm shown in Figure 1 is implemented in the following steps:

1. Assigning \( a := 0; b := \Theta; \)
2. Assigning \( c := (a + b) / 2; \)
3. The construction of a \( c \)-partition and the assignment of the variable \( \Theta \) to its minimum value;
4. When \( m(c) = m(\varepsilon) \) assigning \( c = \Theta; b := c \). If \( m(c) = m(\varepsilon) \) assigning \( a := c \).
5. If the inequality is true \( b - a \leq \rho \),

Assigning \( \Delta(\varepsilon) := b \), otherwise – return to step 2.

**III. The construction of a partition of the set \( \mathcal{R} \) into \( m(\varepsilon) \) clusters of minimal dimension \( \Delta(\varepsilon) \), with the sum of the cluster sizes \( \sigma(\varepsilon) \).** Let \( m = m(\varepsilon) \) and \( \Delta = \Delta(\varepsilon) \) – the values found in the first two stages of the algorithm; \((n_1, n_2, ..., n_{m-1})\) \( \Delta \)-partition of the set \( \mathcal{R} \) into \( m \) clusters. The third minimization is realized by the recurrence relation method. Fix arbitrary numbers \( 1 \leq \rho \leq n \), \( 1 \leq M \leq m \) and take the notation \( \mathcal{R}_\rho = [K_{\Theta_1}, K_{\Theta_2}, ..., K_{\Theta_j}] \). The \( \Delta \)-partition clusters of the set \( \mathcal{R}_\rho \) coincide with the \( \Delta \)-partition the clusters of the set \( \mathcal{R} \). Only the last cluster of the partition \( \mathcal{R}_\rho \) can be part of a cluster of partition \( \mathcal{R} \) with the same number. There is a unique number \( 1 \leq j \leq m \) such that \( n_{j+1} + 1 \leq \rho \leq n_j \). We replace \( \mathcal{R}, \delta, m \) by \( \mathcal{R}, \Delta, M \) respectively. In this case, we replace \( m(\Delta) \) by \( j \). As a result, we come to the assertion about the possibility of dividing the set \( \mathcal{R}_\rho \) into \( M \) clusters of size \( \leq \Delta \) if and only if \( j \leq M \leq \rho \).

The equivalence of this condition for the double inequality \( M \leq \rho \leq n_M \) we shall assume that it is satisfied. The number \( 1 \leq q \leq \rho \) is said to be admissible if there exists a decomposition of \( \mathcal{R}_\rho \) into \( M \) clusters of minimal dimension not exceeding \( \Delta \), with \( M \)-the cluster \([K_{\Theta_1}, K_{\Theta_2}, ..., K_{\Theta_j}] \). For \( q \) to be admissible, it is necessary and sufficient that: 1) the set \( \mathcal{R}_{\rho-1} \) can be partitioned into \( M - 1 \) clusters \( \leq \Delta \), and 2) the range \( \Delta_M \) of the \( M \) cluster does not exceed \( \Delta \). Similarly, it is possible to prove the equivalence of condition (1) to the double inequality \( M - 1 \leq q - 1 \leq n_{M-1} \Leftrightarrow M \leq q \leq n_{M-1} + 1 \). As \( q \) increases from \( M \) to \( \rho \), the quantity \( \Delta_M = \tilde{a}_q - \tilde{a}_\rho \) decreases.

In the case of \( q = \rho \), \( \Delta_M = 0 < \Delta \). Consequently, among the values of \( q \) under consideration there is the smallest \( Q \) for which the inequality \( \Delta_M \leq \Delta \). The condition (2) will
correspond only to numbers \( Q \leq q \leq \rho \). In the case of the inequality \( M < q \leq \rho \), inequality \( M = q \) is satisfied by all values \( M \leq q \leq \rho \) and \( Q = M \). If \( M = q \leq \rho \), inequality \( M \leq q \leq \rho \) is unique solution of \( q \).\[ \widetilde{a}_q - \widetilde{a}_\rho \leq \Delta \]\[ a_q - a_{q-1} > \Delta \Rightarrow \widetilde{a}_q - \Delta \leq \widetilde{a}_\rho < q-1 \leq \Delta \]. Hence the range of allowed values \( q \): \( Q \leq q \leq P \), and \( P := \min(n_{M-1} + 1, \rho) \). The algorithm for performing step 3 is shown in Figure 2.

![Fig.2. The algorithm for performing step 3.](image)

The algorithm shown in Figure 2 is implemented in the following steps:

1) Assigning \( M := 1 \). Calculating quantities \( A^M = a_1 - \widetilde{a}_\rho \)

2) In case of \( M := m \) go to step 7). At \( M < m \) assigning \( M := M + 1 \).

3) If \( M = m \) assigning \( \rho := n \). In case of \( M < m \) assigning \( \rho := M \).

4) Assigning \( P := \min(n_{M-1} + 1, \rho) \),

7) Assigning \( s_n := n \) and then perform a sequential computation of the numbers \( s_{m-1}, s_{m-2}, \ldots, s_1 \) by the recurrence formula \( s_{j-1} = q_{s_j} - 1 \). The resulting partition \((s_1, s_2, \ldots, s_{m-1})\) is optimal, that is, it has the minimum value of \( \Delta \) and the sum of the cluster sizes \( \sigma(\varepsilon) \).

V. RECOMMENDATIONS FOR THE CHOICE OF ALGORITHM PARAMETERS

Recommendations for choosing the value of \( \varepsilon \) for calculation. For each of the numbers \( \rho = 3, 4 \), we denote by \( Q_\rho(m) \) the maximum modulus of brightness increment of a large pixel in the transition from the second image to the \( \rho-th \).

We get

\[ Q_3(m) = \max_{1 \leq i \leq m} \max_{j:|j-i| \leq 1} |\rho_j - a_j| = 0.5 \Delta(\varepsilon) \quad \text{(due to the optimal partition \((s_1, s_2, \ldots, s_{m-1})\))} \]

\[ Q_4(m) = \max_{1 \leq i \leq m} \max_{j:|j-i| \leq 1} |\rho_j - a_j| \]

If it is necessary to define the third image with the value \( Q_3(m) \leq \delta \), it is enough to apply the algorithm for \( \varepsilon = 2\delta \). This will ensure a given degree of proximity of the second and third images. When constructing the fourth image, which corresponds to the condition \( Q_4(m) \leq \delta \) only the choice \( \varepsilon = \delta \) will guarantee its observance.

For the case of limiting the brightness of small pixels to 8 bits (range of values 0, 1, 2, ..., 255). The quantities \( \beta_j, \gamma_j \) will belong to the interval \([0, 255]\), however, they can take fractional values. In this case, they should be rounded according to the formula \( x := \lfloor x + 0.5 \rfloor \) \((\lfloor x \rfloor \) is the integer part of the number \( x \)). As a result, the shift of the brightnesses \( \beta_j, \gamma_j \) to the right or left along the numerical axis can reach 0.5, which requires decreasing \( \varepsilon = 2\delta \) by one, and \( \varepsilon = \delta \) by 0.5.

ACKNOWLEDGMENT

This work was supported by Russian Ministry of Education and Science in frame of the Federal Program "Research and development on priority directions of scientific-technological complex of Russia Federation in 2014-2020" (contract № 14.577.21.0212 (RFMEFI57716X0212)).


Nonlinearity of iris structure as way to improve recognition methods

1st Viktor Chigrinskiy  
Moscow Institute of Physics and Technology  
(State University)  
Moscow, Russia  
viktor.chigrinskiy@phystech.edu

2nd Ivan Matveev  
Federal Research Center  
"Computer Science and Control" of Russian Academy of Sciences  
Moscow, Russia  
matveev@ccas.ru

Abstract—Nonlinear movements of elements of human iris during pupil size variations is studied. Tracking of iris elements is done with the help of optical flow methods. The aim is to estimate a radially symmetric function which describes positions of iris structural elements with respect to pupil size. Knowing this function can help to reduce differences of iris patterns of the same eyes which leads to improving iris recognition methods. To estimate this function pairs

Index Terms—image processing, computer vision, biometrics, iris recognition, optical flow

I. INTRODUCTION

Iris recognition is a method of biometric human identification that uses mathematical pattern-recognition techniques on video images of one or both of the irises of an individual’s eyes, whose complex patterns are unique. But these patterns change with pupil size variation and this change is radial nonlinear. It means that different patterns will be obtained by conformal mapping the same eye’s iris structures with different pupil sizes into rectangles with the same shape. The problem is to reveal this nonlinearity assuming radial symmetry of the iris structure deformations and construct the function

\[ f : [\rho_{\text{pupil}}, \rho_{\text{iris}}] \rightarrow [\rho_{\text{pupil}}, \rho_{\text{iris}}], \]  

which maps iris structure of eye images with any pupil radius into the same eye with pupil radius equals half of iris radius.

To estimate this function optical flow on two eye images is implemented. Obtained optical flow is a velocity vector field which describes movement of any structural element of the iris. The desired function (1) is constructed based on radial components of the optical flow’s vectors.

II. IRIS RECOGNITION PROBLEM

Iris recognition includes three steps: iris segmentation, mapping a segmented area to the pattern with fixed size and comparing iris patterns. In this paper iris segmentation procedure is taken from [4] and pattern comparison is taken from [5]. Recognition quality was compared for this methods with standard (linear) iris pattern mapping and estimated with optical flow (nonlinear) one.

III. LINEAR IRIS AREA MAPPING

Iris area is cutted from the photo and mapped into a rectangle of a fixed size by the following rule:

\[ P(m, n) = I(\xi, \eta), \]

\[ \xi = (1 - \lambda)(\xi_{\text{pupil}} + \rho_{\text{pupil}} \cos \varphi) + \lambda(\xi_{\text{iris}} + \rho_{\text{iris}} \cos \varphi), \]

\[ \eta = (1 - \lambda)(\eta_{\text{pupil}} + \rho_{\text{pupil}} \sin \varphi) + \lambda(\eta_{\text{iris}} + \rho_{\text{iris}} \sin \varphi), \]

\[ \lambda = \frac{m}{M}, \varphi = 2\pi \frac{n}{N}, \]

where \( \xi_{\text{pupil}}, \eta_{\text{pupil}} \) — center of the pupil in the origin image \( I \); \( \rho_{\text{pupil}} \) — pupil radius; \( \xi_{\text{iris}}, \eta_{\text{iris}} \) — center of the iris in the origin image \( I \); \( \rho_{\text{iris}} \) — iris radius; \( M, N \) — size of the result image \( P \).

Demonstration of the polar transform is shown in the Figure 1.

Fig. 1. Polar transform
IV. OPTICAL FLOW ESTIMATION

There are a range of optical flow methods destined for the different problems. There are classic Lucas-Kanade [1] and Horn-Schunck [2] methods or novel TV-L1 [3] method. In this paper TV-L1 optical flow calculation is used because it assumes iris structure continuity and it is robust for noise.

Let’s take two images of the same eye with different pupil sizes and calculate optical flow between them. To visualize optical flow vector field colormap is used where hue means direction of displacement and intensity means absolute value. Colormap is shown in the Figure 2 and example of optical flow calculation is shown in the Figure 3.

The result lives up to expectations — after pupil constriction iris structure moved to the center. But according to the flow map only nearest to the pupil border part of iris was deformed while the periphery remained static. Let’s confirm this observation numerically. In the Figure 4 deformation model is shown. X axis corresponds to the normalized radius (where 0 is pupil border and 1 is iris border) and Y axis corresponds to the normalized (with the pupil radius change) displacement along the radius. Graph shows that the closer to the pupil border structure the greater it’s displacement.

Of course result on the just one example is not representative so 500 eye image pairs from CASIA-4 database were chosen and used for the same. Measured result is shown in Figure 5.

This observation is well described by the Erf function:

\[ f(x) = \text{Erf}(-2 \log(1 - x)) \]  (2)
V. NONLINEAR IRIS AREA MAPPING

According to the results standard deformation model III turns into the following:

\[ P(m, n) = I(\xi, \eta), \]

\[ \xi = (1 - \hat{\lambda})(\xi_{\text{pupil}} + \rho_{\text{pupil}} \cos \varphi) + \hat{\lambda}(\xi_{\text{iris}} + \rho_{\text{iris}} \cos \varphi), \]

\[ \eta = (1 - \hat{\lambda})(\eta_{\text{pupil}} + \rho_{\text{pupil}} \sin \varphi) + \hat{\lambda}(\eta_{\text{iris}} + \rho_{\text{iris}} \sin \varphi), \]

\[ \hat{\lambda} = \frac{(\rho_{\text{iris}}/2 - \rho_{\text{pupil}})(1 + f(\lambda)) + \lambda \rho_{\text{iris}}/2}{\rho_{\text{iris}} - \rho_{\text{pupil}}}, \]

\[ \lambda = \frac{m}{M}, \varphi = 2\pi \frac{n}{N}. \]

VI. RECOGNITION QUALITY ESTIMATION

The following iris databases were used for recognition quality estimation: AOptix, Bath, ES, UBI, CASIA, ICE.

From each database 1000 same eye pairs and 1000 different eye pairs were chosen and compared. Let error type I and error type II be as following:

\[ e_1(\theta) = \frac{FP(\theta)}{TP(\theta) + FP(\theta)}, \]

\[ e_2(\theta) = \frac{FN(\theta)}{TN(\theta) + FN(\theta)}, \]

where TP, FP, TN, TP — true positives, false positives, true negatives and true positives respectively and \( \theta \) is similarity threshold. Let equal error rate (EER) be \( e_1 \) or \( e_2 \) for the threshold value \( \hat{\theta} \) at which they are equal:

\[ \text{EER} = e_1(\hat{\theta}) = e_2(\hat{\theta}). \]

The DET-curves, which show a dependency of \(- \log(e_1)\) on \(- \log(e_2)\), are provided for the clarity.
VII. CONCLUSION

Though iris segmentation play the main role in the human recognition problem, the little details can slightly affect existing results. In this paper feature of nonlinear iris deformation is used to make recognition procedures more precise. TV-L1 optical flow method was used to estimate iris structure displacement between two eye images of the same eye. Obtained empirical function was approximated with analytical one. After that nonlinear iris deformation model was constructed. This model was compared with the standard one on the different iris databases and it turned out that using nonlinearity leads to improvement of recognition.

REFERENCES

Iris Segmentation in Challenging Conditions

1st Mikhail Korobkin  
National Research University of Electronic Technology (MIEE)  
Zelenograd, Russia  
mikhail.korobkin@hotmail.com

2nd Gleb Odinokikh  
Federal Research Center "Computer Science and Control"  
Russian Academy of Sciences  
Moscow, Russia  
g.odinokikh@gmail.com

3rd Yurii Efimov  
Moscow Institute of Physics and Technology (MIPT)  
Dolgoprudny, Russia  
yuri.efimov@phystech.edu

4th Ivan Solomatin  
Moscow Institute of Physics and Technology (MIPT)  
Dolgoprudny, Russia  
ivan.solomatin@phystech.edu

5th Ivan Matveev  
Federal Research Center "Computer Science and Control"  
Russian Academy of Sciences  
Moscow, Russia  
matveev@ccas.ru

Abstract—Iris segmentation is an irreplaceable stage of iris recognition pipeline. Its quality hugely affects overall accuracy. Wrong segmentation decreases recognition performance and makes the technology inconvenient for users. Previously when conditions were mild and controlled the task was solved by image processing techniques and rule based approaches. Nowadays widespread of biometric technologies has relaxed operation conditions for such systems demanding more flexible and robust solutions. 

Constantly increasing data and sensors availability created fertile field for growth of machine learning methods capable to cope with complex conditions. Deep learning and convolutional neural networks represent a class of machine learning methods that secured top results in many computer vision tasks. The latest contribution to iris segmentation was also made by neural networks. In spite of previously achieved great results this work addresses even more challenging conditions that allows iris recognition to be used in wide range of real life cases. Novel CNN architectures are proposed in this work. They were designed to combine the latest achievements in classification and semantic segmentation fields. FCN and SegNet architectures have been picked up as prototypes and were strengthened by residual blocks. This allowed to make lightweight networks that could be shipped on various embedded platforms to successfully operate under less controllable environmental conditions. The approach allowed to obtain 0.93 and 0.92 IoU on original and modified CASIA-Iris-Lamp datasets which is a significant improvement in comparison with the results achieved before.

Index Terms—Semantic segmentation, Deep Neural Networks, Convolutional Neural Networks, FCN, SegNet, Iris recognition

I. INTRODUCTION

An iris is a unique and reliable biometric trait of a person. A recognition system based on iris features has several advantages over the one built on other biometric traits (face, voice, etc.) that make it convenient for daily use and at the same time hard to be compromised [3]. These advantages made the technology to gain popularity and widespread use [4].

where an operator could facilitate this process and help the system to cope with its tasks. Nowadays iris recognition is driven by hardware development steps on another stair where it is demanded to run in various tough environmental conditions: different lighting, over-illumination, obstacles, high diversity, ability to operate in B2C products.

Challenging conditions affect not only characteristic of biometric information but also appearance of an eye in whole. Before the eye image is turned to a biometric template, it should pass several important stages of iris recognition pipeline. One of them is iris segmentation that allows to define eye structure, cope with internal eye variations (contraction and dilation of a pupil) and discern biometric features from obstacles (eyelids, eyelashes, etc).

Previously under controlled environment the techniques based on image processing and handcrafted rules dominated on this stage. Many of the proposed solutions included numerous preprocessing algorithms solely designed to deal with intra-class variations. But when object appearance changes drastically the rules start to contradict to each other. From some level of complexity that kind of systems become error-prone and take tremendous efforts for maintaining.

At the same time, machine learning area has significantly evolved due to increasing amount of various kind of data for training and hardware advancements. Deep learning (DL) is the one of those methods that is capable to efficiently utilize all available data and hardware resources to tackle the problem at hand. Starting from 2012 DL and convolutional neural networks (CNN) were tested in different computer vision tasks [6], [8]–[10]. The achieved results proved success of this approach and secured top places in many benchmarks. It took some time for DL to reach iris recognition field and be applied for the segmentation task [7]. Nowadays development of various applications, including mobile, requires iris recognition to be capable to operate under highly dynamic environment and demonstrate extreme robustness. To meet the constraints, new network architectures based on effective FCN [11] and SegNet [12] with redesigned internal blocks are proposed in this work. The solution based on the approach demonstrated better performance on CASIA-Iris-Lamp dataset [24] in com-
parison with the state-of-the-art works. In order to simulate real environment scenarios and test generalization capabilities of the developed networks, CASIA-Iris-Lamp dataset has been modified by adding more challenging contrast and brightness variations. Both the proposed architectures showed high accuracy results keeping reasonable memory consumption that allows to put them into use in a wide range of embedded solutions.

The paper is organized as follows: related work section describes classic and CNN based approaches have been developed so far; the general structure and implementation details of the network are described further; experimental results and conclusion are given in the end of the paper.

II. RELATED WORK

Traditionally, the majority of iris recognition solutions have been solving segmentation task were based on image processing algorithms and handcrafted rules [4], [13], [14]. The whole task was separated on several steps where each step was dedicated for segmentation of a separate feature: eye area, eyelids, eyelashes, iris and pupil etc. [1]–[3], [15].

There were other approaches tried to apply multilayer perceptron networks. But they suffered from the lack of computational resources, data and internal efficiency preventing their use in COTS systems [16].

The first application of convolutional neural networks and deep learning for iris segmentation was introduced in [7] where authors compared patch- and CNN-based approaches. It was also claimed that the use of CNN drastically improves segmentation accuracy especially for hard cases. Another example of the patch-based network was introduced in work [17].

Contrary to this, shortly before researchers in [11] have shown that such design and training method could seriously impact the final accuracy. Another paper demonstrates the use of standard SegNet architecture for segmentation task. Additionally, the network underwent dropout training that comes down to minimizing variational objective [18], [19]. This allowed authors to perform Monte Carlo (MC) dropout at inference time improving final accuracy. Such approach does not depend on architecture and could easily be incorporated to any CNN. The approach is considered as impractical for commercial applications since its inference time is claimed to be 58.7 seconds. Another CNN architecture proposed in [20] was designed without any pooling layers and showed its high efficiency. The architecture couldn’t learn enough deep features which are accountable for high-order logic because of weights layout. Also pooling layers in convolutional networks are responsible for spatial invariance. Lack of them makes the architecture performance extremely sensitive to changes in position [6]. The method proposed in this work is similar to [7] but used encoder-decoder segmentation architecture with redesigned internal blocks. It allows to incorporate the best performing classification network into structure of segmentation CNN. All the previously mentioned networks do not use any layer normalization but it is a vital element for seamless operability in challenging conditions as discussed in [21].

Advantages of the proposed design and normalization are shown in the results section.

III. NETWORK ARCHITECTURE

A design of the proposed network architecture was decomposed into following steps: selection of general segmentation structure and construction of its internal blocks.

A. Segmentation

Fully convolutional neural network (FCN) architecture, firstly proposed for semantic object segmentation task, had no fully connected layers at all [11]. This allowed to transform any network trained for classification into segmentation one without additional optimizations. It also allows to further improve performance and decrease stride size with unrolling of pooling layers having semantically labeled data. The architecture supports transitioning from patch-based classification, previously dominated in neural networks field, directly to pixel-wise segmentation while the new approach to learning increases training speed and as a consequence affordable dataset sizes. The network uses encoder-decoder architecture which is shown in Fig. 1. While going deeper from layer to layer the encoder reduces the spatial resolution and increases the number of channels encouraging construction of high-level representation of the input. The decoder performs the opposite task: it unwinds accumulated information and produces spatially distributed semantics. In FCNs it is performed using deconvolutional or transposed convolutional layers [22]. Each deconvolutional layer output is fused with feature maps from the correspondent decoder layer during the unrolling (Fig. 1) with so called skip-connections as a way to recover lost structural information.

Another encoder-decoder architecture for segmentation SegNet was proposed to even further reduce memory consumption within inference time [12]. The main contribution was to replace costly deconvolutions with unpooling layers. The aforementioned cost stems from necessity to store previous feature maps before deconvolution layer uses it. Thus, in an ordinary segmentation network a memory consumption pick fall on the central layers when all decoder feature maps are stored at the same time. The SegNet architecture allows to alleviate memory burdens by storing only pooled indices potentially decreasing footprint a hundredfold. Despite good motivation and apparent benefits it decreases network capacity and a lack of gradients for skip-connections makes the training harder. A difference in unrolling between two architectures is shown in Fig. 2. Authors of the SegNet proposed to replicate the same block as were used in encoder following the opinion

![Fig. 1. Encoder-Decoder architecture.](image-url)
that decoder task is as hard as the one solved in encoder. This imposes restrictions on the choice of internal blocks and discussed later in the design section.

B. Internal blocks

Originally, the idea of using residual connections in construction of building units was suggested in context of very deep neural networks [23]. Further, the residual units have established themselves as memory efficient and having high capacity type of the blocks. Their bypass connections support gradient flow and force the connected layer to learn an additive term for result function. An example of a general residual block is depicted in Fig. 3. Authors also proposed modifications of residual blocks that could differ in channel and depth (Fig. 4). All the blocks are merged with batch normalization layer [21] to speedup learning and boost generalization ability making network less sensitive to input variations. This feature also helps segmentation in hard conditions.

C. Design

Both of the aforementioned architectures (FCN, SegNet) were taken as a basement for implementation and modified. SegNet provides attractive footprint decrease but its manifestation strongly depends on hardware platform capabilities. On the other hand, FCN has better convergence prerequisites and straightforward structure. Keeping in mind practical applications, moderate block sizes were picked up for both architectures. The modified FCN proposed in this work is based on ResNet-26 encoder with bottleneck building blocks and thin decoder without any excessive layers except $1 \times 1$ convolutions for channel alignment. The proposed SegNet consists of encoder built from ResNet-18 simple units and the reflected decoder. Bottleneck blocks are not applicable for the SegNet architecture because a number of channels for the latest layer in block requires feature maps of prohibitively large size. For FCN architecture all max-pooling layers were removed and replaced with strided convolutions. It helped to improve accuracy a bit and made learning curves less juggled in the beginning of training. In case of SegNet max-pooling layers were shifted to the end of each encoder unit and convolution strides were set to ones.

IV. RESULTS

Another CNN based solution proposed in work [7] was chosen for comparison as having the most similar purposes and steady design. Intersection over Union (IoU) also known as the Jacard Index was chosen as a main metric for experiments evaluation [11]. It is defined as the size of the intersection divided by the size of the union of the sample sets where the
union is measured between prediction and ground truth pixels. A value of IoU was obtained for three network on CASIA-Iris-Lamp [24] and modified CASIA-Iris-Lamp datasets.

A. Dataset description

CASIA-Iris-Lamp was chosen as the main dataset for evaluation. Its peculiarity is that it introduces additional intra-class variations for image of the same iris which partially helps to simulate changes in environment. Among the variations are the following: different pupil size, gaze angle, brightness etc. All of them make segmentation challenging. The whole dataset contains 16212 iris images of 411 subjects. For segmentation 4865 images of 124 randomly selected subjects are used. Pixel-wise labeling was carried out manually by an expert with the following condition: all the pixels of iris area on the image except eyelashes that overlap iris are considered as belonging to iris (Fig. 5). The dataset has been preliminary separated on training, validation and test subsets with 3386, 478 and 1001 images respectively. Only the training subset is used for evaluation. Images from the validation set were utilized to pick the model showed the best accuracy whereas the final evaluation is performed on the test set.

B. Experiments

All the networks have been trained for 200 epochs from scratch with batch size equal to 8. Tensorflow library implementation of Adam optimization algorithm [25] is applied for training with initial learning rate equal to 1e−3. The same library is used for implementation of all the tested models including MFCN following the original paper [7].

All the networks were trained first on the dataset without any modifications. The results on IoU are represented in Table I. SegNet and FCN showed equally well results and slightly outperform MFCN.

An additional augmentation was applied to images in the dataset. It is made as the second experiment in order to simulate even more challenging environmental conditions. The following changes were applied consistently to every image in the set for training: (i) random contrast in the range [50%, 150%], (ii) random brightness in the range [−20%, 20%]. Final evaluation on the validation and test subsets is performed for original image to imitate uncommon conditions. As the result, both SegNet and FCN outperform MFCN by far (Table I). SegNet surpasses FCN by 0.3 IoU. SegNet demonstrates high readiness for changes in surrounding conditions and excellent generalization capabilities. Several examples of the segmentation result are depicted in Fig. 5.

V. Conclusion

Transformation of iris recognition from exclusive technology to a mass product for everyday use brought new challenges for biometric community. In this paper iris segmentation in tough conditions is considered. To cope with this task new network architectures showing better generalization in spite of drastic appearance changes are developed. Robustness and accuracy of the proposed approach are demonstrated experimentally on CASIA-Iris-Lamp dataset. Being accurate and lightweight proposed modified SegNet and FCN architectures are well-prepared to be shipped on many embedded platforms for commercial applications.

REFERENCES


Using optimal circular path method to match piecewise iris templates

1st Ivan Matveev
Department of Complex Systems
Dorodnicyn Computing Centre FRC CSC RAS
Moscow, Russia
matveev@ccas.ru

2nd Vladimir Novik
Moscow RnD Office
Iritech Inc.
Moscow, Russia
novikvp@mail.ru

Abstract—Majority of methods of iris recognition use templates containing localized texture features. Thus perfect alignment of templates plays crucial role in the work of recognition system. Known state-of-the-art alignment techniques are quite intricate. We present simpler and still efficient method. It is proposed to split the normalized iris image into several segments, calculate distances between possibly matching segments, represent these distances in form of a raster and search for the optimal path. The coordinates of this path’s elements give displacements of individual segments. The method was tested with public domain images of ICE2005 and CASIA databases. Its performance is comparable or superior to existing algorithms.

Index Terms—iris recognition, template matching, optimal path

I. INTRODUCTION

Iris recognition is one of the main methods of biometric identification currently used. The method involves selection of an annular region of eye image enclosed between the pupil and the sclera and comparison of two such regions for a pair of images. It is generally accepted to transform the iris annular region from the original image to a rectangle of specified size (normalization). Of all possible ways of normalization, the so-called “rubber-sheet” model [1] is the most popular and, apparently, close to optimal. Here we will be within this framework.

As a rule, the texture of the iris is uniform throughout its area, but it has a large number of small details. For this reason, the most successful matching methods are based on a comparison of the local texture characteristics, which are calculated by spatially and spectrally local transformations, such as Gabor or LoG wavelets. This approach was proposed already in the earliest papers [2], [3]. Many other types of features and methods of their matching, developed since then, are also essentially local, for example the key points of the iris [4], ordinal statistics [5], correlators [6]. The use of global descriptors such as PCA, LDA or ICA [7], [8], histograms [9], Fourier transforms [10], and a number of others [11], [12] did not justify itself. All these methods show the recognition error of \( EER > 1\% \), which is significantly worse than that of local methods, or their results are obtained on small and specific image bases.

The work is supported by the Russian Foundation of Basic Research, grant no. 16-07-01171.

The use of local characteristics for matching requires the alignment of the matched objects, preferably as accurate as possible. The inaccuracy in determining the iris region by the border detection algorithms is a big problem, since it generates nonlinear distortions.

Figure 1 schematically shows the results of the normalization of the initial image (a) with correctly calculated boundaries (b), with a relative error of 10\% of the radius of the pupil (c) and a relative error of 10\% in determining the radius of the iris (d). It is easy to see that the deformation has a regular character. However, an accurate analytical description is rather difficult, since it requires the account of at least four parameters, which can be determined imprecisely (two coordinates of the eye center and two radii of circles approximating the inner and outer boundaries of the iris). For more elaborate iris models, the number of parameters increases. For example, there are six parameters for the model of two nonconcentric circles. The problem of aligning of irises under the analytical model is poorly conditioned and its numerical solution is practically inapplicable due to instability. Therefore, heuristic ways of matching are investigated mainly.

In [13] it is proposed to locate inner and outer iris boundaries with active contours, and use this detailed description for normalization. The drawback here is also the instability of result especially in the case of partially occluded iris. Most of researchers propose to make imprecise normalization followed by aligning pieces of iris, supposing that such flawed normalization biases local textures weakly and the similarity of local areas is preserved. Considering this, templates should be matched through a set of their local areas, some area from the first template should find its vis-a-vis in the second template.
with some offset, and the offsets for different pairs of areas should not be necessarily equal. In this formulation, the main issue is what kind of offsets this can be and how they are related to each other for different areas. In [14] the case of unbound offsets is considered, which are then combined using the hidden Markov model. In [4] the calculation of nonlinear deformations of the iris is carried out by tracking the corresponding points using a special kind of correlator. In [15] global (to compensate for the angle of view) and local (to compensate for the inaccuracy in determining the pupil and iris) corrections are proposed. In [16]–[18] neighboring areas are aligned by elastic graphs. These methods are quite computationally complex. In this paper we propose a simpler approach based on the calculation of the optimal path.

The following section briefly describes the procedure for converting an eye image to a template. Then the application of optimal path to template matching is presented. The last section describes the statement of the computational experiment, its results and conclusions.

II. MAKING TEMPLATE OUT OF IMAGE

Consider a test database containing a set of eye images. Images are labelled with the persons’ unique identifiers in order to verify the correctness of identification. There is more than one person in the database and more than one image per person. The basic processing scheme is depicted in Figure 2. Here blocks represent the “state of the world” and are labelled with Latin chars as: source image (a), localization (b), normalized image and mask (c), extracted features forming the template (d), cross-matching decisions (e), and matching statistics and aggregated error values (f). Arrows represent the processes, which transform a state to the next one.

![Fig. 2. The processing scheme.](image)

Source image \( I(x, y) \) is processed by method [19], which outputs the iris localization as pupil contour, iris contour and occlusion mask. The pupil contour is the circle given by its center and radius \( (x_p, y_p, r_p) \), which is the best approximation of pupil-iris boundary. The iris contour is the circle \( (x_1, y_1, r_1) \), approximating iris-sclera boundary. The occlusion mask \( M(x, y) \) is an image of same size as source with zero pixels in place where iris is covered by eyelids, eyelashes, flashes etc.

Then the iris normalisation is performed. It is a mapping of a ring, enclosed between iris and pupil circles to a rectangular region. The normalized image coordinate system is rectilinear \( O\phi \rho \), where horizontal axis \( O\phi \) corresponds to angle measured along the pupil and iris circles in source image, and vertical axis \( O\rho \) corresponds to radial shift from pupil circle to iris circle. Both source image and mask are subjected to the transformation, which yields their normalized versions \( I(\phi, \rho) \) and \( M(\phi, \rho) \). Figure 2(c) depicts a sample of normalized image obtained from image in Figure 2(a) and occlusion mask thereof. There are several possible models of this transformation, here the so called “rubber-sheet model” [1] is used. The origin \( (x, y) \) for the point of normalised image \( (\phi, \rho) \) is expressed as:

\[
\begin{align*}
x(\phi, \rho) &= (1 - \rho)x_1(\phi) + \rho x_2(\phi), \\
x_1(\phi) &= x_p + r_p \cos(\phi), \\
x_2(\phi) &= x_1 + r_1 \cos(\phi), \\
y(\phi, \rho) &= (1 - \rho)y_1(\phi) + \rho y_2(\phi), \\
y_1(\phi) &= y_p + r_p \sin(\phi), \\
y_2(\phi) &= y_1 + r_1 \sin(\phi).
\end{align*}
\]

(1)

Coordinates \( y \) is computed accordingly. Dimensions of normalized image are set in ranges: \( \rho \in [0; 1] \), \( \phi \in [0; 2\pi] \). Brightness of the normalized image is obtained with the bilinear interpolation:

\[
N(\phi, \rho) = (1 - \{x\}) (1 - \{y\}) I([x], [y]) + \\
\{x\} (1 - \{y\}) I([x] + 1, [y]) + \\
(1 - \{x\}) \{y\} I([x], [y] + 1) + \\
\{x\} \{y\} I([x] + 1, [y] + 1),
\]

(2)

where \([a]\) and \(\{a\}\) are integer and fractional parts of \(a\) respectively.

Iris features \( V(\phi, \rho) \) are calculated as convolution of normalized image (2) with Gabor filter:

\[
V(\phi, \rho) = N(\phi, \rho) \ast g_{\sigma \lambda}(\phi, \rho),
\]

(3)

\[
g_{\sigma \lambda}(\phi, \rho) = \exp\left(-\frac{\phi^2}{2\sigma^2}\right) \exp\left(-i\frac{\phi}{\lambda}\right) \exp\left(-\frac{\rho^2}{s^2}\right),
\]

where \(\sigma\) and \(\lambda\) determine the spread of the wavelet in spatial domain and the wavelength of modulation, \(s\) is a degree of blurring of the filter along radial axis. Finally, the features used for matching are obtained as binarization of real and imaginary parts of array \( V(\phi, \rho) \):

\[
T_{Re}(\phi, \rho) = \begin{cases} 1, & \Re(V(\phi, \rho)) > 0, \\
0, & \text{otherwise}, \end{cases}
\]

(4)

\[
T_{Im}(\phi, \rho) = \begin{cases} 1, & \Im(V(\phi, \rho)) > 0, \\
0, & \text{otherwise}, \end{cases}
\]

Two components of (4) are joined together to form a template. In this work binary templates are used in experiments. But without loss of generality one can speak about any system of local features, which are calculated in a regular mesh. So, each eye image \( I(x, y) \) is converted to a template \( T(\phi, \rho) \) and accompanying mask \( M(\phi, \rho) \).
III. Template Matching

Any two binary templates can be matched with normalized Hamming distance:

\[ d_0(T_1, T_2) = \frac{1}{|\Omega|} \sum_{(\phi, \rho) \in \Omega} T_1(\phi, \rho) \oplus T_2(\phi, \rho) , \quad (5) \]

where \( \Omega = M_1 \cap M_2 \) is the intersection of non-occluded areas of two matching templates. In fact, more complex distance function should be used, which counts on possible uncertainty of iris angle due to image rotation. The rotation of source eye image turns to cyclic shift along \( \phi \) coordinate in normalized image. One of the templates (together with mask) is rotated and matched, minimum distance is found:

\[ d(T_1, T_2) = \min_{\psi \in [-S:S]} d(T_1, T_2, \psi) , \]

\[ d(T_1, T_2, \psi) = \frac{1}{\Omega(\psi)} \sum_{(\phi, \rho) \in \Omega} T_1(\phi + \psi, \rho) \oplus T_2(\phi, \rho) \quad (6) \]

\[ \Omega(\psi) = M_1(\phi + \psi) \cap M_2(\phi) , \]

where \( \psi \) is a possible rotation limited by maximum allowed rotation angle \( S \) of tested iris. The distance is normalized to the range \([0; 1] \).

A. Applying the optimal path method

Let’s split the template \( T_1 \) into \( N \) equally sized, non-intersecting, fully covering segments \( T_1^{(n)} \), \( n \in [1; N] \), which are located along angular axis as shown in Figure 3. Each such segment may be displaced by some angle \( \psi_n \in [-S; S] \) and matched against corresponding part of \( T_2 \) template according to (5). Partial distances obtained here can be organized as the matrix \( D = \{ d_{\psi,n} \} \) with size \((2S + 1) \times N\). Note that the computational complexity of obtaining this matrix does not exceed that of determining the distance (6). In these terms, the distance (6) is obtained as the minimum sum over the rows of the matrix \( D \):

\[ d(T_1, T_2) = \min_{\psi} \sum_{n} d_{\psi,n} . \quad (7) \]

That is, the angular displacements of all segments are the same, which can be called as a model of an undeformed “rigid body”. On the other hand, angular displacements can be made independent, minimizing each partial distance separately:

\[ d(T_1, T_2) = \sum_{n} \min_{\psi} d_{\psi,n} . \quad (8) \]

This corresponds to a model of a body, perfectly elastic with respect to rotations. A computational experiment with independent template displacements was carried out as described below. The model (8) gives a lesser identification error than (7) in conditions of inaccuracy of border detection when number of segments is small: \( 2 \leq N \leq 6 \). As the number of segments increases, the error increases due to locating false matches for small segments. The question arises whether it is possible to improve the alignment, if we introduce some restrictions on the angular range of segments. So, the nature of template should be something between completely restricted “rigid” body (7) and completely unconstrained (by angle) model (8). In this situation, in [14] it is proposed to make a relationship between displacements by introducing a hidden Markov chain. In this paper, we propose to use the optimal path method.

The smoothness of the normalization transformation means that the values \( \psi_n \) for neighboring indices are close. For distant indexes can vary greatly, but because of the cyclicity of the angle conversion, the values of \( \psi_1 \) and \( \psi_N \) should also be close. Thus, we get the task of selecting a sequence of elements of matrix \( \{ d_{\psi,n} \} \), with the following requirements: one and only one element in each column; row of selected elements changes no more than by small value between adjacent columns; sum of the selected elements is minimized. It is possible to present this problem as the definition of the minimum cost of a cyclic path in a matrix:

\[ d(T_1, T_2) = \min_{(\psi_1, ..., \psi_N)} \sum_{n=1}^{N} [d_{\psi,n} + C(\psi_n, \psi_{n+1})] , \]

\[ C(\psi_n, \psi_{n+1}) = \begin{cases} 0, & \text{if } ||\psi_n - \psi_{n+1}|| \leq 1, \ n \neq N , \\ 0, & \text{if } ||\psi_1 - \psi_N|| \leq 1, \ n = N , \\ \infty, & \text{otherwise} . \end{cases} \quad (9) \]

Figure 4 represents typical view of matrices \( D \) for three cases and the optimal path found. The size of the matrices is \( N = 15 \), \( S = 12 \). The darker is the cell the smaller partial distance is. The optimal path is outlined with white points. First matrix (a) is the case of matching two irises of one person (i.e. “genuine” match), when all border parameters are
detected correctly (or, maybe, have very similar errors). In
this case no rotation is necessary, optimal path is a straight
line and “rigid body” model would suffice as well. Matrix (b)
is the case of genuine matcher, but normalization is distorted
by border detection error. Here one can see that minima in
columns appear in some regular order, which forms a dark
“valley”, and the optimal path is a curved path through it.
Matrix (c) is a case of impostor match. Would it be correct
to check border detection or not, matrix of impostor match will have
chaotic location of minima in its columns, and the optimal path
will be forced through many large values of matrix, yielding
high total cost. The solution of (9) produces the distance and
offset to find their correspondents.

In all the calculations of this section an arbitrary parameter
$N$ is used, the number of segments to which the template
is divided. The selection of this parameter was carried out
experimentally, by calculating the classification error obtained
by using the distance with the given parameter $N$.

IV. EXPERIMENT SETUP AND RESULTS

By adding a threshold $\Theta \in (0; 1)$ to the distance $d(T_1, T_2)$ the classifier is obtained:

$$Class = \begin{cases} 
  \text{genuine, if } d(T_1, T_2) \leq \Theta , \\
  \text{impostor, if } d(T_1, T_2) > \Theta . 
\end{cases}$$

(10)

Since the persons’ unique identifiers are known for test
database, the decision of classifier can be matched against the
ground truth, and the quality of classifier can be evaluated
from the number of wrong classification events. The equal
error rate is defined by a trade-off between false match and
false non-match errors, which is governed by a classification
threshold $\Theta$:

$$EER = \frac{fn(\Theta)}{fn(\Theta) + tp(\Theta)} = \frac{fp(\Theta)}{fp(\Theta) + tn(\Theta)},$$

(11)

where $fn(\Theta)$, $fp(\Theta)$ are numbers of false non-match and
false match events (also referred to as false negative and false
positive), and $tn(\Theta)$, $tp(\Theta)$ are true non-match and true match
numbers respectively. As long as EER depends on number of
template segments, one can think of it as a function $EER(N)$
and search for the optimum:

$$EER^* = \min_N EER(N), \ N^* = \arg \min_N EER(N).$$

(12)

Two publicly available iris image databases were used:
ICE2005 subset of ND-Iris-0405 [20] and CASIA4-Lamp [21].
All images in the datasets are 480 rows and 640 columns.
ICE2005 subset contains 2593 images for 132 persons. Ma-
ajority of the subjects are Caucasian. All images were acquired
with LG 2200 iris biometrics system. The subset of all left
irises including 1527 images of 119 persons was used for
experiments. Number of images per iris is very uneven in this
database and ranges from single (33 persons have only one
image) to 31. Number of genuine matches is 15357, number
of impostor matches is 1149744. CASIA4-Lamp contains
images for over 800 irises, each iris is represented by 20 or
rarely a couple less images. Totally, the DB contains 16312
images All irises are Asian type. Images were collected in
near-infrared illumination using IKEMB-100 camera produced
by OKI-IrisPass (http://www.oki.com). This produces ap-
proximately $800 \times 20 \times 20/2 = 160k$ genuine matches and
$16312^2/2 - 160000 \approx 13M$ impostor matches.

A. Comparing two databases and two alignment methods

Figure 5 presents graphs of dependency of EER on the num-
er of segments $N$, $N \in [1; 30]$ for two involved databases.
Graphs, which are entitled “NoLink”, are obtained for model
of unrelated segments and distance (8), graphs “OptPath” are
derived for optimal path model with distance (9). Note that the
initial points of these graphs $N = 1$ coincide and correspond
to the “rigid body” model (7).

Based on the graphs, the following conclusions can be
drawn. The model of a “rigid body” almost always loses to
models with division into segments. Models with unrelated
segments for a small number of segments are better than
models of the optimal path, but with an increase in the
number of segments, they quickly saturate and further their
error increases. Models of the optimal path are saturated more
slowly, but they achieve substantially better results. Comparing
databases (ICE2005 against CASIA), one can see that at
$N = 1$ ICE2005 yields bigger EER value, but with growing $N$
it allows better classification for both matchers. The reason is
ICE2005 has relatively bigger share of images with low qual-
ity, which produce imprecise border detection and template
matching error with “rigid body” method. On the other hand,
ICE2005 has more diversified iris types and in general less
occlusions; both of this factors allow achieving better precision
if border detection error influence is compensated.

B. Tests with emulated localization error

We have modelled localization error and tested how tolerant
is the optimal path alignment method. Iris regions are localized
in image automatically by a set of methods described in
[19]. Iris region is described as a ring enclosed between two
(possibly non-concentric) circles, which approximate inner and
outer borders of the iris. The circles are defined by their centres and radii, \((x_P, y_P, r_P)\) for pupil and \((x_I, y_I, r_I)\) for iris. Also each image was processed by human expert who marked the same “ground-truth” circles, given by \((x_P^*, y_P^*, r_P^*)\) and \((x_I^*, y_I^*, r_I^*)\) respectively. For a set of \(N\) images of the database average absolute error of detection of pupil circle center \(\varepsilon_{PC}\) and pupil radius \(\varepsilon_{PR}\) are calculated as:

\[
\varepsilon_{PC} = \frac{1}{N} \sum_n \left((x_{P,n} - x_{P,n}^*)^2 + (y_{P,n} - y_{P,n}^*)^2\right)^{1/2},
\]

\[
\varepsilon_{PR} = \frac{1}{N} \sum_n \left| (r_{P,n} - r_{P,n}^*) \right|.
\] (13)

The errors for iris center: \(\varepsilon_{IC}\) and iris radius \(\varepsilon_{IR}\) are defined respectively. The table I gives the error of border detection method [19] for used databases.

<table>
<thead>
<tr>
<th>Database</th>
<th>(\varepsilon_{PC})</th>
<th>(\varepsilon_{PR})</th>
<th>(\varepsilon_{IC})</th>
<th>(\varepsilon_{IR})</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASIA</td>
<td>1.05</td>
<td>1.13</td>
<td>2.44</td>
<td>1.62</td>
</tr>
<tr>
<td>NDIRIS</td>
<td>0.84</td>
<td>1.14</td>
<td>1.97</td>
<td>2.26</td>
</tr>
</tbody>
</table>

Border detection error is modelled by adding random values uniformly distributed in range \([-K; K]\), \(K \in [0; 2]\) to each of parameter of pupil and iris circle. Figure 6 shows dependency of EER value on the localization distortion strength. The database used for calculations for this figure is a mixture of CASIA and NDIRIS. The increase of \(K\) leads to growth of EER. The results for \(K = 0\) (no error is introduced) and \(K = 1\) are close, because original localization data already has errors, as described in Table I, and adding small distortion does not change things much. The optimal number of segments remains near 16.

The Table II gives the EER (11) in percent for proposed approach and some state of the art methods for two databases involved.

```
<table>
<thead>
<tr>
<th>Method</th>
<th>ICE2005</th>
<th>CASIA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhang et al. [4]</td>
<td>-</td>
<td>0.59</td>
</tr>
<tr>
<td>Liu et al. [22]</td>
<td>0.63</td>
<td>-</td>
</tr>
<tr>
<td>He et al. [23]</td>
<td>0.53</td>
<td>-</td>
</tr>
<tr>
<td>Rigid body, (N = 1)</td>
<td>2.72</td>
<td>1.76</td>
</tr>
<tr>
<td>Nolink, optimal (N = 5)</td>
<td>0.64</td>
<td>0.79</td>
</tr>
<tr>
<td>Optimal path, optimal (N = 16)</td>
<td>0.52</td>
<td>0.54</td>
</tr>
</tbody>
</table>
```

“Rigid body”, which is a straightforward implementation of Daugman’s scheme [2] has poor performance and loses to any of state of the art approaches. Splitting into segments and unconstrained matching (8) enhances the situation substantially, but still is inferior to other methods. The optimal path model covers the gap, and the proposed method demonstrates superior recognition quality to known solutions, according to EER. At that optimal path method is quite simple both algorithmically and computationally.

There is a room to further develop the optimal path method. So far the segments were shifted only along angular coordinate that corresponds to their rotation around the center of the eye. Radial movement of the segments (to and from the center) was not involved. However it is easy to see that inaccurate border detection provides radial distortions as well as angular ones, and consequently radial distortions might be treated and compensated. Some sort of compensation is embedded into the Gabor wavelet function (3). The blurring along radial axis (last multiplier in (3)) reduces the influence of radial distortion, but to the same extent it suppresses the information about radial structures of the iris. Such approach was grounded by the assertion that iris mainly exhibits angular structures rather than radial [2]. While looking at a typical iris this assertion seems reasonable, however, to the best of authors’ knowledge, it was not proved by numerical experiments with iris images yet.

Considering the above, the piecewise template matching with optimal path might be modified to incorporate radial movements. This will also require to reconsider the values of \(\lambda, \sigma, \) and \(s\) in (3), since the ability to shift template segments in both directions will yield other different optimal shape of feature filter. Introducing one more dimension in the template matching will require more calculations, but it may be paid off by decreased classification error.

V. CONCLUSIONS

The influence of splitting the iris into the segments upon classification error was studied. The models of “rigid body”, unconstrained segment offset and alignment by optimal path were investigated. Numerical experiments have shown that this algorithm can significantly improve the accuracy of recognition and achieve the performance of complex state of the art approaches. The highest accuracy is achieved with 5 segments for unconstrained offset model and near 16 for optimal path model. The running time of the algorithm with \(N = 16\) is 150 microseconds per one comparison with Intel Core i7-
3770 CPU and allows for multiple acceleration when using multiprocessor systems.

REFERENCES


Iris Anti-spoofing Solution for Mobile Biometric Applications

1st Gleb Odinokikh
Federal Research Center
"Computer Science and Control"
Russian Academy of Sciences
Moscow, Russia
g.odinokikh@gmail.com

2nd Yuri Efimov
Moscow Institute of Physics and Technology (MIPT)
Dolgoprudny, Russia
yuri.efimov@phystech.edu

3rd Ivan Solomatin
Moscow Institute of Physics and Technology (MIPT)
Dolgoprudny, Russia
ivan.solomatin@phystech.edu

4th Mikhail Korobkin
National Research University of Electronic Technology (MIEE)
Zelenograd, Russia
mikhail.korobkin@hotmail.com

5th Ivan Matveev
Federal Research Center
"Computer Science and Control"
Russian Academy of Sciences
Moscow, Russia
matveev@ccas.ru

Abstract—The ability to provide reliable protection against counterfeiting is one of the key requirements for a biometric security system. Iris recognition as the technology emerging on mobile market is assumed to handle various types of spoof attacks to prevent compromise of the user’s personal data. A method of iris anti-spoofing is proposed in this work. It is based on applying of convolutional neural network and capable to work in real-time on the mobile device with highly limited computational resources. Classification of iris sample for spoof and live is made by a single frame using a pair of images: eye region and normalized iris. The following types of iris spoof samples are considered in this particular work: printed on paper, printed on paper with application of transparent glue. Testing of the method is performed on the dataset manually collected and containing all the mentioned spoof sample types. The method revealed its high performance in both classification accuracy and processing speed as well as robustness under uncontrollably changing environmental conditions, which are specific and significant when interacting with the mobile device.

Index Terms—mobile biometrics, iris recognition, anti-spoofing, machine learning, neural networks

I. INTRODUCTION

Iris recognition has some advantages over other biometric technologies [1], [3], [4], that make it one of the most preferable for mobile biometric applications. In recent years, several companies introduced iris based authentication built into their smartphones, the most noted among them are [5]–[7]. Biometric authentication is assumed to become a replacement for conventional schemes with all kind of passwords (account information, personal data storage etc.) in general. It is intended to make interaction with the device more convenient and, at the same time, to increase security level of user’s personal information.

After the release of devices equipped with iris scanner on the market, several attempts to break the biometric sensor have been taken. Some of the spoof attacks were successful as reported in articles [8] and [9]. It is confirmed by our experiments that ideas of both mentioned spoofing methods are workable, except several important conditions that must be met: iris image should be captured with high resolution infrared camera in the way that iris diameter must have size of at least 250-300 dots on the paper printed with resolution of at least 600 dpi, which means the image should be captured either from a very short distance or using high-resolution telephoto lens; eyes should be opened enough with the sight directed to the camera; iris image should not be blurred and under- or over-illuminated; Thus, it might be concluded that iris anti-spoofing problem remains actual nowadays, especially for mobile applications.

The method proposed in this work is based on applying of convolutional neural network (CNN) and capable to operate in real-time on a smartphone. Classification of the iris sample for spoof and live is made by a single frame using a pair of images: eye region and normalized iris. Since anti-spoofing is a problem for which it is difficult to foresee all possible types of input data (attack scenarios), this work is concentrated on the following iris spoof types: printed on paper, printed on paper with application of transparent contact lens, printed on paper with application of transparent glue. Testing of the method is performed on the dataset manually collected using Raspberry Pi 3 single-board computer with compatible compact camera module and active infrared illumination. Variations in environmental condition, playing the key role when interacting with mobile device, have been taken into account during the dataset collection. Experimental results showed high efficiency of the proposed method on both the classification accuracy and processing speed.

The paper is organized as follows: state-of-the-art methods are discussed in Section 2; sections 3 and 4 describe the proposed method and experimental results respectively; conclusion is given in Section 5.
II. RELATED WORK

All of the existing iris anti-spoofing techniques described in the literature could be divided into two classes depending on the part of a biometric system where they are integrated: (i) techniques that require special hardware or user interaction (sensor-level approaches) and (ii) algorithms designed to work on either single or sequence of images (feature-level approaches). Basically, the first class is considered to a lesser degree when it comes to their application to mobile device mainly because it could significantly increase the cost and, at the same time, decrease its usability. On the other hand, the feature-level approaches are reliable and cost-effective, which makes them attractive for researchers and, by this reason, they are only considered in this work.

One of the pioneer works in this field was carried out by J. Daugman [2] where he noticed that printing process can leave detectable traces on counterfeit samples and propose to detect them by simple 2D Fourier analysis of the acquired iris image. The approach showed its efficiency against spoof attacks with iris images printed on paper but can simply be bypassed by another types of attack described further. Several methods to analyze features inherent to artificial iris samples in frequency domain were proposed by Czajka [10] and He et al. [11]. In work [12] a method of iris image decomposition into Laplacian pyramids of various scales is proposed. It allows to analyze the frequency responses for different iris orientations and detect spoofing artifacts using the sequence of iris images. Methods based on local descriptors are also used for analysis and representation of iris texture with the aim of spoofing attacks detection. For instance, in works [13], [14] efficiency of using different configurations of Local Binary Patterns (LBPs) against a number of known attacks (e.g. contact lenses, photo attacks, artificial irises, etc.) is demonstrated. Binary Statistical Image Features (BSIF) have also been studied in iris liveness detection context and tested on different databases, including the one comprising five types of spoof attacks [15]. A complex anti-spoofing solution based on a combination of multiple local descriptors (LBP, BSIF and Local Phase Quantization (LPQ)) for texture representation is presented in the comprehensive study [16] and tested on iris, face and fingerprint spoof databases.

It has also been shown that the image quality assessment metrics can be used for spoofing attack detection. The idea comes from the assumption that captured fake image is expected to have different quality than a real sample acquired in the normal operation scenario for which the sensor was designed. Several iris-specific quality measures [18] have been studied to distinguish between real iris images and captured iris printouts.

One of the most promising approaches for image classification nowadays is the use of deep neural networks. They have been applied to the problem of iris, face and fingerprint liveness detection as well, reporting promising results against iris print attacks in [17].

A good comparative study of several iris anti-spoofing methods, including above mentioned, can be found in articles reporting the results of recent Liveness Detection-Iris competitions (LivDet-Iris) held in 2013 [19], 2015 [20] and 2017 [21].

The aforementioned liveness detection techniques are also studied in the context of mobile iris applications, where feature-level approaches meet memory and processing time limitations. Furthermore, there are anti-spoofing solutions based on image quality assessment [22], [23] and multi-scale BSIF iris texture analysis [15], that were initially stated as being capable to work on mobile device.

III. PROPOSED METHOD

A method of iris anti-spoofing is proposed in this paper. The input data of the algorithm is a pair of images: eye region image spatially centered to the iris center $I_{ER}$ and the normalized iris image $I_{NI}$ (Fig. 1). The method relies on information about pupil and iris where both, in the simplest case, can be set parametrically as circles.

Basic structure of the algorithm is depicted on block-diagram Fig. 1. It is applied after the iris normalization stage, which is common for iris recognition pipeline, and consists of two steps: scaling of both images and passing them through the neural network.

Eye region image $I_{ER}(M_{ER}, N_{ER})$ is cropped first with relation to iris radius $R_i$ as $M_{ER} = N_{ER} = 3 R_i$, where the image and iris centers correspond to each other.

![Fig. 1. Iris anti-spoofing algorithm structure.](image-url)
Further both $I_{ER}$ and $I_{NI}$ images are scaled to a constant size as depicted on Fig. 1. The target resolution of both images is chosen as optimal considering the expected accuracy and execution time as well as the network architectural features.

A. Network architecture

The proposed CNN architecture is inspired by the idea of Google’s MobileNets [25], showed their efficiency and suitability for running on mobile platforms. It has two inputs for eye region and normalized iris images correspondingly. The images are obtained from the same image containing iris and passed to corresponding blocks of the network (Fig. 1). Both eye region and normalized iris blocks of the network are described in Table I. They have similar structure and built from smaller depthwise separable convolution blocks, proposed in [25] and represented in Tab. II.

Feature maps of eye region and normalized iris images passed through the corresponding convolution blocks are concatenated after global average pooling layer (Fig. 1). The last fully connected layer has two outputs corresponding to spoof and live classes (0 and 1 respectively). Softmax is used next as classifier and determines probabilities ($Prob_{spoof}$ and $Prob_{live}$) of belonging of the sample to each class.

In addition, except the proposed network architecture has fewer count of parameters in comparison with the original one [25] it also uses ‘valid’ paddings only. In conjunction with an optimally selected resolution of the input image this approach allows to avoid the loss of information on the image borders when performing a convolution operation as well as to speed up forward passing of the image through the network.

![Figure 2](image_url)  
Fig. 2. Examples of iris images of different classes of the same eye from captured dataset: (a) - captured under normal illumination condition (IN), (b) - captured outdoors (OUT), (c) - printed on paper (PR), (d)-(e) - printed on paper with transparent lens (PWL), (f) - printed on paper with transparent glue (PWG).

### TABLE I

<table>
<thead>
<tr>
<th>Layer</th>
<th>Eye Region Conv. Block</th>
<th>Norm. Iris Conv. Block</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv. 3x3 ($s' = 2$)</td>
<td>1 x 91 x 91</td>
<td>1 x 59 x 123</td>
</tr>
<tr>
<td>Conv. Block ($s' = 1$)</td>
<td>8 x 45 x 45</td>
<td>8 x 29 x 61</td>
</tr>
<tr>
<td>Conv. Block ($s' = 2$)</td>
<td>16 x 43 x 43</td>
<td>16 x 27 x 59</td>
</tr>
<tr>
<td>Conv. Block ($s' = 1$)</td>
<td>32 x 21 x 21</td>
<td>32 x 13 x 29</td>
</tr>
<tr>
<td>Conv. Block ($s' = 2$)</td>
<td>64 x 19 x 19</td>
<td>64 x 11 x 29</td>
</tr>
<tr>
<td>Conv. Block ($s' = 1$)</td>
<td>64 x 9 x 9</td>
<td>64 x 5 x 13</td>
</tr>
<tr>
<td>Global Avg. Pooling</td>
<td>64 x 7 x 7</td>
<td>64 x 3 x 11</td>
</tr>
</tbody>
</table>

### TABLE II

<table>
<thead>
<tr>
<th>Layer</th>
<th>Stride</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depthwise Conv. 3 x 3</td>
<td>s</td>
</tr>
<tr>
<td>BatchNorm</td>
<td>-</td>
</tr>
<tr>
<td>ReLU</td>
<td>-</td>
</tr>
<tr>
<td>Conv. 1 x 1</td>
<td>1</td>
</tr>
<tr>
<td>BatchNorm</td>
<td>-</td>
</tr>
<tr>
<td>ReLU</td>
<td>-</td>
</tr>
</tbody>
</table>

B. Dataset description

There are several datasets containing images of live and spoof iris available. By analogy with the datasets collected for evaluation of iris recognition performance, they could be divided into two groups: captured in visible (VIS) and near-infrared (NIR) light spectrum. Since NIR systems are more common due to several advantages they have, iris spoofs captured in NIR are only considered in this work. There are also several common types of iris spoofs have been considered in the related works, among them the following: iris printed on paper; live iris covered with a textured (patterned) contact lens; live iris covered with a transparent contact lens. The live eye plus contact lens type of the spoof attack is not essential in case of mobile device because the device is personal and the main purpose of biometric system built into it is to provide secure access to the user’s personal data. One of the possible scenarios in this case is the attacker has managed to reproduce the structure of user’s iris, but it seems very difficult. The scenario of attack with iris printed on paper is more simple and intuitive. Some of state-of-the-art works have succeeded in solving this problem but none of them consider the mobile device case. There are no available spoof/live iris datasets captured using mobile device yet.

For this reason, a special dataset was collected. The following types of spoof samples were included: (i) iris image printed on paper (PR); (ii) iris image printed on paper with imposition of transparent contact lens (PWL); (iii) iris image printed on paper with application of transparent glue (PWG). Such types of spoofing samples were chosen as they had been successfully used for bypassing a mobile biometric system [8], [9]. The iris images have been captured using high resolution NIR camera in a distance range from 20 to 40 (cm) and printed further on white paper with resolution 600/1200 (dpi) approximately half-and-half. The same printed papers were used for capturing of images of all the three spoof classes. Two categories were chosen to represent live iris samples: (i) iris image captured...
under normal illumination condition inside a well-lit room (IN); (ii) iris image captured outdoors (OUT) in order to consider changing in environmental conditions inherent to a mobile device.

As the mobile device for capturing Raspberry Pi 3 single-board computer with compatible compact camera module (PiCamera v2.1) covered with IR filter was chosen. An additional active NIR unit (LED) was also used for illumination. More detailed information about the dataset is represented in Table III. A split of the data for train/test sets is done in such way that the subsets contain different eyes. Several examples of eye region images from the dataset, cropped and centered as described on the diagram Fig. 1 are represented in Fig. 2.

### IV. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed solution several anti-spoofing methods based on different feature-level approaches were implemented and compared with the proposed one on the collected dataset. Among them frequency-based methods proposed by Czajka [10] and He et al. [11], as well as solutions using LBP by Gupta et al. [14] and multi-scale BSIF by Raghavendra and Busch [15] and method proposed by Sequeira [22] utilizing image quality features. The aforementioned methods were chosen as showing the best performance on the datasets captured using NIR iris sensor [24]. For the reason of high computational complexity the method based on applying of a pair of CNNs (trained separately for the printed and contact lens spoofing attack cases) in conjunction with a set of decision rules, proposed by researchers from CASIA in work [21], was excluded from consideration as unsuitable for mobile applications working in real-time. Its total execution time was approximately 400 times higher then for the method proposed in this work.

The following metrics are used for the accuracy evaluation: **FerrLive** corresponding to the rate of misclassified fake iris images, **FerrFake** corresponding to the rate of misclassified live iris images and **Correct Classification Rate (CCR)** for the whole dataset. Table IV contains accuracy values for the implemented state-of-the-art solutions and proposed method. It is important to note, that only two of the mentioned solutions [15], [22] were designed especially to use in mobile iris recognition system. While image quality metrics in [22] are based on lightweight algorithms, such as FFT, the use of multi-scale BSIFs in [15] involves computationally expensive non-separable image convolutions with pre-learnt filters of various sizes from $7 \times 7$ to $17 \times 17$ that makes this method less preferable for deployment on mobile device.

Testing of the proposed method for speed performance is performed on a mobile device with Android platform. The total execution time is measured on Qualcomm Snapdragon 835 CPU (2.45 GHz) and was 4-6 milliseconds. The measurements are carried out on the single core.

### V. CONCLUSION

An iris anti-spoofing solution is proposed in this work. It is based on applying of convolutional neural network and performed right after iris segmentation and normalization stages of common iris recognition pipeline. The trained model takes eye region and normalized iris images as the input and makes the decision about possible spoof attack by a single frame. The method showed high efficiency in detection of different kinds of iris spoof attacks, some of which have not been considered in the literature before. A special dataset of live and fake iris image samples was collected using a mobile device. It is demonstrated experimentally that the method outperforms existing state-of-the-art approaches and capable to operate in real-time on mobile device.

### REFERENCES


### TABLE III

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>People/Eyes</td>
<td>23/46</td>
</tr>
<tr>
<td>Spoof/Live</td>
<td>18548/18031</td>
</tr>
<tr>
<td>IN/OUT/PR/PWL/PWG (full set)</td>
<td>10679/7869/6233/5907/5891</td>
</tr>
<tr>
<td>IN/OUT/PR/PWL/PWG (test set)</td>
<td>2534/2006/1436/1452/1568</td>
</tr>
</tbody>
</table>


Underwater Image Enhancement Algorithm Based on Logarithmic Transform Histogram Matching With Spatial Equalization

Viacheslav Voronin  
Lab. "Mathematical methods of image processing and intelligent computer vision systems"  
Don State Technical University  
Rostov-on-Don, Russia  
voroninslava@gmail.com

Evgenii Semenishchev  
Lab. "Mathematical methods of image processing and intelligent computer vision systems"  
Don State Technical University  
Rostov-on-Don, Russia  
sea.sea@mail.ru

Sos Agaian  
Dept. of Computer Science  
CUNY/The College of Staten Island  
Staten Island, New York, United States  
sos.agaian@csi.cuny.edu

Abstract— In this paper a novel algorithm is proposed for underwater image enhancement. The proposed method based on combined local and global image processing in the frequency domain. The basic idea is to apply logarithmic transform histogram matching with spatial equalization approach on different image blocks. The resulting image is a weighted mean of all processing blocks driven through optimization of measure of enhancement (EME). Some presented experimental results illustrate the performance of the proposed algorithm on real underwater images in comparison with the classical contrast enhancement techniques and adaptive histogram equalization techniques.

Keywords— underwater image; enhancement; equalization; logarithmic transform; the measure of enhancement (EME).

I. INTRODUCTION

Image enhancement is the important tool in the image processing with a purpose to improve the visual appearance of the image or to provide “better” transform for future analysis (segmentation, contour detection and recognition) [1, 2]. In many cases, enhancement procedures are used as a preprocessing step for image visualisation.

Underwater cameras are widely used to observe the sea floor. They are usually included in autonomous underwater vehicles (AUVs) [3].

Underwater image processing is challenging problem and its importance for the environment [4]. The quality of underwater images is still much worse than that of air images because of the limitations imposed by the physical properties of the water medium [4]. Underwater images usually appear foggy and hazy. Such images are characterized by their poor visibility because light exponentially attenuated underwater. This leads to image has low contrast. Therefore, underwater images can hardly be used for understanding without image enhancement processing.

Enhancement approach can be classified into two categories: spatial and frequency domain methods [5].

The first group use spatial domain image processing which directly manipulates the pixels. Many spatial image enhancement methods are based on histogram analysis and modification; other methods are based on the local contrast transformation [1].

The most popular image enhancement method is histogram equalization. It is a global processing approach, so the entire tone of the image has been changed like more bright or dark image. In many cases, these methods extend the dynamic range of an image in local regions, leading to artefacts and overall tonal change of the image.

The second group use transformation in the frequency domain through modification magnitudes and altering the frequency content of the image. These enhancement techniques use frequency transform such as DCT, Fourier, etc. Sometimes the image properties such as low and high-frequency coefficient’s histograms may be so tightly packed that distinguishing them from one another may be impossible [1, 6]. Logarithmic transform allows improving the difference between levels of images [1].

Adaptive histogram equalization (AHE) is an image processing technique used to improve contrast in images [7, 8]. An adaptive version of this algorithm called contrast limited adaptive histogram equalization (CLAHE) [8,9].
II. PROPOSED ALGORITHM

The block diagram of the proposed enhancement algorithm for the underwater image is depicted in Fig. 1:

The procedure for the proposed algorithm is expressed as following steps:

*Input - Original Image.*

*Step 1 – Image splitting.*

We split image in moving windows on disjoint blocks with different sizes (8 by 8, 16 by 16, 32 by 32 and, i.e.) (Fig. 2).

*Step 2 – Enhancement Processing.*

For every sub-image, we use the frequency domain enhancement method based on the logarithmic transform histogram matching with spatial equalization. The block diagram of the enhancement processing shows in Figure 3.

We perform image transformation, which needs to be enhanced, then manipulated the transform coefficient, and then perform the inverse orthogonal transform.

In proposed approach is used Discrete Fourier Transform as orthogonal transform functions. The histogram of this data is usually compact and uninformative. Logarithmic transform in our scheme helps to maps a narrow range of low grey level values into a wider range of the output level. This type of transformation is used to expand the values of dark pixels in an image while compressing the higher-level values [11].

The Log transformation includes two steps:

- the creation of a matrix to preserve the phase of the transformed image, which is given by the equation:

\[ \Theta(i, j) = \text{angle}(\hat{X}(i, j)). \]

The angle function returns the angle of the coefficient. And will be used to restore the phase of the transform coefficients.

- the transformation of the coefficients according to the following equation:

\[ \hat{X}(i, j) = \gamma \ln(\eta |\hat{X}(i, j)| + \lambda). \]

where \( \eta, \gamma, \) and \( \lambda \) are enhancement parameters, usually set to 1.

To return the coefficients to the standard transform domain the signal is exponentiated, and the phase is restored as shown by the following equation:

\[ \tilde{X}(i, j) = e^{\hat{X}(i, j)} \, e^{j\Theta(i, j)}. \]

This allows to keep the overall image characteristics and return them to the original image.

It is important to keep the phase information unchanged because the angle contains most of the images underlying characteristic information [12].

Histogram equalization maps the input image’s intensity values so that the histogram of the resulting image will have an approximately uniform distribution as follow [12]:

\[ s = T(r) = \int_{0}^{r} p_s(w)dw, \]

where \( r \) is the grey level of an input image, \( T \) is the transformation function, \( s \) is the transformed value, \( p_r \) is the probability density function of the given image and \( p_s \) is the specified probability density function.
The following equation can obtain the probability density function of \( r \) and respectively:

\[
p_s(s) = \frac{p_s(r)}{ds}.
\]

The adaptive histogram equalization is operating on small local regions, rather than the global image. The contrast transform function is calculated for each of these regions individually.

The drawback of the original adaptive histogram equalization method is choosing the optimal size of regions. In the most cases, it depends on the type of the input image. Therefore, we propose to use CLANE on every disjoint block with different sizes (8 by 8, 16 by 16, 32 by 32 and, i.e.) in moving windows.

Histogram mapping is a more generalised version of histogram equalization which allows us to specify the shape of the histogram that we wish the processed image to have [12]. The method used to generate a processed image that has a specified histogram is called histogram matching or histogram specification.

Fig. 3. Block diagram of the enhancement processing.

This method includes three steps:

1) Histogram equalize the original image.
2) Histogram equalize the output image.

Histogram equalization \( G(z) \) calculates by the following equation

\[
G(z) = \int_0^z p_z(t)dt.
\]

3) The inverse of the second transform to the original equalized image.

For histogram equalization \( G(z) \) should be equal to \( T(r) \), i.e. \( z = G^{-1}[T(r)] \).

Step 3 – EME calculation.

There are several methods introduced as a measure of image enhancement [13]. To measure the quality (or contrast) of images and select the optimal processing parameters, we use the following quantitative measure of image enhancement proposed in [12]. The EME is image enhancement measure introduced by Agaian:

\[
EME_{k_i} = \max(1 \frac{1}{k_1 \cdot k_2} \sum_i \sum_j 20 \cdot \log \frac{X_{0}^{\text{max}(i,j)}}{X_{0}^{\text{min}(i,j)}})
\]

where \( X_{0}^{\text{max}(i,j)} \) and \( X_{0}^{\text{min}(i,j)} \) respectively are the minimum and maximum of the image \( x(n,m) \) inside the block \( \omega_{k_i} \).

We calculate EME for every enhanced image:
- for enhanced image 1 (all image) is \( \overline{EME_{\tilde{X}1}} \);
- for enhanced image 2 (blocks 8 by 8) is \( \overline{EME_{\tilde{X}2}} \);
- for enhanced image 3 (blocks 16 by 16) is \( \overline{EME_{\tilde{X}3}} \);
- for enhanced image 4 (blocks 32 by 32) is \( \overline{EME_{\tilde{X}4}} \).

These values allow to calculate weights, as follow:

\[
W_{\tilde{X}_1} = \frac{\overline{EME_{\tilde{X}1}}}{\overline{EME_{\tilde{X}1}} + \overline{EME_{\tilde{X}2}} + \overline{EME_{\tilde{X}3}} + \overline{EME_{\tilde{X}4}}};
\]

\[
W_{\tilde{X}_2} = \frac{\overline{EME_{\tilde{X}2}}}{\overline{EME_{\tilde{X}1}} + \overline{EME_{\tilde{X}2}} + \overline{EME_{\tilde{X}3}} + \overline{EME_{\tilde{X}4}}};
\]

\[
W_{\tilde{X}_3} = \frac{\overline{EME_{\tilde{X}3}}}{\overline{EME_{\tilde{X}1}} + \overline{EME_{\tilde{X}2}} + \overline{EME_{\tilde{X}3}} + \overline{EME_{\tilde{X}4}}};
\]

\[
W_{\tilde{X}_4} = \frac{\overline{EME_{\tilde{X}4}}}{\overline{EME_{\tilde{X}1}} + \overline{EME_{\tilde{X}2}} + \overline{EME_{\tilde{X}3}} + \overline{EME_{\tilde{X}4}}};
\]

Step 4 – Weighted Average.

The resulted enhanced images define as:

\[
\tilde{X} = \tilde{X}_1 \cdot W_{\tilde{X}_1} + \tilde{X}_2 \cdot W_{\tilde{X}_2} + \tilde{X}_3 \cdot W_{\tilde{X}_3} + \tilde{X}_4 \cdot W_{\tilde{X}_4}.
\]

Output – Enhanced Image.

III. EXPERIMENT RESULTS

In this section, we compare our results with well-known algorithms histogram equalization and CLAHE. The dataset contains 100 underwater sequences.

Figures 4-7 demonstrate the underwater image enhancement results obtained by various algorithms respectively (a - original image; b - the enhanced image by the histogram equalization; c - the enhanced image by the CLAHE; d - the enhanced image by the proposed method). The results achieved by current proposed scheme have visually more contrast.
Proposed approach shows more details in the obtained enhanced underwater images. The analysis shows, what the proposed method gives better visual quality than histogram equalization technique and CLANE.

The experimental results (Table 1) show that the original images have the lowest EME. After applying proposed local and global processing, the EME has risen. It is noticeable that the quality of the obtained results by the proposed algorithm has several times better in regard to the EME measurement.

The results with EME for different methods can be found in Table 1.

<table>
<thead>
<tr>
<th>Images</th>
<th>Original</th>
<th>Histogram equalization</th>
<th>CLAHE</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image 1</td>
<td>2.93</td>
<td>13.59</td>
<td>6.66</td>
<td>29.15</td>
</tr>
<tr>
<td>Image 2</td>
<td>4.36</td>
<td>19.77</td>
<td>9.76</td>
<td>39.34</td>
</tr>
<tr>
<td>Image 3</td>
<td>4.65</td>
<td>24.92</td>
<td>17.92</td>
<td>31.93</td>
</tr>
<tr>
<td>Image 4</td>
<td>9.07</td>
<td>11.87</td>
<td>9.94</td>
<td>54.59</td>
</tr>
</tbody>
</table>
IV. CONCLUSION

We present novel enhancement technique based on a new application of contrast limited adaptive histograms on transform domain coefficients called logarithmic transform coefficient adaptive histogram equalization with local and global processing. This strategy for image enhancement allows getting more contrast and detailed underwater image with the following properties: irregular lighting and brightness gradient. Comprehensive validation experiments performed on real underwater images reveal that the proposed method performs better than the current state-of-the-art.

ACKNOWLEDGMENT

This work was supported by Russian Ministry of Education and Science in accordance to the decree of the Government of April 9, 2010, №218 (project number 03.G25.31.0284).

REFERENCES

Computational Graphology Applied to Handwriting Images

Afnan H. Garoot  
Center for Pattern Recognition and Machine Intelligence  
Concordia University  
Montreal, Canada  
aaf_garou@encs.concordia.ca

Maedeh Safar  
Center for Pattern Recognition and Machine Intelligence  
Concordia University  
Montreal, Canada  
ma_safar@encs.concordia.ca

Nicola Nobile  
Center for Pattern Recognition and Machine Intelligence  
Concordia University  
Montreal, Canada  
nicola@cenparmi.concordia.ca

Ching Y. Suen  
Center for Pattern Recognition and Machine Intelligence  
Concordia University  
Montreal, Canada  
suen@cs.concordia.ca

Abstract— Graphology is a technique used to assess the writer’s personality traits from his/her handwriting features. Manual feature extraction and analysis is a time consuming and labor intensive task. Therefore, computational graphology systems have been developed by researchers to overcome these issues. In this paper, we present a list of the latest state-of-the-art on computerized graphology systems in terms of their characteristics, their performance, and the image data used. We also illustrate how a real system works on several groups of famous people and professionals, and different languages (English, French, and Chinese) for one particular writer over a period of four decades.

Keywords— handwriting; graphology; recognition; feature extraction; personality trait; handwriting analysis; computerized graphology system.

I. INTRODUCTION

The study of a person’s character and personality from his/her handwriting is known as Handwriting Analysis or Graphology. Graphologists use handwriting as an indication of personality trait represented by neurological patterns in the brain. It is done first by extracting some specific features from the handwriting samples. Then the extracted features are interpreted based on some graphological rules. Handwriting analysis done by a computer helps graphologists to reveal and identify the personality of a person in an automated way. Development of computerized graphology systems is an active research area. Therefore, the goal of this paper is to present a literature review on their methodologies and achievements and several innovative experiments to discover the personality traits of different groups of people and their variations over different languages and time.

The paper is organized as follows: section II describes graphology and handwriting features used in handwriting analysis. Section III explains the problems associated with extracting and analyzing features manually and describes computerized graphology system and its advantages. Section IV presents a state of the art on works related to computerized graphology systems. Section V presents an illustration of a real computerized graphological system in analyzing the character of a group of professionals and of a particular person in writing three different languages over a long period of more than four decades. Section VI discusses the most recent papers. Section VII summarizes our findings and presents future work.

II. GRAPHOLOGY

In [1], Graphology is considered as a modern form of psychology that reveals personality traits, including emotional outlay, fears, honesty, defenses and others, from the individual’s handwriting, but not identifying the writer’s age, race, gender, religion, or nationality. In other words, it is a technique used to evaluate and interpret the character of the writer from his/her handwriting [2]. Nowadays, there are different formal tests on the market used to discover someone’s personality such as aptitude studies, psychometric tests, and others. However, by comparing the practicality of these different diagnostic methods, it has been found that graphology is the fastest way.

There are many uses and applications of handwriting analysis. The followings are the most popular applications used today: dating and socializing, roommates and landlords, entertainment at parties and conventions, business and professional, employee hiring and human resources, police profiling, self-improvements and professional speakers, counsellors, therapists, and coaching applications [1].

III. COMPUTERIZED GRAPHOLOGICAL SYSTEM

In 1995, Sheikholeslami, Srihari and Govindaraju [4] mentioned a number of problems associated with extracting features manually from handwriting samples. These problems are: (1) it is a tedious, subjective, and error prone task, (2) the same features of the same handwriting sample might be extracted differently by different graphologists, (3) the analysis of graphologists might be influenced by the content of handwriting, and (4) validity of graphology rules used in analyzing features is still in question. Therefore, a computerized graphology system has been developed in order to overcome these problems.

A computerized graphology system helps graphologists to extract handwriting features and analyze them faster and more precisely using computers. It takes a handwriting sample as an input and produces a personality description of the writer as an output. It usually consists of four main modules which are: 1) scanning, 2) preprocessing, 3) feature extractions, and 4) analyzing features. [7] mentioned a number of advantages of a computerized handwriting analysis, which are: fast, accurate, identifies the handwriting better than visual assessment, efficient, and devoid of human errors. It is being applied in
different areas such as business, psychiatry, medicine and criminology [3].

Previously, various researchers have studied handwriting analysis with the aid of a computer. Several reviews are focused on existing technologies and implementation of different methodologies in graphology [9], various types of features [13] and applications of personality assessments [14]. However, this paper reviews related works on computerized graphology systematically in terms of preprocessing operations, extracted features, analysis techniques, used data bases and experimental results.

IV. RELATED WORKS ON COMPUTERIZED GRAPHOLOGY SYSTEMS

Early research on computerized graphology system was done in 1995 by Sheikholeslami et al. [4]. They developed a system called Computer Aided Graphology (CAG). They performed three preprocessing operations on scanned images, which are: removing digitization artifacts, thresholding using Osu method [5], and guide line removal using Hough Transform algorithm [6]. They segmented the input image into lines in order to extract features. Based on information collected from line images, the following features were extracted: page left margin, page right margin, page bottom margin, line spacing, line direction, slant, and upper, middle, and lower zone ratios. The system utilized syntactic pattern recognition to interpret features. As a result, most of the features had the same graphologist interpretations, but for some of them they were different. In general, the output of the system was consistent with graphologists’ interpretation. Since then, many studies have been carried out by researchers in different parts of the world.

The recent research in 2016, regarding automated graphology system reported a method developed for segmentation, baseline recognition and pressure of the writing. Bal and Saha [15] improved techniques of horizontal and vertical projections to decline incorrect line segmentation due to overlapping. Thus Lines and words segmentation and also skew normalization methods were developed in this research. Furthermore, the proposed methods are able to predict personality of the writer through baseline and pressure of the writing. A list of some works done since 1995 is presented in Table 1. A full list can be found in [19] (ICDAR paper Garoot et al.).

<table>
<thead>
<tr>
<th>Authors and Year</th>
<th>Preprocessing Operations</th>
<th>Extracted Features</th>
<th>Analysis Technique</th>
<th>Data Base</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sheikholeslami et al. (1995)</td>
<td>Removing digitization artifacts, thresholding, and guide line removal</td>
<td>Page left margin, page right margin, page bottom margin, line spacing, line direction, slant, and upper, middle, and lower zone ratios</td>
<td>Syntactic pattern recognition.</td>
<td>25 handwriting samples</td>
<td>The output of the system was consistent with graphologists’ interpretation</td>
</tr>
<tr>
<td>Matalib et al. (2007)</td>
<td>Digitization and normalization, scaling, noise removal, binary conversion and segmentation</td>
<td>Small letter ‘t’ character</td>
<td>Two back propagation neural network</td>
<td>50 respondents to questionnaires</td>
<td>average accuracy for the small letter “t” recognition is 90.27% and for the ambition recognition is 60%</td>
</tr>
<tr>
<td>Ahmed &amp; Mathkour (2008)</td>
<td>Not mentioned</td>
<td>Slants, base line, speed, size, continuity, form, arrangement, and pressure</td>
<td>A rule-based system that is equipped with the conventional pattern recognition techniques as well as an inference engine</td>
<td>35 students’ handwriting samples</td>
<td>Not mentioned, but in general the results were encouraging</td>
</tr>
<tr>
<td>Prasad et al. (2010)</td>
<td>Not mentioned</td>
<td>Size of letters, slant of letters and words, baseline, pen pressure, spacing between letters, and spacing between words</td>
<td>Support Vector Machine (SVM)</td>
<td>100 handwriting samples</td>
<td>94% of accuracy rate with RBF kernel</td>
</tr>
<tr>
<td>Raut &amp; Bobade (2014)</td>
<td>Smoothing by removing the noise, line segmentation, letter segmentation and word segmentation</td>
<td>Pen pressure, baseline, size of the letters, spacing between letters, spacing between the words, slant of words and letters, margins and speed of the writing</td>
<td>Support Vector Machine (SVM)</td>
<td>100 handwriting samples and 50 words in a plain paper for each sample</td>
<td>various kernels such as linear and polynomial were tested with SVM; however, the RBF shows a better accuracy, near 90%</td>
</tr>
<tr>
<td>Hashemi et al. (2015)</td>
<td>Pen width extraction, noise and scratch removal</td>
<td>Right and left page margins, word expansion, letter size, line and word spacing, line spacing, word spacing, line skew, the ratio of vertical to horizontal elongation of words, and slant</td>
<td>Support Vector Machine (SVM)</td>
<td>120 handwriting samples</td>
<td>Not mentioned, but in general the system showed promising results</td>
</tr>
<tr>
<td>Bal and Saha (2016)</td>
<td>Noise removal, thresholding, line segmentation, skew normalization, measures width of inter-word and intra-word gaps, normalize the skew of the segmented words</td>
<td>Baseline and writing pressure</td>
<td>Rule-Based system</td>
<td>IAM database over 550 text images containing 3800 words and some sample handwriting image which are written by the different writer on the different background</td>
<td>Accuracy rate of lines segmentation and words segmentation is 95.65% and 92.56%, respectively. Moreover, 96% of lines and words were normalized perfectly with tiny error rate</td>
</tr>
</tbody>
</table>
V. Experiments with a Real Computational Graphology System

To demonstrate how a computerized graphology program works, we have conducted several experiments using a computer program developed by Ahmed and Mathkour [3]. It is a rule-based system that is equipped with the conventional pattern recognition techniques as well as an inference engine with encouraging results. Based on the measurement of slants, base line, speed, size, continuity, form, arrangement, and pressure, the software produced the following traits about the writer: emotion, mood, self-confidence, coherence of thoughts, strength, and organization.

A. Analysis of famous people and professionals

We ran our Graphology software on several famous (and infamous) people. We must remind the reader that the results may not reflect the true nature of the person, but rather their state of mind at the time the document was written. This is particularly true of presidents where successes and setbacks can change day to day.

Our Graphology software was trained to calculate the following six psychological/emotional features based on a handwriting sample:

1. **Emotions**: Restraint to Expressive
2. **Mood**: Bad to Good
3. **Self Confidence**: Low to High
4. **Coherence of thoughts**: Low to High
5. **Strength**: Low to High
6. **Organization**: Low to High

When a writer's emotional frame of mind is close to "Restraint", it indicates that the writer is unemotional, dispassionate, or is holding back their true feelings. What they are writing and thinking may actually be contradictory. "Mood" reflects the writer temperament, from bad to good. A writer's self-confidence indicates the certainty in oneself, usually on the topic that is being written. Coherence of thoughts reveals a writer's sequence of thinking. That is, if the writer has a structured line of thinking from one thought to another as he/she is writing. This line of thinking can be broken if a writer has other thoughts in mind, has erratic thoughts, or is "daydreaming". Strength indicates how controlled the writing is and organization is a measure of how well organized the writer is. Each trait is given a numeric value from [-1.0, 1.0] representing the lowest and highest feature types respectively.

a) Results

We analyzed the six features on four types of people: World Leaders, Royalty, Painters, and Criminals. The people chosen span different centuries, continents, and genders.

We can see from Table 2 that, for the most part, the features appear to be consistent among the members within each group.

<table>
<thead>
<tr>
<th></th>
<th>Emotions</th>
<th>Mood</th>
<th>Self Conf.</th>
<th>Coh/Thoughts</th>
<th>Strength</th>
<th>Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>World Leaders</td>
<td>23</td>
<td>0.28</td>
<td>0.22</td>
<td>-0.09</td>
<td>0.01</td>
<td>-0.33</td>
</tr>
<tr>
<td>Royalty</td>
<td>5</td>
<td>-0.22</td>
<td>0.11</td>
<td>-0.09</td>
<td>0.01</td>
<td>0.76</td>
</tr>
<tr>
<td>Painters</td>
<td>14</td>
<td>0.51</td>
<td>0.24</td>
<td>-0.11</td>
<td>0.01</td>
<td>0.26</td>
</tr>
<tr>
<td>Criminals</td>
<td>19</td>
<td>0.45</td>
<td>0.17</td>
<td>-0.15</td>
<td>0.03</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Features that have a low variance (less than 0.05) indicate that that particular feature average value is a common trait among that group. For example, World Leaders tend to have a slightly less than normal mood (-0.09). Royalty and Criminals tend to have above normal self-confidence (0.76 and 0.34). Painters appear to be greatly disorganized (-0.88).

Fig. 1 shows the average trait values for the four groups. We can see that Royalty appears to be emotionally restrained compared to the other three groups of people. They do, however, seem to have better coherence of thoughts and much higher self-confidence than the rest. Finally, Criminals have a much higher strength than all other groups.

B. Analysis of One Person’s Handwriting Across Languages and Over time

We are interested in knowing the results for the same person's handwriting in different languages: English, French, and Chinese over a long period of 4 decades in writing (a) English, (b) French, a language with Latin roots as English, and (c) Chinese, a language which differs considerably from English and French. Due to limitation of space, we shall present only some typical results as illustrations.

a) English over 4 decades from 1963 to 2007

The charts produced by the software are presented in Fig. 2 from which we can deduce the following: Emotion: consistent, even different materials were written, Mood: fairly consistent, Self Confidence: consistent, Coherence of thoughts: fairly consistent, Strength: variations, and Organization: very consistent.
b) French over 13 years from 1981 to 1994

From Fig. 3 we can deduce the following: Emotion: consistent, Mood: consistent, Self-Confidence: very consistent, Coherence of thoughts: consistent, Strength: very consistent, and Organization: very consistent.

c) Chinese over 20 years from 1990 to 2010

From Fig. 4 we can deduce the following: Emotion: consistent, Mood: very consistent, Self Confidence: consistent, Coherence of thoughts: consistent, Strength: very consistent and Organization: very consistent.

From the above results, one can draw the following conclusions:

- This program predicts well the following characters of the writer: mood, self-confidence, coherence of thoughts, and organization.

- The program works across languages which have similar structures, e.g. English and French, but varies considerably when applied to a language with a very different structure.

- The handwriting of an adult person does not change much over the years, except in strength which may also depend on the writing instrument and paper used.

- The Chinese language is more structured, e.g. character by character which can be enclosed in boxes. Hence a Chinese text passage is like a structure formed by lining up boxes, giving the impression of a better organization in the writing. Hence a different set of features should be extracted, e.g. those listed in refs. [20] [21] [22].

VI. DISCUSSION

As mentioned earlier the input to the computerized graphology system is a handwriting sample. This sample would be taken offline or online. In offline systems, the sample is taken on a plain white A4 size paper. Then, it is scanned using a scanner in JPEG or BMP format or it can be a photo taken by a camera [13]. However, in online systems the user can directly write onto the scratch pad using a special pen or stylus or any other devices [13].

After the acquisition of handwriting samples, the image of sample is given to the next phase called preprocessing. Based on the recent works, this process includes a number of operations such as normalization, binarization, noise removal,
smoothing, pen width extraction, and scratch removal [8] [11] [12] [15]. In order to get only the text and remove all unwanted information, the image is then segmented. The segmented image acts as input to the next phase which is feature extraction [17].

In order to define the individual’s personality traits, handwriting features should be extracted. It extracts several writing features such as baseline, slant, size of the letters, pen pressure, spacing between the words, spacing between the letters, zone, speed, margin etc. We can classify the writer’s character in a particular class depending on these extracted features [13]. Based on the recent works, different methods can be used to extract handwriting features [10].

After defining the feature vectors, these features are classified using any classification method. This process is done in the next phase called classification which is usually performed by comparing the feature vectors corresponding to the input feature with the representative of each class of personality trait. However, before doing this a number of training samples should be provided. On the basis of recent works, a number of classification methods have been used widely in graphology which are: Support Vector Machine (SVM) [11], Artificial Neural Networks (ANN) [12], and rule-based systems [15].

There are other methods in classification which are: Statistical methods and template matching. Statistical methods determine which category the given pattern belongs to by making observations and measurement processes and preparing a set of numbers used to prepare a measurement vector. The most common statistical methods are k-nearest neighbour (K-NN), Bayesian classification, Quadratic Discriminant Function (QDF), Linear Discriminant Function (LDF), Euclidean distance, cross correlation, Mahanalobis distance. Regularized Discriminant Analysis (RDA). Template matching is one of the simplest approaches. It is done by comparing the given pattern with the stored patterns [18].

VII. CONCLUSION

Handwriting consists of a bunch of written symbols which can be considered as brain writing. More specifically, the handwriting is an indicative tool to reveal the personality of the writer [16]. According to prior studies we have identified several important features that have been used in computerized graphology such as page margin, baseline, line and word spacing, line direction, slant of words and letters, size of letters, pen pressure, speed of the writing, the letter ‘t’, letter ‘i’, letter ‘f’, the lower loop of letter ‘y’, etc. In addition, it has been observed that different analysis methods such as support vector machine (SVM), artificial neural networks (ANN), rule-based systems have been used widely in computational graphology. However, there is a powerful analysis method, called deep learning which has not been applied yet in this area. Therefore, applying deep learning method for designing a fully automated graphology system with a high accuracy rate should be considered as future work. The current trend of computerized graphology shows that applying a combination of analysis methods [10] and using features of signature besides handwriting features [10] resulted in remarkable accuracy. Moreover, the quantity of training data has an impact on achieving better results [8].

In this paper, several experiments have been conducted on the handwriting samples of a group of professionals and of a particular person in writing three different languages over a long period. The results offer some new insights on how a real computational graphology system works.

ACKNOWLEDGMENT

This research was supported by a research grant from NSERC, Natural Sciences and Engineering Research Council of Canada.

REFERENCES


Deep Learning is not a Matter of Depth but of Good Training

1st Björn Barz
Computer Vision Group
Friedrich Schiller University Jena
Jena, Germany
bjoern.barz@uni-jena.de

2nd Joachim Denzler
Computer Vision Group
Friedrich Schiller University Jena
Jena, Germany
joachim.denzler@uni-jena.de

Abstract—In the past few years, deep neural networks have often been claimed to provide greater representational power than shallow networks. In this work, we propose a wide, shallow, and strictly sequential network architecture without any residual connections. When trained with cyclical learning rate schedules, this simple network achieves a classification accuracy on CIFAR-100 competitive to a 10 times deeper residual network, while it can be trained 4 times faster. This provides evidence that neither depth nor residual connections are crucial for deep learning. Instead, residual connections just seem to facilitate training using plain SGD by avoiding bad local minima. We believe that our work can hence point the research community to the actual bottleneck of contemporary deep learning: the optimization algorithms.

Index Terms—deep learning, stochastic gradient descent, learning rate schedule, residual networks, cyclical learning rates

I. INTRODUCTION

In the past few years, deep learning using convolutional neural networks (CNNs) has continuously advanced the state-of-the-art in most image processing and other machine learning applications: The first notable success of modern deep learning was in 2012, when a CNN often called AlexNet [1] won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This network, which is composed of 8 trainable hidden layers, was outperformed two years later by the so-called VGG architecture [2], which roughly doubles the number of trainable layers to between 16 and 19.

Likewise, VGG was surpassed in 2015 by deep residual networks (ResNets) [3] with up to 152 trainable layers. To avoid vanishing gradients during backpropagation through all those layers, ResNet contains skip-connections that add the output of earlier layers to that of some later layers. On smaller benchmark datasets such as CIFAR-100 [4], where training does not take weeks, the trend of increasing the number of hidden layers has continued and lead to ResNets with up to 1001 layers [5].

The paradigm of creating CNNs as deep as possible arises from the fact that the degree of abstraction of the learned image features generally increases with the depth of the network: The first layer learns to recognize simple visual features such as edges, while activations of later layers often refer to semantic object parts [6]. Therefore, it is often postulated that the abstraction and generalization capability of CNNs is a monotonically increasing function of its depth, i.e., the number of trainable layers [2], [3], [5].

However, there also are other important properties of CNNs such as their width (i.e., the number of channels per layer) and the optimization algorithm used during training. Both VGG and ResNet have originally been trained using stochastic gradient descent (SGD) with momentum, starting with a manually selected learning rate that is reduced by a factor of 10 each time the performance on a held-out validation set reaches a plateau [2], [3].

In this work, we show that a shallow but wide network with as few as 11 trainable layers and no residual connections can achieve the same performance on CIFAR-100 as a thin deep residual network with 110 parametric layers when trained using a more sophisticated learning rate schedule. Moreover, our network requires less run-time during inference and training due to its reduced depth.

This finding provides evidence that the depth of a CNN is less important than commonly thought. Intuitively, a certain depth is required for sufficient abstraction capability, but more layers do not seem to contribute significantly to this aspect.

The main contributions of our work are as follows:

1) We propose a novel strictly sequential CNN architecture with 11 layers that can achieve a performance competitive to a 10 times deeper ResNet-110 on CIFAR-100.
2) We compare the effects of two recent learning rate schedules (CLR and SGDR) on sequential and residual networks.
3) This leads to the insight, that ResNets are not necessarily much more powerful than wide shallow sequential networks, but just easier to train with plain SGD.

Although our experiments are not meant to beat the current state-of-the-art in image classification, we are convinced that this is an interesting finding and hope that our work can help pointing the research community to the real bottleneck of deep learning, which is neither insufficient depth nor a lack of architectural complexity, but poor optimization algorithms.
II. RELATED WORK

The effects of depth and width of residual networks on their performance have already been investigated by Zagoruyko and Komodakis [7], who have shown that increasing the width of ResNets while decreasing their depth improves classification performance. They state “that the main power of deep residual networks is in residual blocks, and that the effect of depth is supplementary” [7, p. 3].

Our findings take this insight a step further and suggest that the residual connections do not increase the representational power of the network either, but just facilitate training using plain SGD. However, strictly sequential wide shallow networks without residual connections can achieve similar performance when trained with more sophisticated optimization algorithms.

Our work is furthermore similar in spirit to the analysis of Melis et al. [8], who have shown that rather shallow LSTMs can outperform deeper and more recent neural language models thanks to thorough hyper-parameter tuning and regularization.

We present analogous findings for the computer vision domain, but achieve this by focusing on sophisticated learning rate schedules instead of hyper-parameters.

III. NETWORK ARCHITECTURE

The convolutional part of our wide shallow CNN is composed of 4 sequential blocks of convolutional units, as outlined in Fig. 1. Each unit consists of a convolutional layer followed by a ReLU activation and batch normalization [9].

The first block consists of 2 units with 64 channels each, the second and third block of 3 units with 128 and 256 channels, respectively, and the last block of a single unit with 512 channels. Sub-sampling by a factor of 2 along the spatial axes is performed between the blocks using average pooling and the last block is followed by global average pooling.

The subsequent fully-connected part of the network consists of a fully-connected layer with 512 channels followed by ReLU and batch normalization, and a final fully-connected layer with softmax activation, whose number of channels equals the number of classes.

This architecture is similar to the VGG networks [2], but with a few important differences:

1) We use batch normalization [9] between layers to mitigate the problem of exploding and vanishing gradients.
2) We use average pooling instead of maximum pooling between convolutional blocks.
3) Like ResNet [3], we use global average pooling after the last convolutional layer instead of fully-connected pooling. This enables the network to handle input images of varying size.

IV. OPTIMIZATION

Both VGG [2] and ResNet [3] have originally been trained using SGD with momentum and a simple learning rate schedule: Starting with a small initial learning rate to avoid exploding gradients (“warm-up phase”), the learning rate is set to a higher value after a few iterations and then decreased by a factor of 10 either after a handpicked number of iterations or when the performance on a held-out validation set has not improved significantly during the last few epochs.

However, this approach is prone to getting trapped in bad local minima, since the learning rate is only decreased, but never increased. Two approaches have recently been proposed for overcoming this issue by periodically increasing and decreasing the learning rate. These are reviewed in the following subsections and depicted in Fig. 2, along with the schedule that He et al. [3] used for training ResNet-110.
A. Cyclical Learning Rates (CLR)

Cyclical Learning Rates (CLR) [10] start with a small base learning rate \( lr_{\text{min}} \) and increase it linearly during a fixed number \( s \) of iterations ("step size") up to a certain maximum learning rate \( lr_{\text{max}} \). Thereafter, the learning rate is decreased again over the same number of iterations until reaching the base learning rate. After that, the next cycle of increasing and decreasing the learning rate begins. Formally, the learning rate \( clr(t) \) at iteration \( t \) is

\[
clr(t) = lr_{\text{min}} + (lr_{\text{max}} - lr_{\text{min}}) \cdot \left(1 - \frac{t}{s} - 2 \left\lfloor \frac{t}{2s} \right\rfloor - 1\right),
\]

where \( s \) is the length of the current cycle and \( t \) is the number of epochs passed since the beginning of the cycle.

Starting with a certain initial cycle length \( S_0 \), the length \( S_i \) of cycle \( i \) results from multiplication of the length of the previous cycle with a constant factor:

\[
S_i = \sigma \cdot S_{i-1}.
\]

The motivation behind increasing the learning rate from time to time is to enable the learning process to escape from bad local minima.

The authors also proposed a variant where the maximum learning rate is reduced after each cycle. In this work, we achieve this in a slightly different way by applying a global learning rate decay so that the maximum learning rate at the end of the training is \( \delta \) times lower than the initial maximum learning rate:

\[
\tilde{clr}(t) = \frac{clr(t)}{1 + (\delta - 1) \cdot \frac{t}{t_{\text{max}}}},
\]

where \( t_{\text{max}} \) is the maximum number of iterations after which the training procedure is terminated.

B. Stochastic Gradient Descent with Warm Restarts (SGDR)

A similar approach is taken by SGD with Warm Restarts (SGDR) [11]. In contrast to CLR, the learning rate is not decreased linearly but according to cosine annealing. It is furthermore not increased smoothly at the end of each cycle, but in an instant. Formally, the learning rate \( \text{sgdr}(\epsilon) \) during epoch \( \epsilon \) is

\[
\text{sgdr}(\epsilon) = lr_{\text{min}} + \frac{1}{2} (lr_{\text{max}} - lr_{\text{min}}) \left(1 + \cos \left(\frac{\epsilon_i}{S_i} \pi\right)\right),
\]

where \( S_i \) is the length of the current cycle and \( \epsilon_i \) is the number of epochs passed since the beginning of the cycle.

Since SGDR starts with a very high learning rate, we combine it with gradient clipping [12] to avoid exploding gradients: The maximum norm of the gradients propagated back through the network is restricted to 10.0.

V. Experiments

A. Setup

We compare the performance of our strictly sequential VGG-like network architecture and deep residual networks (ResNets) trained with different learning rate schedules on the CIFAR-100 dataset. CIFAR-100 is a heavily benchmarked dataset consisting of images from 100 different classes. Each class comprises 500 training and 100 test images, leading to a total amount of 50k training and 10k test images. All images are in RGB color format and of size 32 × 32 pixels.

We compare 3 different learning rate schedules for training our plain 11-layer network, referred to as “Plain-11” in the following, and ResNet-110, which has ten times more parametric layers:

- The handcrafted learning rate schedule used by He et al. [3] for training ResNet-110: Starting with a learning rate of 0.01 to avoid divergence, the learning rate is increased to 0.1 after the first epoch and then divided by 10 after 80 and 120 epochs. Training is terminated after a total of 164 epochs.
- CLR with a step size \( s \) corresponding to 10 epochs (i.e., \( s \) is the number of batches per epoch multiplied with 10), a minimum learning rate of \( lr_{\text{min}} = 10^{-5} \), a maximum learning rate of \( lr_{\text{max}} = 0.1 \) and a final learning rate decay of \( \delta = 0.1 \).
The number of training epochs for CLR and SGDR has been chosen to be near 164 for being comparable with the ResNet schedule, but so that the learning rate arrives at $l_{r_{\text{min}}}$ during the last epoch. This is important, because the performance of the network often behaves unstable while training with higher learning rates.

We use a mini-batch size of 100 images—as opposed to the most commonly used batch-size of 128—because it divides the total amount of training images evenly.

### B. Comparison of Architectures

For all combinations of network architectures and learning rate schedules, we trained 3 networks with different random weight initializations and report the average error rate on the test set in Table I. It can be seen that our strictly sequential network does not achieve the performance of ResNet-110 when trained with the same hand-tuned learning rate schedule. Using SGDR, however, improves its validation error rate by 5%, while the improvement for ResNet-110 is only 2%. This indicates that the residual connections used in ResNet already facilitate training, so that different learning rate schedules do not add much benefit.

On the other hand, more sophisticated training procedures allow us to train a shallow sequential network that achieves competitive performance compared to ResNet-110. At the same time, the architecture of our model is less complex and can be trained 4 times faster, which is an important aspect in the compute-intensive realm of deep learning.

### C. Comparison of CLR and SGDR

It is also worth noting that SGDR achieved a better performance than CLR for both network architectures, while CLR even performed worse than the handcrafted learning rate schedule for training ResNet-110. Of course, our experiments do not include a sufficient number of network architectures and datasets to claim that one method for scheduling learning rates would perform better than the other, but it at least provides some evidence that SGDR is a good choice.

On the other hand, CLR provides more flexibility for choosing the total number of training epochs, since all cycles have the same, comparatively small length. Since the length of the SGDR periods grows exponentially, one cannot just add a few but only a lot more training epochs when using SGDR.

### VI. Conclusions

Wide residual networks [7] have already demonstrated that the depth of a CNN is much less important than previously thought and that a wide shallow ResNet can outperform thin deep ResNets significantly. We have shown that a plain VGG-like network without any residual connections can achieve the same performance as a 10 times deeper ResNet when using sophisticated learning rate schedules such as CLR or SGDR, while being 4 times faster to train.

This provides evidence that the skip-connections used in ResNets do not increase the representational power of the network, but rather lead to loss functions with fewer bad local minima and hence facilitate learning using SGD.

The main problem of modern deep learning is, thus, neither insufficient depth nor a lack of structural complexity, but the deficiencies of the ubiquitous stochastic gradient descent (SGD) optimization algorithm and its many variants, which easily get stuck in bad local minima.

While there is a large amount of ongoing work about new network architectures, types of layers, and activation functions, we believe that a large fraction of the potential of most neural networks is wasted due to poor optimization algorithms. More complex learning rate schedules such as the cyclical learning rates [10], [11] discussed in this paper or the parameter-specific learning rates used by AdaGrad [13], RMSProp [14], and Adam [15] are a step towards getting to the root of the problem, but are still based on SGD and backpropagation.

While the shape of the categorical cross-entropy loss function might be one part of the problem, Nguyen et al. [16] have shown that even in the face of convex loss functions, SGD makes steep improvements in the beginning, but then stagnates at a sub-optimal state. Thus, the next big breakthrough in deep learning might rather be achieved by thinking out of the box and applying completely novel optimization algorithms for training artificial neural networks.

### References


Dynamic-HOG Descriptor for Structured Object Recognition: Case Study on Coins

Ali K. Hmood, Ching Y. Suen, Louisa Lam
Centre for Pattern Recognition and Machine Intelligence (CENPARMI)
Department of Computer Science and Software Engineering
Concordia University
Montreal, Canada
{a_alfraj, suen, llam}@encs.concordia.ca

Abstract—This paper presents a novel and robust method for structured object recognition based on improvements to the widely-used Histogram of Oriented Gradients (HOG) method. The main drawback of HOG is the exhaustive search mechanism using a fixed window that slides through the whole image to recognize objects. This research proposes a dynamic-HOG method that works on locating and recognizing structured objects in images dynamically w.r.t. the object size. Additionally, this method eliminates the exhaustive search by locating the objects first and then recognizing them which saves a great deal of time. The method works on structured objects due to its dependency on finding the right size of objects on which to fit the dynamic window. Unlike other research works, this research proposes a new solution for coin recognition by considering small parts of the coins, characters, only instead of the whole image. The character orientation is challenging due to the circular shape of coins, therefore we transform the coin from its circular shape into rectangular shape to facilitate character segmentation. A set of histograms of oriented gradients is extracted for every character based on its height and width. Also a dimensionality reduction method is applied to reduce the size of feature vector and a multiclass SVM is used to classify the resulting character set. Our method is evaluated on coins from 4 countries with different background complexity. The proposed method achieved precision and recall rates as high as 98.08% and 98.23% respectively; which demonstrate the effectiveness and robustness of the proposed method.

Keywords—Computer Vision, Dynamic Histogram of Oriented Gradients, Character Segmentation, Coin Recognition.

I. INTRODUCTION

Object recognition is a long established research direction that received numerous research works to enable computers better understand images. Several methods have been discussed in the literature to extract different features from images to recognize objects accurately. Histogram of Oriented Gradients (HOGs), proposed by Dalal and Triggs [1] in 2005, received a raising attention from researchers due to its better representation of image information. The HOG has been applied to images of different objects and reported a highly accurate object representation e.g. cars, pedestrian, and faces. However, the main obstacle of using HOG is the exhaustive search process throughout the whole image using a small fixed window to locate objects. [1] stated that adding windows of different scales to capture bigger objects in images would improve results on one hand and considerably increase the computation cost on the other. Therefore deciding the optimal window size is a major drawback of HOG method. There are few researchers tried improving the exhaustive search process or at least better deciding the accurate window size needed for objects [2]. To the best of our knowledge, this is the first paper to address the structured objects recognition with no a priori knowledge, sliding window, or classifier based localization.

On the other hand, a lot of research works have been proposed for coin recognition systems in recent years. The purpose of introducing the computer vision approaches in coin recognition rather than physical characteristics is grounded on different factors, such as several coins around the world are sharing the same set of physical characteristics (i.e. weight, thickness, size, and diameter) and also use the same metal type. Therefore, studying the design features on the coin surface can prevent falsification and improve the classification of coins. Coins generally consist of two parts at each side (obverse and reverse sides) where the middle part usually contains a symbol and the outer part contains the characters. Characters are minted in all coins and considered as one of the main parts of every coin. Unlike character recognition from documents, coins are (1) subject to severe degradation due to circulation in daily use; (2) no prior knowledge of character orientation; (3) salient of minted characters on the coins surface varies which affect the stroke sharpness and width; (4) different lighting sources highly affect the character appearance due to highlight and shadow variations; (5) identical color of the characters as the background. The degradation diverges from worn out edges to detached strokes into invisible characters. With those given challenges, we used coin recognition problem as an application to the proposed method.

In this research, we propose a new method for structured object recognition that is completely based on the exact size of the object. Coins are used as an application to verify and test the results. The proposed method was evaluated on different coin images including a publicly available dataset.
II. RELATED WORK

Object recognition is a well-established research field that aims to determine the objects within an image, in other words, better understanding of images. The most common approach is to extract features from the whole coin image (mostly using sliding window) and use complex and expensive classifiers [3]. There are some research papers discussed the possibility of limiting the number of candidate locations of objects using segmentation which was found to have limited success scenarios due to overlapping issues [4] or bounding box which has better results but still deciding the right size of bounding box is the main challenge [3]. However, few researchers have tried improving HOG method by specifying a smaller grid placed over the object and apply the HOG on the grid to improve prediction results and reduce the computational cost. Those methods reported a better results than using the sliding window throughout the whole image.

[5] proposed an extension method to HOG by concatenate another HOG feature descriptor to the traditional descriptor (by combining the HOG from different fixed sized blocks). The new HOG feature descriptor obtained from a dense-grid bounding box over landmark areas. Those landmarks are decided based on cascaded deformable shape model. Additionally, [6] proposed a similar extension to HOG method by detecting the face first and then the relative place of the face objects i.e. eyebrow, noise, and mouth. The HOG feature is then applied to those areas and SVM classifier used to classify the face expression. Lekdioui et al. [7] proposed another method for detection of facial expression using different local feature descriptors from 7 areas in the face that were selected or located using IntraFace method. Yet, those methods are either very specific to work on the used dataset due to the assumptions and a priori knowledge or not fully dynamic in term of changing the grid size according to each object from the image.

Additionally, coins have been receiving increased attention by machine vision and intelligence researchers in recent years. Numerous studies have been presented in the literature for coin recognition, grading, and authentication systems. [8] proposed a solution to spot the text on ancient Roman coins. The SIFT descriptor is used to find key points of region of interest (ROI) considered as candidate text location. The authors extended their work to recognize the text on the ancient Roman coins [9]. The extended work took advantage of the limited dataset in their experiments to manually specify location of text on the coin surface. The SIFT descriptor is obtained for each candidate location and compared against the training set of SIFT descriptors. In both papers, the authors used an SVM to train a manually selected key point (centered to each character) and to test the candidate locations. [10] introduced a combined method of image matching and text recognition to classify ancient coins. The authors used the same scheme in their paper [8] for text recognition and added the image matching to improve the coin recognition rate. Unlike previous works, [11] was the first to present character extraction from coins. A number of small windows are placed on the coin image and for each window the HOG-like descriptor is found and compared against a manually annotated dataset to recognize characters. However, the evaluation of the system was carried out on 25 coin images belonging to ancient Romans, and more specifically the system covered only the ancient Denarrii coin. The recognition rate of the 25 coin dataset was not reported in this paper. [12] proposed a character segmentation method from coins though no character recognition is studied in their paper. On the other hand, [13] proposed a solution for coin date detection. The proposed system requires a priori knowledge about the coin to locate the date. The histogram is computed for gradients of each candidate location and is then compared to the mean histogram of synthetic date models.

However, the above mentioned methods and many more have focused on either extracting features from the whole coin surface using various feature extraction methods or semi-manually segmented the characters and found set of features for each. Yet, optimal selection of an object to be recognized within an image is the major challenge in object recognition. Nevertheless, locating characters on coin images is less challenging than on natural scenes images where characters size fall within specific range and the structured nature of characters in coins.

III. DYNAMIC-HOG FOR COIN RECOGNITION

Although the proposed method fits other structured objects, the focus of this paper will be on coins due to: (1) the increased demands on coin recognition in recent years have raised the need for an accurate and reliable coin recognition system, (2) the scale and rotation of coins have added extra challenge to the recognition process, and (3) the challenges of working on coin images stated earlier in Section I. Generally, coin recognition aims to classify large numbers of coins in a timely manner. Therefore, we propose a new coin recognition system that considers only certain parts from the coin i.e. the characters.

The proposed solution starts by applying the scaling algorithm and then the straightening algorithm to transform the coin into rectangular shape. Then a dynamic window, w.r.t. height and width thresholds, is placed on each candidate location. The vertical and horizontal projection profiles are computed to set the thresholds of character heights and widths. The recursive process is considered to include detached character strokes. Then gradient features of each character are extracted and the feature vector size is reduced by dimensionality reduction method. Finally, a classifier is trained and tested on the new feature vectors.

A. Scaling and Rotation Algorithms

Prior to feature extraction and classification, coin location is defined in every coin recognition method. Several papers have performed coin scaling, also called coin segmentation. The goal is to scale the coin to fit the whole image and remove the margins occurs outside the coin borders to facilitate feature extraction process (as show in Fig. 1 (c)). In this research, we perform coin scaling in three steps: (1) edge detection, (2) morphological operations, and (3) Circular Hough Transform (CHT). Fig. 1 (a), (b), and (c) illustrate the coin scaling algorithm. The scaling algorithm detects all edges $E$ in the image $I(x,y)$ using Sobel edge-detection and a dynamic threshold and the edge image $I_E$ is then dilated using a morphological
structuring element (e.g. circular shape). Then CHT works on a 3-dimensional parameter space where 2 dimensions represent the circle center \((C_x, C_y)\) and the third dimension is the radius \((R)\) as given in the circle equation (1) below.

\[
(x - C_x)^2 + (y - C_y)^2 = R^2
\]

(1)

The algorithm starts identifying all circular shapes in the image and defines a set of center points \((C_{x}, C_{y}) \in I(x,y)\) and radii \(R\) for each circle. The largest radius \(r \in R\) and its corresponding center point \((C_{x}, C_{y}, R)\) are selected as a candidate circle of the coin. An adaptive mask \(M\) of size \((x, y)\) is used to mask the coin. The circular shape of the coin is defined in the mask by equation (2):

\[
M(m,n) = \begin{cases} 1 & \text{iff } (R_{\text{max}} - 0.1)^2 < m^2 + n^2 < (R_{\text{max}} + 0.1)^2 \\ 0 & \text{otherwise} \end{cases}
\]

(2)

The adaptive mask is then placed over the actual coin; the coin is cropped to fit the whole image to yield a new image with centers \(I_{L}(x, y)\) set to \((C_{x}, C_{y})\) and \(R\) equals 1.

On the other hand, to achieve a rotation invariant solution, we transform the circular shape of coins into a rectangular shape using a straightening algorithm (as shown in Fig. 1 (d)). The goal of the transformation step is to overcome the orientation problem when extracting the characters and their features. Since the focus of this paper is on the characters minted on the coin, we transform the outer circle of the coin.

The straightening algorithm reads the image pixels in diagonal style and writes each pixel value into a new rectangular matrix. Given the circle equation, assume \(p(x,y)\) is a value pixel from the original image \(I(h, w)\) at \((x, y)\) coordinates \(x\) and \(y\) and \(I_{L}(x,y)\) is the \(p(x,y)\) value in the new pixel location at \(x\) and \(y\) of the straightened image \(I_{L}(h, w)\), therefore the \(x\) and \(y\) values can be found by equation (3):

\[
\begin{align*}
\bar{x} &= n * \sin(z) + (w / 2) \\
\bar{y} &= n * \cos(z) + (h / 2)
\end{align*}
\]

(3)

Where \(w\) and \(h\) are the width and height of the original image respectively. \(t\) is half the total number of columns from the original image; \(Q\) is number of rows of the outer circle of image containing the characters. Straightening the coin is an essential step for our system to simplify the character segmentation step and obtain similar gradient features for identical characters on different coins.

### 3.2 Dynamic-HOG

The Histogram of Oriented Gradients (HOGs) [1] is a rotation invariant feature descriptor that counts the occurrences of edge orientations in local neighborhood. The algorithm is based on evaluation of normalized local histograms of neighborhood pixels gradient orientations in small spatial regions, called cells. The key feature of HOG descriptor is the division of the image into cells, overlapping blocks, and normalization step. However, selecting the cell and block sizes play vital role in determining the feature representation accuracy while finding the best window size to place over a positive location that contains the object are the main challenges in the traditional HOG method [2]. On the other hand, avoiding the exhaustive sliding window originally introduced in HOG method to search for objects in the image is the focus in this research. Therefore, we introduce the dynamic-HOG method for character recognition. The key feature of the proposed method is to have a dynamic window size placed over each character w.r.t. its height and width. Characters vary in sizes where, for instance, character ‘m’ is wider than the character ‘i’ and shorter than ‘t’. [2] studied the window size and placement problem and proposed method based on different window scales. Yet, their method uses a predefined static window size (scales). We argue that the use of a dynamic window size improves the results of HOG for different characters. The dynamic-HOG can be applied to any structured objects as long as the horizontal and vertical projection profiles can be obtained accurately.

The window size selection uses the vertical and horizontal projection profiles to determine the height and width of each character. This step starts by applying a 3 x 3 Gaussian filter to remove noise and Otsu binarization algorithm to obtain a binary image. The vertical projection profile \(f(x,p(x))\) is applied to determine the characters distribution on the coin and to define the average width \(w_{\text{avg}}\) If \(v_{a,b}\) denotes a value pixel at coordinates \((a, b)\) in image \(I(x,y)\), the vertical projection value \(p(b)\) at the \(b^{\text{th}}\) column of \(I(x,y)\) can be calculated by (4). On the other hand, the horizontal projection profile \(g(y,p(y))\) is also defined to set the average height \(h_{\text{avg}}\) of connected components (blobs).

\[
p(b) = \sum_{i=0}^{w_{\text{avg}}-1} v_{a,b}
\]

(4)

After defining the \(w_{\text{avg}}\) and \(h_{\text{avg}}\) of all connected components, the spaces between characters are determined by calculating the slopes between each peak and the next valley. Slopes are defined on the vertical projection profile to determine the spaces between characters by the following formula (5):

\[
\text{Slope}_{\text{vertical}} = \frac{p(x_{\text{peak}}) - p(x_{\text{valley}})}{x_{\text{peak}} - x_{\text{valley}}}
\]

(5)

Where \(p(x)\) is the total number of white pixels in column \(x\), and \(x_{\text{peak}}\) and \(x_{\text{valley}}\) are the coordinates of columns where the peak and valley are located.

The algorithm then checks for all connected components \(C\) with height \(h_{\text{c}}\) and width \(w_{\text{c}}\) between:

\[
\begin{align*}
h_{\text{avg}} + E_1 &> h_{\text{c}} \geq h_{\text{max}} \cdot E_1 \\
w_{\text{avg}} + E_2 &> w_{\text{c}} \geq w_{\text{max}} / E_2
\end{align*}
\]

where \(E_1\) and \(E_2\) are constant thresholds. \(E_1\) is the difference between \(h_{\text{avg}}\) and \(h_{\text{max}}\), where \(h_{\text{max}}\) is the maximum height of all characters and is obtained from the horizontal projection profile; and \(E_2 = 4\). Note that the use of different arithmetic operations for defining the range of the height and the width is due to the fact that variation in height is much less than the variation in width from one character to another.
The algorithm starts by placing a bounding window around each connected component that has the height and the width within the range and discards the ones that occur outside the range. For improving window placement, we apply a recursion process to detect all connected components $\mathcal{C}$ that occur below the width and height ranges but within the maximum height $h_{\text{max}}$ of all characters that we obtain from the horizontal projection profile. These $\mathcal{C}$ commonly occur due to degradations on the coin surface which result in broken character’s stroke. For each $\mathcal{C} \subseteq \mathcal{C}$ check the distance $d$ between centers $C(\mathcal{C})$ and $C(\mathcal{C})$, where $\mathcal{C} \subseteq \mathcal{C}$, that occurs within the immediate. If $d$ is less than $h_{\text{max}} / \alpha$ for a $\mathcal{C}$ that occurs above or below the $\mathcal{C}$, or is less than $w_{\text{max}} / \alpha$ for a $\mathcal{C}$ that occurs to the left or right the $\mathcal{C}$, where $\alpha$ (equals 2 for our coin datasets obtained experimentally as discussed in Section 0) is the threshold for acceptable distance between two $\mathcal{C}$. $\mathcal{C} \subseteq \mathcal{C}$, then the two blobs are combined in one window and checked against the original range of acceptable characters width and height. However, the size of $\mathcal{C}$ and $\mathcal{C}$ should not be less than $h_{\text{avg}} / \beta$ and $w_{\text{avg}} / \beta$ where $\beta$ equals 4 in height and width respectively.

Fig. 2 (a) Dynamic window size using the horizontal and vertical projection profiles. (b) Static window size selected based on the largest character size and fitted manually over all characters.

The window size is specified from the dynamic adaptive window placed over each character in the previous step. The HOG feature set is obtained from each window and is used for classification. Since the HOG features are extracted from the exact location of characters based on the dynamic bounding window, there will be a minimal redundant and noise features in the HOG vector. Fig. 2 (b) shows a fixed window size over characters; we argue that even if an exact placement of window is achieved, the window that covers one character may also cover other character or part of it. On the other hand, a dynamic window covers each character based on its size and also characters that are missing part of their stroke for instance character H, as shown in Fig. 2 (a), where the bottom part of the stroke is erased due to circulation effects. In addition, the static window has a number of overlapping windows and fits some characters while it over fits others on the same coin. The dynamic window size is very applicable to characters and works for different coins with different character sizes.

C. Classification

The Dynamic-HOG features are calculated based on the orientation histograms of edge intensity (bins) in a local region. Therefore, the feature vector size can be calculated by multiplying the number of bins $B$ in each cell $C$, by the number of cells $C$ in each block $Y$, and finally by the number of blocks $Y$ in each window $W$. The final HOG feature size can be calculated by formula (6) below:

$$\text{Size}_Y = Y_{\text{vertical}} \times Y_{\text{horizontal}} \times C \times B$$  \hspace{1cm} (6)$$  

Given that in Fig. 3 (a)(b) we used $8\times8$ pixels for each cell and $2\times2$ cells in each block, this gives us a $25$ blocks vertically and another $12$ blocks horizontally. The total number of HOG features in the feature vector is $10,800$.

The feature vector has a rich descriptor as shown in Fig. 3, where each block contains the radius and weight of each bin to form a small illustration of the 9 bins where each bin has an angle between 0 to 180 degrees and a normalized relative weight. However, the $10,800$ features is a very large number that contains redundant and irrelevant information due to the overlapping blocks or non-edge areas such that in the green rectangles in Fig. 3. Moreover, due to the different size of window for each character, the feature vector size would vary and thus we add zero values at the end of each vector to match the length of the longest feature vector. Therefore, the goal of using dimensionality reduction methods is to reduce the feature vector size, increase the learning accuracy, improve result comprehensibility, and avoid overfitting of data.

The principal component analysis (PCA) transforms the feature vector dimensional space into a subspace of the original dimension space. The PCA has the advantage of faster matrix multiplication due to the linear transformations which in return requires lower computational cost. It is a data dependent, which rely on the data only to find the maximum variance between them. The PCA has been widely used in computer vision applications [2] and combined with HOG to detect human, eyes, and cars in several research papers.

![Example on HOGs descriptor using 8*8 pixels in each cell and 2*2 block size from a $52$ Canadian coin’s character of size (280*104). (b) Example on HOGs descriptor using the same cell and block size settings from Chinese coin’s character of size (139*94)](image)

The reduced feature vectors obtained from the dimensionality reduction method is now the input to the Support Vector Machine (SVM) classifier. A linear kernel function is chosen for the proposed system over the radial basis kernel function to achieve a better accuracy, faster computation, and less overfitting. Moreover, given the $k(k-1)/2$ classifiers we obtain at the end of the training, each feature vector is assigned to a class based on majority votes. The positively classified characters are then segmented from the coin and used to identify the coin’s minting country.

IV. EXPERIMENTAL RESULTS

The goal of our experiments is to evaluate the coin recognition rate by segmenting and recognizing the characters based on Dynamic-HOG descriptor. Datasets play an important role in evaluating the system performance. Hence, we used different coins from Canada, Denmark, China, and US to evaluate the performance of our proposed solution. A total of 828 images belonging to 5 coins are used in two forms for evaluation: (1) the 828 images used as 9 sets each belongs to either the obverse or the reverse side on the coins except for the
Danish coin that has obverse side only. (2) the 828 images separated into 7 sets each with mixture of coins to evaluate the coin recognition rate. In the two forms of datasets separation, we used 70% of the images for training and 30% for testing in each dataset.

In the experimental work, sets of feature vectors are extracted from each dynamic window using the HOG descriptor and are used to recognize characters. The recall, precision, and f-measure values (shown in Fig. 4) for the 5 different coins each with the obverse and reverse side (except for the Danish coin having obverse side only) confirms the effectiveness of the Dynamic-HOG descriptor. The f-value shows a high accuracy rate in recognizing characters. The HOG settings for this experiment are 8*8 pixels in each cell and 2*2 cells in each block based on the experiment suggestions. The optimal window size is selected for each character as described in Section III.B. Characters belonging to the Danish coins returned the highest accuracy rate due to (a) clarity of the images with clear strokes representing each characters, and (b) the filled stroke of the characters where the stroke is a thick white line on black background. The characters of the Canadian coins returned the second highest accuracy due to the minimal noise around and between the character strokes. The lowest rate was reported for the US coins with a 1.87% drop in f-value compared to the Danish coins. In addition, the US coins have the smallest image size of 500*500 which makes it harder to determine the exact size of the bounding window where noise has higher effect on characters. In addition, we investigated the relationship between the parameter settings and the recognition rate.

![Fig. 4 Recall, precision, and f-measure values for the obverse and reverse side for each of the 5 coins in our dataset.](image)

The character segmentation step is performed to confirm the results of character recognition and for further processing such as studying the character characteristics (i.e. stroke width) in coin authentication. The character segmentation process is performed on correctly classified blobs using the Dynamic-HOG. The character segmentation step is evaluated using the segmentation error rate which complements the recall rate. The segmentation error rate results are shown in Fig. 5 where the x-axis represents the 5 different coins images each with 2-sides of the coin (except for the Danish coins) used in our experiments. The y-axis depicts the accuracy rate of the evaluation method. The highest segmentation error rate also belongs to the US coins dataset due to the fact that: (1) The US coins dataset contains more than one series of coins where the character shape has a slight change. (2) The US coins dataset has the smallest image size. Therefore, scratches and noise were present on the US coin images more than images of other datasets. (3) The US coins dataset has a small set of images, thus it is more vulnerable to the overall segmentation error rate. Generally, the average segmentation error rate is 0.026, where the highest is 0.0387 and lowest is 0.010. Since the character height and width ranges that we determine for window size selection would increase and it increases the false positive segmentation. On the other hand, the coin recognition rate is more vulnerable when there are very few characters on the coin; the more characters on the coin, the lower the segmentation error.

![Fig. 5 Segmentation Error Rate for the proposed character segmentation method.](image)

Furthermore, the coin recognition has been assessed using f-measure based on characters recognized by dynamic-HOG method. The aim is to measure the accuracy rate of recognizing coins using features from subparts of the image to reduce the computational cost. Each coin has its own text that refer to the minting country which is the part that used by human vision system to recognize the coin. Therefore, those characters can reliably reveal the coin origin after recognizing them. Fig. 6 depicts the normalized number of correctly recognized coins over the 7 datasets of mixed coin types to evaluate the performance. The results demonstrate the effectiveness of the proposed method and suggest that the proposed method outperforms the state-of-the-art methods. The overall performance of the proposed method from all coins is 99.5% which confirms the reliability of using subparts of the coin to recognize it rather than the whole coin image.

![Fig. 6 Normalized number of coin recognition rate from the 7 datasets of mixed coin types.](image)

Table I shows the performance of our method as well as these of other methods. The six directly related research works have shown promising results to the field. Four of these research works focused on ancient coins and two used modern coin datasets. While, four methods proposed character segmentation in which two of them specified the region of interest (ROI) of character locations manually, while [11] used HOG-like descriptors to locate and extract characters automatically but no results was reported for the character recognition. [12] proposed character segmentation only and no character recognition have been discussed. The first three methods in the table used the SIFT descriptors to represent features of characters. The reported results suggest a better performance for SIFT features when applied to the exact ROIs that contain characters as shown in [9] while a lower recognition rate resulted for approximate candidate locations of characters [8, 10].

[13] worked on identifying the numbers on the coin to recognize the minting year of the coin. The authors used a binary gradient map of the segmented characters to study the blob areas.
and its properties. The utilization of such an approach is not reliable for circulated coins as shown in their results of 44% on real coins and 92% for synthetic coins. The Dynamic-HOG descriptor used in the proposed solution returned the highest recognition accuracy of up to 98.15%. Bounding each character (w.r.t. its size) with a dynamic window and extract the HOG features from this dynamic window shows a significant improvement over methods in related work. It also reduced the number of false positive characters by specifying the range of acceptable height and width for characters.

### Table 1

<table>
<thead>
<tr>
<th>Research Paper</th>
<th>Feature extraction Method</th>
<th>Dataset (Coins)</th>
<th>Character Segmentation</th>
<th>Recognition Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>[8]</td>
<td>Word recognition using SIFT descriptor</td>
<td>Ancient Roman coins</td>
<td>NA</td>
<td>53%</td>
</tr>
<tr>
<td>[9]</td>
<td>Character recognition using SIFT descriptor</td>
<td>Ancient coins</td>
<td>Manually specified ROI</td>
<td>75.6%</td>
</tr>
<tr>
<td>[10]</td>
<td>Word recognition using SIFT descriptor and image matching</td>
<td>Ancient Roman coins</td>
<td>464</td>
<td>40% (word recognition only)</td>
</tr>
<tr>
<td>[13]</td>
<td>Binary gradient map and connected component properties</td>
<td>US coins</td>
<td>375</td>
<td>44% real</td>
</tr>
<tr>
<td>[12]</td>
<td>NA</td>
<td>Canadian, Danish, Chinese, US</td>
<td>348</td>
<td>Projection profiles</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>Dynamic-HOG descriptor</td>
<td>Canadian, Danish, Chinese, US</td>
<td>828</td>
<td>Dynamic-HOG</td>
</tr>
</tbody>
</table>

### CONCLUSIONS

In this paper, we studied the problem of structured object recognition with the focus on recognizing the characters minted on coins. We proposed a Dynamic-HOG descriptor for character recognition. The proposed solution was evaluated on various coin types including a publicly available dataset. The challenges of this work are the character orientation, character sizes, heavily degraded coin quality, and character clarity problems due to highlight and shadow variations caused by different lighting sources. The proposed method works on recognizing the characters after placing a bounding window over each character w.r.t. its size from the coin image to recognize the coin in return. The method was experimentally proven to be capable of handling different languages and extracting characters accurately. Experimental results suggest that in addition to the image quality of the coin having the highest impact on the recognition rate, the number of characters on the coin also has an impact on the recognition and segmentation accuracy. The Dynamic-HOG descriptor shows robust feature extraction for characters with much lower computational cost.

### ACKNOWLEDGMENT

This research was supported by a research grant from NSERC, Natural Science and Engineering Research Council of Canada.

### REFERENCES


Increase Efficiency of Simple Images Segmentation Using Detectors Based on Doubly Stochastic Random Fields

Nikita Andriyanov
Telecommunications Department
Ulyanovsk State Technical University
Ulyanovsk, Russia
nikita-and-nov@mail.ru

Vitaly Dement'ev
Telecommunications Department
Ulyanovsk State Technical University
Ulyanovsk, Russia
dve@ulnic.ru

Abstract— The problem of segmentation of simulated images containing simplest objects is considered in the article. In order to improve the accuracy of segmentation, it is proposed to use an algorithm for detecting anomalies, based on doubly stochastic random fields. The proposed algorithms are investigated when it is different signal-noise-ration.

Keywords— doubly stochastic models, image modeling, image processing, anomalies detection, segmentation, k-means, isodata

I. INTRODUCTION

Recently, the technology of machine vision has been developing rapidly [1-3], related, among other things, to the development of image processing techniques based on neural networks [4]. The use of such methods, first of all, is connected with the need to solve problems of understanding images and pattern recognition. Typically, neural networks are used to cluster images. At the same time, it is not possible to obtain a universal method for solving this problem. Indeed, this is due to the fact that in nature there are many real objects of completely different nature, and training of neural networks for each object is impossible. However, a number of real objects can be formally represented by simple geometric figures, or by their combinations. Therefore, image segmentation as an important step before pattern recognition can be performed based on the detection of such objects against a background of correlated interference with a complex structure [5-8]. This in theory will reduce the computational costs associated with training neural networks, and also obtain segmentation results that are not inferior to common algorithms such as k-means and isodata, in some cases with known signal levels.

II. A COMPLEX STRUCTURE IMAGE MODEL

It was noted in [5] that most real images possess the property of heterogeneity, therefore it was suggested to use doubly stochastic models. Consider the simplest case of a doubly stochastic model of a random field (RF) [5].

Let the simulation of the image take place in accordance with the three-stage simulation algorithm. Fig. 1 shows all of the three stages.

Fig. 1 The way to simulate image with complex structure

The first stage involves the generation of a component of a homogeneous RF $\varnothing$ (it is base RF). In the second stage, it is necessary to perform the transformation of the obtained RF $\varnothing$ so that all RF values are correlation parameters $\rho_j$, $j \in (j_1, ..., j_M)$, where $M$ is a parameter that describes the dimensions of the simulated image. These parameters characterize the correlation between the formed pixel of the image and its neighbors, similar to the usual autoregression (AR). Finally, using RF of parameters $\rho_j$ we generate the main image. Then write

The study was supported by RFBR, project 16-41-732027.
\[
\rho_{ui} = r_{1x}\rho_{x(i-1)j} + r_{2x}\rho_{x(j-i-1)} - r_{1x}r_{2x}\rho_{x(i-j-1)} + \xi_{ui}, \quad (1)
\]
\[
\rho_{uj} = r_{1y}\rho_{y(i-1)j} + r_{2y}\rho_{y(j-i-1)} - r_{1y}r_{2y}\rho_{y(i-j-1)} + \xi_{uj},
\]
where \(\{\xi_{ui}\}\) and \(\{\xi_{uj}\}\) are two-dimensional RF of independent Gaussian random variables (RV) with zero means and variances \(M\{\xi_{ui}^2\} = (1-r_{11}^2)(1-r_{22}^2)\sigma_{\rho_i}^2\) and 
\(M\{\xi_{uj}^2\} = (1-r_{11}^2)(1-r_{22}^2)\sigma_{\rho_j}^2\).

\[
\sigma_{\rho_i}^2 = M\{\rho_{ui}^2\} , \quad \sigma_{\rho_j}^2 = M\{\rho_{uj}^2\}.
\]

Then we substitute the varying correlation parameters (1) in the first-order AR model:

\[
x_{ij} = \rho_{ui}x_{i-1,j} + \rho_{uj}x_{i,j-1} - \rho_{ui}\rho_{uj}x_{i-1,j-1} + \sigma_i\sqrt{(1-\rho_{ui}^2)(1-\rho_{uj}^2)}\xi_{ij}, \quad (2)
\]
where \(x_{ij}\) is simulated RF with normal distribution \(M\{x_{ij}\} = 0, \quad M\{x_{ij}^2\} = \sigma_x^2\); \(\xi_{ij}\) is RF of independent standard Gaussian RV, \(M\{\xi_{ij}^2\} = 0, \quad M\{\xi_{ij}^2\} = \sigma_x^2 = 1\); \(\rho_{ui}\) and \(\rho_{uj}\) are correlation coefficients between neighboring elements of the RF \(x_{ij}\) along the axes \(x\) and \(y\) respectively.

Thus, the image generated by the model (2) will have a complex structure. At the same time, such images can be fully used as a background underlying surface for objects detected before segmentation.

**III. SIMULATION OF SIMPLE AND COMPLEX OBJECTS ON A DOUBLY STOCHASTIC BACKGROUND**

The obtained model (2) can be easily modified by adding a conditional dependence of the RF values on the location. For example, in the simplest case, to form an object of the square form on the image, we write the following expression

\[
x^*_ij = \begin{cases} 
  x_{ij}, & (i,j) \notin D_{sq}, \\
  x_{ij} + Sq_{ij}, & (i,j) \in D_{sq}, 
\end{cases} \quad (3)
\]

where \(Sq_{ij}\) is the value of a square signal at a point with coordinates \((i,j)\); \(D_{sq}\) is the given area on which it is necessary to simulate a square signal.

On the basis of (3), one can obtain records for other signals of a simple geometric form such as round, triangular, oval. You can also use combinations of different signals to form more complex objects

\[
x^*_ij = \begin{cases} 
  x_{ij}, & (i,j) \notin D_{sq}, \\
  x_{ij} + Sq_{ij} + Ci_{ij} + Tr_{ij}, & (i,j) \in D_{sq} \cup D_{ci} \cup D_{tr}, 
\end{cases} \quad (4)
\]

where \(Sq_{ij}, Ci_{ij}, Tr_{ij}\) are the values of the square, round and triangular signals at the corresponding points with coordinates \((i,j)\); \(D_{sq} \cup D_{ci} \cup D_{tr}\) indicates that the signal being generated is a single structure, and all adjacent areas.

This approach ensures greater proximity of the objects to the real. However, the most interesting is the case when there are many objects in the image. Then their qualitative segmentation is necessary, as a result of which it is possible to produce further recognition. We write the following form of the model

\[
x^*_ij = \begin{cases} 
  x_{ij}, & (i,j) \notin D_{sq}, \\
  x_{ij} + Sq_{ij}, & (i,j) \in D_{sq}, \\
  x_{ij} + Ci_{ij}, & (i,j) \in D_{ci}, \\
  \ldots \\
  x_{ij} + Tr_{ij}, & (i,j) \in D_{tr},
\end{cases}
\]

where \(Sq_{ij}, Ci_{ij}, Tr_{ij}\) are the values of the square, round and triangular signals at the corresponding points with coordinates \((i,j)\); \(D_{sq}, D_{ci}, D_{tr}\) correspond to the areas of each of the objects that do not intersect in the image.

Fig. 2 shows the images of objects obtained on the basis of expressions of the form (2) - (5). In this case, the signal in the images at each point assumes RV with a uniform distribution and with a given average value.
Thus, it is possible to easily create objects of different shapes on the image. This is important in the development of segmentation algorithms. In this case, the modeling of objects allows you to accurately know their presence in one or another place, which implies the possibility of numerical evaluation of segmentation results.

IV. DETECTION OF SIGNALS AGAINST A BACKGROUND OF DOUBLY STOCHASTIC INTERFERENCE

Compensation or weakening of the correlated background occurs by subtraction from the observed \( i \)-th reference of information field \( X(\hat{i}) \) prediction \( \hat{X}(\hat{i}) \), which is based on observations from the vicinity of the reference. To calculate the forecast \( \hat{X}(\hat{i}) \) we can use the linear prediction

\[
\hat{X}(\hat{i}) = \sum_{j=0}^{\infty} a(j) x(\hat{i} - j),
\]

where \( M \) is prediction area; \( a(j) \) are weighting coefficients. Choosing the best coefficients \( a(j) \) is carried out, proceeding from a condition of a minimum of a dispersion of a prediction error.

Then the solution of the signal detection problem is possible in the case of finding the likelihood ratio

\[
L = \frac{w(\{z_j\}/H_1)w(\{z_j\}/H_0)}{w(\{z_j\}/H_1)w(\{z_j\}/H_0)}
\]

and its comparison with the threshold value \([9]\). For example, the often mentioned probability density \( w(\{z_j\}/H_1) \) and \( w(\{z_j\}/H_0) \) can be approximated by Gaussian:

\[
w(\{z_j\}/H_0) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}(z_j - m_{0j})^T V_k^{-1} (z_j - m_{0j})\right\},
\]

where \( m_{0j} = M\{z_j/Z_{0}, H_1\}, s_j = s_j + \tilde{x}_{\theta j} \); \( m_{0j} = M\{z_j/Z_{0}, H_0\}, \tilde{x}_{\theta j} = M\{x_j/Z_{0}\} \) is optimal (in the sense of the minimum variance of the error) the forecast of the RF, made on the basis of all observations \( Z_0 \), in which the useful signal is known to be missing (forecast in the region \( G_0 \));

\[
V_k = M\{z_j - m_{0j}, z_j - m_{0j}\} = P_{\theta jk} + \sigma^2_{\theta jk} E_{jk},
\]

\( P_{\theta jk} = M\{x_j - \hat{x}_{\theta j}, x_j - \hat{x}_{\theta j}\} \) is covariance error matrix for optimal prediction; \( E_{jk} \) is unit matrix; \( s_j \) is the value of a useful signal at a point with coordinates \( j \).

Taking into account the above, the optimal signal detection rule is rewritten as

\[
L = \sum_{j=k} s_j V_k^{-1} (z_j - \tilde{x}_{\theta j}) \left\{ \begin{array}{ll} > L_0, & \text{signal} \\ \leq L_0, & \text{no\_signal} \end{array} \right\}.
\]

It is difficult to make a prediction \( \tilde{x}_{\theta jk} \) in the case of a large area \( G \). This fact is explained by the fact that at each point \( \tilde{k} \in G_0 \) to build the forecast, we use our own specialized algorithms. Strictly optimal forecast

\[
\tilde{x}_{\theta j} = \sum_{k} \alpha_{j k} z_j
\]

is a weighted summation of all (and there are quite a few) observations \( \{z_j\}, j \in G \), with coefficients \( \alpha_{j k} \), depending on the point \( \tilde{k} \in G_0 \).

However, the detection rule (8) can be replaced by a simpler rule, which is based on statistics:

\[
L = \sum_{j=k} s_j V_k^{-1} (z_j - \tilde{x}_{\theta j}) \left\{ \begin{array}{ll} > L_0, & \text{signal} \\ \leq L_0, & \text{no\_signal} \end{array} \right\},
\]

where \( \tilde{x}_{\theta j} \) is optimal prediction made on the basis of all observations, except (forecast to the point):

\[
V_{ij} = P_{\theta jk} + \sigma^2_{\theta jk} E_{ij}, \quad P_{\theta jk} \]

is covariance error matrix \( e_{ij} = x_j - \tilde{x}_{\theta j} \).

V. RESULTS OF JOINT DETECTION AND SEGMENTATION

The algorithms of modeling and detection of objects considered earlier will be checked in the course of statistical modeling. We will assume that the object has a constant addition to the background, i.e. \( S \) is constantly. We will also assume that we have a priori information about the possible location of the object. We perform detection and subsequent segmentation of the square signal at different signal-to-noise ratios. In this case, the dispersion of the background SP is considered noise here. If an object is found, its location data will be used for segmentation. The image size is 1000x1000, the size of the square is 70x70. The probability of false alarm is \( P_f = 0.001 \).

Table 1 shows the segmentation data. It should be noted that the segmentation characteristics with detection were obtained on the basis of 1000 experiments. The first line characterizes signal-noise ration, the second line characterizes detection-based segmentation, the third line characterizes isodata segmentation, the fourth line characterizes k-means segmentation.

Fig. 3 shows the efficiency of such segmentation. The solid line corresponds to the method with the detector, the dashed line corresponds to the k-means method, and the dot-dash line corresponds to the isodata segmentation.
<table>
<thead>
<tr>
<th>Signal-noise-ratio</th>
<th>0.1</th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>With pre-detection, %</td>
<td>16</td>
<td>52</td>
<td>91.7</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>isodata, %</td>
<td>0</td>
<td>36</td>
<td>68.8</td>
<td>93.2</td>
<td>100</td>
</tr>
<tr>
<td>k-means, %</td>
<td>0</td>
<td>39.3</td>
<td>77.1</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

**Fig. 3 Segmentation quality**

Analysis of the curves in Fig. 3 shows that segmentation based on detection is on average 15-20% better than known algorithms.

**VI. CONCLUSION**

In this paper, we propose algorithms for obtaining images with a complex structure, characterized by the presence on them of objects of various geometric shapes. In addition, the efficiency of segmentation of such images was investigated and compared with existing methods. It is established that when segmenting images containing one simple object, the use of detectors for a doubly stochastic model provides the 15-20% gains.

**ACKNOWLEDGMENT**

The study was supported by RFBR, project 16-41-732027.

**REFERENCES**


Missing data imputation based on stochastic neighbor embedding

Petrov I.B.  
Department of Computer Science  
Moscow Institute of Physics and Technology  
Moscow, Russia  
petrov@mipt.ru

Ryazanov V.V.  
Department of Computer Science  
Moscow Institute of Physics and Technology  
Moscow, Russia  
vasily.ryazanov@phystech.edu

Abstract—The problem of missing data imputation is considered. An approach of stochastic neighbor embedding (SNE), which is usually applied within tasks of dimensionality reduction, is adapted to the task of missing data preprocessing. This approach is based on probabilistic comparing of vectors for the same objects but in different feature spaces. Optimizing process is suggested. Derivations in optimization process are shown to be linear to dataset size. Perspective of given approach are confirmed by preliminary results.

Keywords—missing data, stochastic neighbors, optimization, data mining.

I. INTRODUCTION

The problem of missing data imputation is very important part of preprocessing in most of data analysis problems (classification, clusterization, regression, …). In this paper we consider only case of table datasets, which can be represented as matrix \( N \times M \), where \( N \) is the number of objects and \( M \) is the number of features. All known values are considered to be real values (not text, audio, enumeration, …). Some values are unknown and one can call them missing values, or \textit{nan} (from programming languages NaN is “Not a Number”).

There is a couple of reasons which can lead to increasing nans in table. Some of them are errors during measurements, anomalies, aggregations from different databases etc. Regardless of the underlying reason the existence of nans in data can lead to significant decrease of classification (or other) metrics. Some classifiers can even deny to process data with unknown values. In order to overcome these difficulties missing data is usually processed before classification.

The types processing data are usually applied – “marginalization” and “imputation”. In case of “marginalization” or “skipping incomplete objects” [1] incomplete objects are simply removed from the dataset. This approach is seriously limited to specific tasks.

“Imputation” is considered both as simple filling data with mean, median or other statistics and more complicated classifiers, regressors. Among more complicated approaches most popular are those based decision trees or on comparison of objects based on their distance similarity. In [2] Honghai and others investigated SVM-based approach in comparison with different means, medians and local embeddings. Their study shows that SVM performed better results. Also well-known are EM-algorithms which fill missing values according to maximum likelihood [1]. In [3] three approaches of imputation are suggested: one based on local distance optimization, second is optimization of Euclidian functional between all vector pairs. The third approach is a series of classification tasks among all possible values of the feature.

In this paper authors came with idea of applying stochastic neighbor embedding (SNE) to the task of data imputation. SNE-algorithms was introduced by G.E. Hinton and S.T. Roweis in [4] and was used to dimensionality reduction. This approach was modified to t-distributed SNE or t-SNE [5] which is the most popular algorithm nowadays to reduce dimensionality.

In [4] paper authors construct completely new feature space (usually 2- or 3- dimensional) which is followed by comparing probabilistic distances between different objects in old features space and in new feature space. This is followed by optimizing Kullback–Leibler divergence between 2 sets of probabilities. SNE or t-SNE are usually used in tasks of data visualization or embedding high dimensionalities.

The approach presented in current paper is an adaptation of SNE for dimensionality reduction to the task of missing data imputation. Analogically to [4] 2 feature spaces are considered. One feature space is the set of features with no missing data. The second feature space is equivalent to initial full feature space, but unknown values are filled with some values which are being optimized. Next, similarly, Kullback–Leibler divergence is considered as optimizing functional. In order to optimize this functional all partial derivatives are needed to be calculated. Though KL divergence has quadratic dependence to the number of objects in dataset, one can show that every derivative has linear dependence to \( N \).

Preliminary practical results show the perspective of given approach.
II. STOCHASTIC NEIGHBOR EMBEDDING

A. Stochastic neighbor embedding for dimensionality reduction

Let's a table dataset with missing data is given. Let's consider 2 types of object vectors: 1st type are objects limited to subspace of features where all features among dataset nave not-nan values. 2nd type are objects in the full space of features, unknown values are filled with temporary variables which are being optimized. The goal is to find “good” values for variables in order to minimize functional based on SNE.

As mentioned above two object vector types are given: 
\[ x_i, \tilde{x}_i, i = 1, N, x_j - \text{are vectors in subspace with only known values, and} \quad \tilde{x}_j \text{- are vectors in full values space.} \] 
For each feature space let’s transpose Euclidian Distance to conditional probabilities of having \( j \) object in the area of \( i \) object:

\[
P_{ji} = \frac{\exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}ight)}{\sum_{k \neq i} \exp\left(-\frac{\|x_i - x_k\|^2}{2\sigma^2}ight)}
\]

Similarly for \( \tilde{x}_i, \tilde{x}_j \):

\[
\tilde{P}_{ji} = \frac{\exp\left(-\frac{\|\tilde{x}_i - \tilde{x}_j\|^2}{2\tilde{\sigma}^2}\right)}{\sum_{k \neq i} \exp\left(-\frac{\|\tilde{x}_i - \tilde{x}_k\|^2}{2\tilde{\sigma}^2}\right)}
\]

Values \( \sigma_i, \tilde{\sigma}_i \) can be set by hand or found by the binary search that makes the entropy of the distribution over neighbors equal to \( k \). \( k \) here is “perplexy” or effective number of local neighbors and is chosen by hand. Entropy and perplexy are described by formulas:

\[
H(P_i) = -\sum_j p_{ji} \log_2 p_{ji}; \quad \text{Perp}(P_i) = 2^{H(P_i)}
\]

To simplify calculations one can set \( \forall i \to \sigma_i = \tilde{\sigma}_i = 1/\sqrt{2} \). Next we need to find and optimize some functional of similarity for these two probabilistic distributions. Similarly to SNE for dimensionality reduction one can use Kullback – Leibler divergence.

\[
C = \sum_i KL(P_i \parallel \tilde{P}_i) = \sum_i \sum_j p_{ji} \log \frac{p_{ji}}{\tilde{p}_{ji}}
\]

B. Optimizing process for Kullback – Leibler divergence

Next in order to start optimizing we need to calculate partial derivatives for all \( \tilde{x}_{i_1, i_2} \) where \( x_{i_1, i_2} = \text{nan} \). Let’s show that this calculation is linear to the number of objects and we don’t need much computation power to run it.

One can notice how to simplify derivative which is independent to \( p_{ji} \):

\[
\left( p_{ji} \log \frac{p_{ji}}{\tilde{p}_{ji}} \right)' = \frac{p_{ji}}{\tilde{p}_{ji}} - p_{ji} \frac{\log \tilde{p}_{ji}}{\tilde{p}_{ji}} + \frac{p_{ji}}{\tilde{p}_{ji}} \frac{\tilde{p}_{ji}'}{\tilde{p}_{ji}}
\]

\[
= -\sum_i p_{ji} \frac{\partial \log \tilde{p}_{ji}}{\partial \tilde{x}_{i_1, i_2}} = -\sum_i q_{ji}
\]

As calculating derivations is not easy to do the formula is broke up into 4 parts. Consider values \( q_{ji} \) for different \( i, j \) pairs.

1 case:

\( i \neq i_1, j \neq i_1 : q_{ji} = p_{ji} \frac{\partial \log \tilde{p}_{ji}}{\partial \tilde{x}_{i_1, i_2}} = 0 \)

\[
p_{ji} \frac{\partial}{\partial \tilde{x}_{i_1, i_2}} \left( -\|\tilde{x}_i - \tilde{x}_j\|^2 - \log \left( \sum_{k \neq i} e^{-\|\tilde{x}_i - \tilde{x}_k\|^2} \right) \right) =
\]

\[
= p_{ji} \left( 0 + \frac{e^{-\|\tilde{x}_i - \tilde{x}_j\|^2}}{\sum_{k \neq i} e^{-\|\tilde{x}_i - \tilde{x}_k\|^2} d_{i_1, i_2}^i} \right) = p_{ji} \tilde{p}_{ji} d_{i_1, i_2}^i
\]

2 case:
\[ i \neq i_1, j = i_1: q_{j|k} = p_{j|k} \frac{\partial \log \tilde{p}_{k|h}}{\partial x_{i_1,i_2}} = \]
\[ p_{j|k} \left( -\frac{1}{\|	ilde{x}_j - \tilde{x}_i\|^2} - \log \left( \sum_{k \neq i} e^{-\|\tilde{x}_j - \tilde{x}_i\|^2} \right) \right) = p_{j|k} d_{i_2}^j \left( \tilde{p}_{j|k} - 1 \right) \]

3 case:
\[ i = i_1, j \neq i_1: q_{j|k} = p_{j|k} \frac{\partial \log \tilde{p}_{j|h}}{\partial x_{i_1,i_2}} = \]
\[ p_{j|k} \left( -\frac{1}{\|	ilde{x}_j - \tilde{x}_i\|^2} - \log \left( \sum_{k \neq i} e^{-\|\tilde{x}_j - \tilde{x}_i\|^2} \right) \right) = p_{j|k} \left( \sum_{k \neq i} \left( \tilde{p}_{j|k} d_{i_2}^k \right) - d_{i_2}^j \right) \]

4 case:
\[ i = i_1, j = i_1: q_{j|k} = p_{j|k} \frac{\partial \log \tilde{p}_{j|h}}{\partial x_{i_1,i_2}} = \]
\[ p_{j|k} \left( -\frac{1}{\|	ilde{x}_j - \tilde{x}_i\|^2} - \log \left( \sum_{k \neq i} e^{-\|\tilde{x}_j - \tilde{x}_i\|^2} \right) \right) = p_{j|k} \left( \sum_{k \neq i} \left( \tilde{p}_{j|k} d_{i_2}^k \right) \right) \]

Sum all parts altogether.

\[ \frac{\partial C}{\partial x_{i_1,i_2}} = -\sum_{i \neq i_1, j \neq i_1} q_{j|k} \sum_{i \neq i_1} q_{h|i} - \sum_{j \neq i_1} q_{j|k} + q_{i|h} = \]
\[ = -\sum_{j \neq i_1} p_{j|k} \tilde{p}_{j|k} d_{i_2}^j - \sum_{i \neq i_1} p_{j|k} d_{i_2}^j \left( \tilde{p}_{j|k} - 1 \right) \]

one can simplify this derivative in order to find it with linear time. Also we use the fact that sum of all probabilities are equal to 1:
\[ \forall i \rightarrow \sum_{j \neq i} p_{j|k} = \sum_{j \neq i} \tilde{p}_{j|k} = 1 \]

Now one can simply see for 1st component:
\[ \sum_{i \neq i_1} p_{j|k} \tilde{p}_{i|k} d_{i_2}^j = \sum_{i \neq i_1} \tilde{p}_{i|k} d_{i_2}^j \sum_{j \neq i_1} p_{j|k} = \]
\[ = \sum_{i \neq i_1} \tilde{p}_{i|k} d_{i_2}^j \left( 1 + p_{j|k} - p_{j|k} \right) \]

As for double sum in 3rd component \( \sum_{k \neq i} \left( \tilde{p}_{k|j} d_{i_2}^k \right) \) can be calculated only once and then used in every sum. Thus one can see that partial derivative is linear (not quadratic) to the number of objects, though this equation can be further simplified. Also all values \( p_{j|k} \) do not change throughout time and be calculated only once. Then:
\[ \frac{\partial C}{\partial x_{i_1,i_2}} = -\sum_{k \neq i_1} \tilde{p}_{i|k} d_{i_2}^k \left( 1 + p_{j|k} \right) + \sum_{i \neq i_1} p_{j|k} d_{i_2}^j = \]
\[ = \sum_{i \neq i_1} d_{i_2}^j \left( \left( p_{j|k} + p_{j|k} \right) - \tilde{p}_{j|k} \left( 1 + p_{j|k} \right) - \tilde{p}_{j|k} \left( 1 + p_{j|k} \right) \right) \]

After we find all partial derivatives \( \frac{\partial C}{\partial y_i} \) it means that we found gradient for our functional and can run gradient descend algorithm, for example using momentum:
\[ y^{(t)} = y^{(t-1)} + \eta \frac{\partial C}{\partial y_i} + \alpha(t) \left( y^{(t-1)} - y^{(t-2)} \right) \]

where \( \eta \) is learning rate (or step size) and \( \alpha(t) \) is inertia coefficient. Some stopping criterion can be chosen. First steps \( y^{(0)}, y^{(1)} \) can be set to mean or median of the corresponding features.
C. Practical experiments

Experiments are run in the following way. At first some table dataset with full descriptions is chosen. Full description means that there is no nan in data. Next some part of values are set as nan. After this optimizing process described in the paper is run. So there are two vectors $y$ of real(initial) values and $\hat{y}$ of predicted values.

Next $\text{MAE}(y, \hat{y}), \text{MRSE}(y, \hat{y})$ are checked for different algorithms of predicting $\hat{y}$. Also one can check which impact gives this procedure to the classification accuracy(or other metric). Preliminary results show the perspective of current approach.

CONCLUSION

During every data analysis task a very important part of data preprocessing is missing data imputation. In this paper an approach based on stochastic neighbor embedding is proposed. Missing data is considered as optimizing variables and procedure of minimizing Kullback–Leibler divergence is run.

It is shown that every partial derivative is linear to the number of objects in dataset and can be calculated rapidly. Test practical experiments have been performed.

REFERENCES

Shape of Basic Clusters: Finding Coherent ELR-2s via Hough-type Transform

Elena Nelyubina  
*department of Water Resources and Water Use,*  
Kaliningrad State Technical University  
Kaliningrad, Russia  
e.nelyubina@gmail.com

Vladimir Ryazanov  
*Dorodnicyn Computing Centre,*  
Federal Research Centre ‘Computer Science and Control’ of Russian Academy of Sciences  
Moscow, Russia  
rvyccas@mail.ru

Alexander Vinogradov  
*Dorodnicyn Computing Centre,*  
Federal Research Centre ‘Computer Science and Control’ of Russian Academy of Sciences  
Moscow, Russia  
vngccas@mail.ru

Abstract—A new method for improving a wide class of linear decision rules is proposed on the basis of using the concept of generalized precedents and analogues of the Hough transform in higher dimensions.

Keywords—logical regularity, generalized precedent, convex hull, linear manifold, Hough transform

I. INTRODUCTION

As known, linear decision rules are of the fastest, this is especially important in cases with high data dimensions and volumes [3], [4], [5]. In problems where up to several hundred features are used, the calculation of nonlinear quantities, including the distances of particular kinds, becomes a separate problem. At the same time, intuitive grounds for using this or that type of distances often lose their obviousness. For instance, in dimensions 2 and 3, the approximation of the empirical distribution by a set of uniformly filled circles or balls looks natural. But the 256-dimensional hyper-ball, for example, contains 95% of its volume in the border layer with thickness of only 0.02 radii. In these conditions, the search for the "nearest neighbor" can be related to unjustified computational costs.

These, as well as many other similar considerations were taken into account when developing an approach to recognition problems based on the use of logical regularities [1] [2].

Elementary logical regularity (ELR) is a conjunction of predicates corresponding to linear threshold constraints. In the feature space $\mathbb{R}^n$, single ELR corresponds to the inner volume of convex linear hull. Initially, ELRs of the first kind (ELR-1) were investigated, in which predicates of constraints along the principal axes of the feature space are used

$$L_1 = \mathcal{E}_n R_{ni}, \text{ where } R_{ni}=(a_{ni}<x_{ni}<b_{ni}).$$

The convex hull in this case is hyper-parallellepiped, and the relation of belonging to the class is realized through belonging to the covering hyper-parallellepipeds. Later, ELRs of the second kind (ELR-2) were also investigated, where the predicates correspond to linear constraints of the general form

$$L_2 = \mathcal{E}_j R_{ji}, \text{ where } R_{ji}=(n_{jx_i}x_i<Thr_{ji}),$$

where $n_j$ is the normal vector to the $j$-th facet of the $i$-th convex hull, $Thr_{ji}$ is the boundary threshold.

II. GENERALIZED PRECEDENTS AND HOUGH-TYPE TRANSFORM IN HIGHER DIMENSIONS

Subsets of a certain type that are rather densely filled with objects of the training sample we consider as independent objects of higher level - generalized precedents (OPs). OPs are implementations of some typical local dependencies in data.

Algorithms using ELR-1 in a number of cases show record results for the criteria of speed and accuracy, since they require performing only comparisons of the object's coordinates with the boundaries on the axes. Even faster methods work, where the hyper-cubes of positional representation are used in the role of ELR-1, because here it is sufficient to perform only consecutive bit comparisons on the binary grid [6].

ELR-2 systems implement a more flexible tool that more economically describes both the filling of classes and the boundaries between them. At the same time, the fact that the new vector hits linear boundaries requires preliminary calculation of the convolutions with the normal vectors, which is more laborious than simple comparisons with thresholds on the principal axes. If two different ELR-2s have the same boundary normal vector, they are called coherent in the corresponding direction [12].

Our goal is to build an ELR-2 system with the maximum proportion of normal vectors participating in the coherence relations. When maximizing this proportion, it is important to which class the objects of one or another ELR-2 belong, since the a priori class probabilities are taken into account in error estimates.

In this paper, we present an approach to minimizing the total number of convolutions, based on the unification of close directions of the boundary normals. For these purposes, the concept of generalized precedents is used [7], [8]. The last is an analogue of the Hough transform in higher dimensions [9], and transition to the parametric space of coefficients of normal vectors is carried out.

Analysis of secondary distribution on the parametric space and further choice of the minimum number of representatives for the main density peaks, are performed taking into account the error levels introduced with the use of these unified representatives. As a result, it is possible to reduce the average expected number of convolutions that need to be performed when recognizing a new object.
that manifest themselves as condensations of empirical density and possess a characteristic spatial shape.

This geometric aspect of the dependency can be known a priori, or it may be revealed as a result of automatic selection of significant clusters with respect to one or another numerical criterion. Clusters of this kind are called basic clusters [7], [8]. Frequently occurring examples are:

- lineaments and other almost 1-dimensional subsets describing the consequences of data blurring or high correlation of features;
- Gaussian kernels, which represent predictable variations in the feature values under noisy registration conditions;
- hyper-cubes, hyper-spheres, ELR-1 and ELR-2 hulls, and the like.

It is assumed that the shape of basic cluster has simple parametric description. Suppose that for a certain class \( X_\lambda \), \( \lambda = 1,2,\ldots,l \), a covering of this class by basic clusters is found in the training sample. The covering corresponds to disjunction of parametric conditions (in our case, disjunction of elementary logical regularities)

\[
R = V \bigcup_{j} R_{ji}.
\]

The simplicity of describing basic clusters has many positive aspects. The basic cluster is not only an efficient component of the constructed decision rule, but it can also contain new important information, if it is associated with the geometric shape of manifestation of the regularity and was not previously known to specialists in the applied field. Moreover, this information is already automatically represented in a form that is easy to understand by a person.

Further, convex hulls of logical regularities of the second kind are considered as the basic clusters, which are uniquely represented by parameters of their piecewise linear boundary. In particular, substantial interpretation and understanding different regularities \( R = V \bigcup_{j} R_{ji} \) as sets of simple threshold restrictions does not cause in this case any difficulties.

Common in using all types of basic clusters is a transition to corresponding parametric space in which both the typical nature of the local regularity and the most frequently encountered values of their parameters can be revealed during the analysis of the secondary cluster structure. In particular, one can investigate some of low-dimensional parametric subspaces separately by applying appropriate methods, and after that aggregate and combine results.

All of the above contains obvious correlations with the main steps and methodology of applying the conventional Hough transform in IP tasks. But there are also serious differences.

1. The main difference is that the parameterized object in the case under consideration may correspond to a cluster in abstract feature space of arbitrary dimension.

2. It is equally important that in the role of the primary transformation (spatial differentiation) there can be recognized a variety of procedures for identifying significant clusters, when the shape of these clusters is predetermined and/or may vary in some controlled limits. In particular, many well-studied methods for approximating the empirical distribution by a mixture of elements of a certain type are suitable for these purposes, just as in the standard case of approximation of a sample by normal mixture

\[
\sum C_i = \sum \mu_i \exp(-0.5(x-x_j)^T \sigma^2 (x-x_j))
\]

with some or other restrictions on the form of covariance matrix \( \sigma \).

3. There is one more significant difference from the classical scheme of Hough transform: the construction of the best approximation of the sample by a set of basic clusters is essentially non-local process, the result of which depends on the geometry of the sample as a whole.

Let’s analyze the last in more detail. At primary glance, non-locality can devalue all the constructions presented above. But it is also obvious that the presence of local relationships of parameters is not an exceptional or even a rare event.

In fact, some features of data can be known a priori, and this directly affects the choice of the shape of basic clusters. For example, in IP, when working with images damaged by linear smear, lineament is usually chosen as basic cluster because it is only necessary to find a region with an increased density of realizations in the parametric space (as we use to do in conventional Hough transform), and then the center of the region will reveal the main direction and other parameters of the smear.

Let’s consider another practical example of the use of a priori information when choosing a system of basic clusters. As additional information for each class \( K_\lambda, \lambda = 1,2,\ldots,l \), sets of LR-1’s of classes \( P_\lambda = \{ P_\lambda(x) \} \) are used [7], [10], i.e., the predicates of the form

\[
P^{\Omega_1, \Omega_2, \epsilon_1, \epsilon_2}(x) = \bigcup_{j \in \Omega_1} \bigcup_{j \in \Omega_2} (c_j \leq x_j) \bigcup (x_j \leq c_j),
\]

where

\[
\Omega_1, \Omega_2 \subseteq \{1,2,\ldots,n\}, \epsilon_1, \epsilon_2 \in R^n,
\]

\[
\exists x_1, x_2 \in \tilde{K}_\lambda : P^{\Omega_1, \epsilon_1, \Omega_2, \epsilon_2}(x_1) = 1,
\]

\[
\forall x_1, x_2 \in \tilde{K}_\lambda : P^{\Omega_1, \epsilon_1, \Omega_2, \epsilon_2}(x_1) = 0,
\]

and \( P^{\Omega_1, \epsilon_1, \Omega_2, \epsilon_2}(x) \) is locally optimal criterion for standard predicate quality.

Training sample objects of the class \( K_\lambda \) are referred \( \tilde{K}_\lambda \). Sets of objects that satisfy predicates \( P_\lambda \) are put in correspondence to the set of real precedents \( \tilde{K}_\lambda \).

We used two variants of the "nearest neighbor" algorithm in computational experiments. Lets \( \rho \) is standard Euclidean metrics in \( R^n \). We considered that new object \( x \) belongs to the class, LR-1 of which was found the nearest.
The distance from \( \mathbf{x} \) to this LR-1 was calculated by the following formula, where geometric shape of basic clusters was exploited

\[
d_a(\mathbf{x}) = \sum_{x_i: P_{\Omega}^{\alpha} e^{\Omega_\lambda} \cdot e^{\Omega_\lambda} (\mathbf{x}_i) = 1} \rho(\mathbf{x}, \mathbf{x}_i)
\]

Thus, we used a priori belief that all revealed ELR-1 fit objective dependencies quite adequately.

In experiments a comparison was performed on the data of credit scoring: 2 classes, 15 features, 348 test objects [11]. The accuracy of standard and modified method of "nearest neighbor" on the test data was, respectively, 75.6% and 77.5% of correct answers.

As we see, knowledge of local specialties of the empirical distribution and about probable shape of basic clusters proves to be very useful. But it is always preferable to use some other, less subjective selection technique that does not require direct references to a priori information.

Let \( B_s, s=1,2,...,S \), is a set of cluster descriptions that could claim to be the basic. Each object \( B_s \) contains parameters of cluster shape that may be relevant to the task of detecting differences between classes \( \lambda = 1,2,...,l \). Let \( Q_z, z=1,2,...,Z \), is a set of quality criteria that are applied in approximation task for representation of class \( K_\lambda, \lambda=1,2,...,l \), using basic clusters of certain shape \( B_s \). Thus, we keep in denotation just two variables we need for setting the criterion \( Q_z, z=1,2,...,Z \), with which we establish \( S\times Z \)-matrix of votes for selection this or that shape of cluster as basic. Applying the shape set \( B_s, s=1,2,...,S \), and the list of criteria \( Q_z, z=1,2,...,Z \), to \( \lambda \)-th class \( X_\lambda \subset X \) of the training sample, we obtain a set of matrices \( q_{\lambda z}(\lambda), \lambda=1,2,...,l \), containing votes for basic shape \( B_s \) for class \( \lambda \) with respect to \( z \)-th criterion \( Q_z \).

The set \( q_{\lambda z}(\lambda) \), \( \lambda=1,2,...,l \), may serve further as an objective basis for selection certain shapes of clusters as basic. Of course, such choice can be made on the base of different strategies. Notice, that in all ways we’ve used minimum of a priori or subjective knowledge here.

We present below the calculation scheme of Hough-type transform in higher dimensions, where set of ERL-2 is used as set of basic clusters. Goal is to find most typical orientations of border hyper-planes represented in parametric space:

a) we construct a set \( L = \{L\} \) by finding all ERL-2s that form some covering of a class \( K_\lambda, \lambda=1,2,...,l \);

b) it is chosen a limited number of parameters characterizing border hyper-plane of ERL-2 and their position relatively to the main axes of the feature space. In particular, we consider parametric space \( C \) which provides representation of guide angles \( \alpha_n \) of the normal vector \( n \) to some border hyper-plane;

c) one-to-one mapping \( \theta: L \rightarrow C \) of the set \( L \) into selected parametric space \( C \) is constructed, and there some secondary clustering is performed;

d) while clustering we search for the set \( C^\dagger \) of expressed compact clusters \( c^\dagger, t \in T \), in the space \( C \). Each cluster \( c^\dagger \in C^\dagger \) represents some typical direction of normal vectors to border planes of different ERL-2 revealed at the first step;

Having the set \( C^\dagger \) we can deform representations of ERL-2s collected in the set \( L=\{L\} \) with aim to arrange more pairs of coherent regularities among them.

### III. COHERENT SUBSETS OF ERL-2

The most direct way is to choose a single representative \( c^\dagger \in C^\dagger \) for each cluster \( c^\dagger \in C^\dagger \), and then replace all normal vectors represented in \( c^\dagger \) with this representative. The representative can be chosen as average of parameters \( c^\dagger = \frac{1}{|c^\dagger|} \sum_{r \in C^\dagger} r \), or the median \( c^\dagger = \text{med}(c^\dagger) \), etc.

As the result, all modified ERL-2 of the set \( c^\dagger \in C^\dagger \) become coherent on unified axes, represented by different \( c^\dagger \). In particular, when \( c^\dagger = n^\dagger \), it will be enough to calculate direct product \( (x, n^\dagger) \) just once for each new object \( x \), instead of calculating all \( |c^\dagger| \) products for the whole cluster \( c^\dagger \).

---

**Fig.1.** Modeled 4-component ERL-2 covering \( \{L\} \) of class \( K_\lambda \).

In the same time, covering \( \{L\} \) of the class \( K_\lambda, \lambda=1,2,...,l \), will be harmed by such replacement, and the best decision on changes of this kind will be subject to the quality restrictions. The main reason for the choice of representatives \( c^\dagger \) should be based on the solution of appropriate optimization task, where both calculation efficiency and the accuracy of the improved covering \( \{L\} \) should be taken into account.

**Fig.2.** One-dimensional section \( C^\dagger \) of parametric space \( C \) for the covering \( \{L\} \). Black dashes show angles between facet normals and the horizontal axis by mod(180°). Red dashes are unified representatives of the main clusters.
In what further, we describe two concretizations of the presented scheme. In the first case we deal with normalized orthogonal vectors (normals) themselves: $c^t = \mathbf{n}^t, t \in T$. In the second case, we investigate coefficients of polynomials that describe border hyper-plane as whole. The second case provides an illustration, too, of double superposition of Hough-type transforms, with two kinds of basic clusters.

Let $f=1,2,...,F^k$ be the index of facets in ELR-2 hulls of $\{L\}_f$, and $\mathbf{n}_f$ is the normal to the $f$-th facet. We consider clustering in parametric subspace $\mathbf{C'} \subset \mathbf{C}$ of coordinates of $\mathbf{n}_f$. Other parameters of ELRs represented in $\mathbf{C}$ are ignored while clustering in $\mathbf{C'}$, and all facets of all ELRs are considered simultaneously. Moreover, to improve the decision rule as whole, we have to mix ELR-2s of the coverings of all classes $K_{\lambda}, \lambda = 1,2,...,I$, of the training sample and further look for coherent ELR-2 subsets that may unite different classes.

Since $||\mathbf{n}_f||=1$, the dimension for $\mathbf{C'}$ is chosen $N-1$. In Fig. 1 a model example of ELR-2 covering $\{L\}_f$ constructed for a class in two dimensions $N=2$ is presented. Corresponding one-dimensional-parametric subspace is shown in Fig.2.

Fig.3. Covering $\{L\}_f$ improved with unified representatives. Calculating membership in the class $K_{\lambda}$ requires 3 convolutions instead of 16 ones.

Reverse assembly of new covering of the class $K_{\lambda}$ is done on the base of improved ELR-2s. Inversion of the mapping $\theta^{-1} : \mathbf{C} \rightarrow \{L\}_f$

reestablishes places of improved ELR-2s in the structure of renewed decision rule. Elements of the new covering will have now restricted variety of normals to facets. Fig.3 shows a new covering improved with unified representatives of main clusters.

Notice that normalizing procedure is not an obligation. Here we show another way to construct the same improved covering for the class $K_{\lambda}$.

Each border hyper-plane corresponding to facet is a linear manifold of co-dimension 1. It means that the ideal $I_f$ in the ring $R(x_1,x_2,...,x_N)$ of polynomials on variables $x_n, n=1,2,...,N,$ that defines the manifold containing $f$-th facet, is principal ideal produced by a single polynomial of the first order $P_f = (\mathbf{x}, \mathbf{a}_f)-\mathbf{b}_f$.

Let $I_f = (\mathbf{P}_f)$, where $\mathbf{a}_f$ is a vector orthogonal to $f$-th facet. If we just look for variety of vectors $\mathbf{a}_f$ orthogonal to different facets of all ELR-2s revealed in the training sample and selected at the first step a), we can use directly the parametric subspace of coefficients without normalizing each vector $\mathbf{a}_f$.

Fig.4 shows 2-dimensional parametric subspace of such kind for the ELR-2 covering $\{L\}_f$ presented in Fig.1. As known, each polynomial $(\mathbf{x}, \mathbf{a}_f)-\mathbf{b}_f$ being multiplied by any real value still belongs to the ideal $I_f$ of corresponding hyper-plane. Thus we obtain a priori knowledge, that in this case any hyper-plane that's parallel with $f$-th facet is represented by a point of one and the same line crossing zero in parametric subspace.

Fig.4. Parametric subspace of coefficients $\mathbf{a}_f$ can be used again in the same scheme in the role of new feature space; centered lineaments (blue) or narrow cones (green) are preferable as basic clusters.

So, the whole scheme can be applied once again to the parametric subspace itself. To find coefficients of unified polynomial-representative one should use centered lineaments or narrow cones as basic clusters, since close orientations of facets of different ELR-2s produce almost coinciding lines in the space of coefficients. Triple of dotted red lines in Fig.4 corresponds to the same choice of representatives improving $\{L\}_f$ as in the case of parametric subspace for angles.

IV. CONCLUSION

In this paper we have considered promising relationships among three important topics:

- the computational efficiency of linear decision rules;
- the concept of generalized precedents, which allows the use of local dependencies and regularities in data;
- the construction and use of Hough-type transforms in higher dimensions.

A special problem arising in the optimization of linear solutions is investigated. It is shown how for this purpose the process of automatic finding generalized precedents as typical local regularities in data representation can be used. In general, the concept of generalized precedents seems very prospective and unites a number of approaches and methods that are close
in structure and sense. Usage of such instruments allows one to explicitly indicate the intrinsic local dependencies of object parameters, including those of them that were not known to specialists and were not used before in the considered applied area.

On this basis, a new approach to the construction of Hough-type transforms for higher dimensions has been proposed and is currently being developed, where different procedures for approximating the empirical distribution by clusters of one or another specified shape appear in the role of spatial differentiation. A computational scheme is presented that is suitable for further optimization of fast linear decision rules from a wide class of them. Examples of the use of proposed approach are given, in particular, an example of double Hough-type transform in higher dimensions.

ACKNOWLEDGMENT
This work was done with support of Russian Foundation for Basic Research.

REFERENCES


Abstract—The work is devoted to the recognition / identification of the objects of predefined shape on the images of a special class. This class of images is associated with registration of weak radiations and, accordingly, characterized by a relatively low rate of counts (low count rate images). The latter leads to a low signal-to-noise ratio, low contrast and fuzzy shapes of the objects. For this reason, the known methods, designed for traditional image recognition, are not effective enough in this case and new recognition approaches, oriented to a low count rate images, are required. In this paper we propose such an approach. It is based on the machine learning paradigm and designed for identifying (low count rate) objects given by a point–set. So, in the frames of the approach the recognition problem is posed as the statistical inference via already observed data (precedents), rather than the traditional in statistics problem of hypothesis testing. For the implementation of this recognition approach, we had to reduce the problem of object recognition to the machine learning problem of maximizing the tested point–set likelihood with respect to the classes of modelling object shapes up to shape size and position. The resulting recognition algorithms have a structure close to the well-known EM algorithm in the Variational Bayesian approach interpretation.

Index Terms—point process intensity shape identification, formal shape description, inhomogeneous Poisson point process, finite mixture models, machine learning, effective computational schemes, EM, VB EM algorithms

I. INTRODUCTION

The main problem considered in the paper is connected with the statistical inference about the intensity of an inhomogeneous spatial Poisson point process (PPP), whose realization is given as a set of discrete points (events) \( \{ \vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n \} \). It is well-known [1], that such processes are a good model of the images related to the registration of weak radiation and, accordingly, characterized by a relatively low rate of counts (low count rate images). Such images are common in astronomy (optical radiation), computer and positron–emission tomography (X-ray radiation), thermography (infrared radiation), terahertz applications.

There are many methods and approaches to the PPPs analysis. Among the first works devoted to the analysis and evaluation of PPP characteristics, in particular its intensity, we should mention the papers of Cox [2], [3]. These papers are characterized by the systematic application of classical statistical methods to the class of stochastic point processes. Many problems of evaluating the PPP intensity and comparing the intensities of several PPP have been considered and solved. But it should be noted, that most problems were solved only for the case of homogeneous Poisson processes. The inhomogeneous case is analytically greatly complicated, and therefore the problems of estimating the temporal dynamics of intensity were not practically considered (except for the problem of trend estimation).

With the development of computer technologies, statistical methods have become more and more algorithmic in nature and more in line with the principles of machine learning approaches [4], [5]. Several complex problems have been solved by successfully combining processing power of modern computers and effective computational schemes of some algorithms (usually of recurrent type) developed over the last 20 to 30 years [6]. For example, several problems of estimating the Poisson process intensity, that widely use methods of machine learning are set forth in the work [7]. The author considers parametric intensities and discusses approaches to estimating their parameters depending on the type of the problem - either it is related to the estimation of the intensity of a single process or to the mixture (superposition) of the processes. It is noted that in the case of mixtures the EM (Expectation–Maximization) algorithm for Gaussian mixtures is most effective. Several different approaches concerning the estimation of random intensities are presented in [8]. The author of the proceedings considers Poisson processes with random intensities as twice stochastic, or “Cox processes”, and proposes his own method for estimating the intensity based on the Monte Carlo methods for Markov processes.

In this paper, we also propose a new method for identifying the intensity shape of inhomogeneous Poisson process based on a certain variant of machine learning approach. In accordance with the principles of statistical (machine) learning, the problem considered is posed as the statistical inference via already observed data, precedents, rather than the traditional...
II. Model

To proceed to the solution of the problem posed in the work in the spirit of the machine learning, it is necessary to clarify some initial assumptions. Let us consider two main groups of them. The first group of assumptions concerns the Poisson point process (PPP) characteristics accepted as its description for some database of precedents. The second group is connected with the type of correspondence between the identified, registered process and the corresponding target precedent from the database.

Let us assume that each (k-th) precedent is the realization of spatial PPP with intensity \( \lambda_k(\vec{x}) \), defined in some plane region \( \vec{x} \in S \). We also assume that this intensity could be approximated by a superposition (mixture) of \( N_k \) Gaussian components (Gaussian mixture):

\[
\lambda_k(\vec{x}) = \Lambda_k \sum_{j=1}^{N_k} p_{k,j} \frac{\sqrt{\det(A_{k,j})}}{2\pi} \exp \left\{ -\frac{1}{2} Q_{k,j}(\vec{x}) \right\},
\]

(1)

where \( \Lambda_k \) is the integral over \( S \) of \( \lambda_k(\vec{x}) \) - total PPP intensity power, \( p_{k,j} \) is the fraction of power for \( j \)-th component, mean vector \( \vec{m}_{k,j} \in S \) and reverse covariance matrix \( A_{k,j} \) are the parameters of this component. Let us note, that \( \Lambda_k \) is also the parameter for Poisson distribution of \( n \) - total number of process points.

The power \( \Lambda_k \) and the set \{ \((p_{k,j}, \vec{m}_{k,j}, A_{k,j})\), \( j = 1, \ldots, N_k \) \} of the above listed parameters exactly describe intensity (1) of PPP. However, for comparing the intensities these parameters are not equally important. For example, the parameter \( \Lambda_k \), determining the overall "brightness" of the process does not say anything about intensity distribution, so it is not important when comparing the intensities in shape. Therefore, let us exclude \( \Lambda_k \) from the set of parameters describing the form of the mixture (1). As a result, the parameters describing each precedent will be the following: \( N_k \) – the number of mixture (1) components and the set of triples \{ \((p_{k,j}, \vec{m}_{k,j}, A_{k,j})\) \} in the number of \( N_k \) entities – parameters of all components of the mixture. Thus, setting \( \Lambda_k \) in (1) to unity, we come to a description of the precedents by parameters of Gaussian mixture which is the very popular probability distribution in statistics.

Regarding the introduced precedent feature (parameter) space, it is important to emphasize the following fact. For a fixed number \( n \) of Poisson process points, their coordinates \( \{\vec{x}_i\} \) are the independent random variables identically distributed (iid) with the probability density \( \lambda_k(\vec{x})/\Lambda_k \) [7]. In this connection, the choice of the Gaussian mixture parameters space as a precedents feature space is important not only from the formal, discussed above point of view, but also from pragmatic position for implementing effective computational identification procedures. Indeed, within the framework of machine learning there are many effective algorithms that allow to find the maximum likelihood estimates of Gaussian component parameters for the sets of iid random points. This group of algorithms includes, for example, the popular EM-like algorithms [6] that recursively refine parameters \{ \((p_{k,j}, \vec{m}_{k,j}, A_{k,j})\) \} for Gaussian mixtures like (1).

Having determined the structure of the database of precedent descriptions and the way of its formation by EM-type algorithms, it is necessary to clarify the procedure for identifying any given test set of points whether it is an implementation of some description of precedent from the database. Theoretically, it would be possible to assume that there are precedents in the database of all possible shapes, so the description of the tested point-set will coincide with the normalized version (1) of one of them. Thus, if we form for \( n \) discrete samples \( \{\vec{x}_i\} \) of the tested set the (logarithmic) likelihood function \( L_k(\{\vec{x}_i\}) \) based on descriptions of all (k-th) precedents:

\[
L_k(\{\vec{x}_i\}) = \ln \prod_{i=1}^{n} p(\vec{x}_i | k) = \ln \prod_{i=1}^{n} \sum_{j=1}^{N_k} p_{k,j} \frac{\sqrt{\det(A_{k,j})}}{2\pi} \exp \left\{ -\frac{1}{2} Q_{k,j}(\vec{x}_i) \right\},
\]

(2)

then it would be possible to use the maximum likelihood (ML) principle to find the target precedent (K-th) as followed:

\[
K = \arg \max_k L_k(\{\vec{x}_i\}).
\]

III. Method

Unfortunately, from practical point of view, the above straightforward approach to identification can’t be realized. Indeed, with this approach, for each shape of intensity (1) the shift of the coordinates \( \vec{x} \) origin, for example, will lead to a precedent different from the original one (with new parameters \( \vec{m}_{k,j} \) that differ from the old ones by the shift). Taking into account this and a number of other reasons, we consider the identification of the tested point-set and the target precedent from DB as a coincidence of their shapes \( \lambda(\vec{x}) \) and \( \lambda_k(\vec{x}) \) up to affine plane transformation (change in coordinates origin on \( \vec{t} \) and change of scale on \( s \)): \( \lambda(\vec{x}) \sim \lambda_k(s\vec{x} + \vec{t}) \). The log-likelihood function for this conditions should be rewritten as:
\[ L_k(\{x_i\}) = \ln \left[ \int \int \rho_{appr}(\vec{t}, s) \prod_{i=1}^{n} p(\vec{x}_i | \vec{t}, s, k) \, d\vec{t} \, ds \right], \]
\[
= \frac{N_k}{2\pi} \sqrt{\frac{A_{k,j}}{s^2}} \exp \left\{ -\frac{1}{2} Q_{k,j}(s\vec{x} + \vec{t}) \right\},
\]
where \( \rho_{appr}(\vec{t}, s) \) is the a priori probability distribution of not related to the precedent description parameters \( \vec{t} \) and \( s \).

In order to give the log-likelihood function (4) an appropriate EM algorithm form, in addition to the point-set of registered data \( \{x_i\} \), let us introduce the hidden variables \( \{z_i\} \), where \( z_i \in \{1, \ldots, N_k\} \) is an indicator of the belonging point \( \vec{x}_i \) to the component \( z_i \) of the mixture. From this point of view \( p(\vec{x} | \vec{t}, s, k) \) (4) is the marginal distribution of \( \vec{x} \) from the joint distribution \( p(\vec{x}, z | \vec{t}, s, k) \) of the complete – observed and hidden data \( (\vec{x}, z) \). Taking this into account, we rewrite (4) as follows:

\[ L_k(\{x_i\}) = \ln \left[ \int \int \rho_{appr}(\vec{t}, s) \sum_{z_1, \ldots, z_n} \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k) \, d\vec{t} \, ds \right], \]
\[
= \frac{N_k}{2\pi} \sqrt{\frac{A_{k,j}}{s^2}} \exp \left\{ -\frac{1}{2} Q_{k,j}(s\vec{x} + \vec{t}) \right\}. \tag{5}
\]

The computational aspects associated with calculating \( L_k(\{x_i\}) \) (5) can be significantly complicated by the necessity to determine the sums for \( \{z_i\} \) and integrals over \( \vec{t} \) and \( s \). In order to avoid this complication, let us introduce the conditional distribution of \( (\{z_i\}, \vec{t}, s) \) (for a given \( \{x_i\} \)):

\[ p(\{z_i\}, \vec{t}, s | \{x_i\}, k) = \frac{p(\{x_i\}, \{z_i\}, \vec{t}, s | k)}{p(\{x_i\} | k)} = \frac{\rho_{appr}(\vec{t}, s) \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k)}{\int \int \rho_{appr}(\vec{t}, s) \sum_{z_1, \ldots, z_n} \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k) \, d\vec{t} \, ds}. \tag{6}
\]

Substituting left side (6) denominator into (7), we get:

\[ L_k(\{x_i\}) = F_{k,vw}(\{x_i\}) + D_{k,vw}(\{x_i\}), \tag{7}
\]
\[ F_{k,vw}(\{x_i\}) = \int \int \sum_{z_1, \ldots, z_n} v_{z_1, \ldots, z_n} w(\vec{t}, s) \times \int \int \rho_{appr}(\vec{t}, s) \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k) \, d\vec{t} \, ds \times \ln \frac{p(\{z_i\}, \vec{t}, s | \{x_i\}, k)}{v_{z_1, \ldots, z_n} w(\vec{t}, s)} d\vec{t} ds, \tag{8}
\]
\[ D_{k,vw}(\{x_i\}) = -\int \int \sum_{z_1, \ldots, z_n} v_{z_1, \ldots, z_n} w(\vec{t}, s) \times \ln \frac{p(\{z_i\}, \vec{t}, s | \{x_i\}, k)}{v_{z_1, \ldots, z_n} w(\vec{t}, s)} d\vec{t} ds. \tag{9}
\]

The first term \( F_{k,vw}(\{x_i\}) \) (8) in (7) is usually called the “free energy”, and it can be, with accuracy of averaging over \( v_{z_1, \ldots, z_n}, w(\vec{t}, s) \) (and the information term – \( \langle \ln v_{z_1, \ldots, z_n} w(\vec{t}, s) \rangle_{vw} \) formally obtained from \( L_k(\{x_i\}) \) (5) by permuting the sum and integral operations with the logarithm. This, obviously, greatly simplifies the calculations. The second term \( D_{k,vw}(\{x_i\}) \) (9) in (7) is the Kullback–Leibler divergence of the distribution density \( p(\{z_i\}, \vec{t}, s | \{x_i\}, k) \) with respect to \( v_{z_1, \ldots, z_n} w(\vec{t}, s) \). It is always non-negative and vanishes in the only case - if the first distribution coincides with the second. That is why \( F_{k,vw}(\{x_i\}) \) (8) is always a lower bound estimate of \( L_k(\{x_i\}) \) (5), and this estimate is as better as the free energy is greater (or divergence \( D_{k,vw}(\{x_i\}) \) is smaller). Based on the above analysis, we can formulate the following variational method (VM) of the likelihood function calculation:

\[ L_k(\{x_i\}) \approx \max_{v_{z_1, \ldots, z_n}, w(\vec{t}, s)} F_{k,vw}(\{x_i\}). \tag{10}
\]

Let us note, that the equality in (10) is possible only in the case when \( p(\{z_i\}, \vec{t}, s | \{x_i\}, k) \) can be factorized to the product of the hidden variables \( \{z_i\} \) and parameters \( (\vec{t}, s) \) distributions.

Solving the variational problem for the functional \( F_{k,vw}(\{x_i\}) \) (8) in the usual way (using the Lagrange multipliers method), we obtain the following system for optimal solutions:

\[ w(\vec{t}, s) = \frac{\rho_{appr}(\vec{t}, s) \exp \left\{ \sum_{z_1, \ldots, z_n} v_{z_1, \ldots, z_n} \ln \left[ \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k) \right] \right\}}{\sum_{\vec{t}, s} \exp \left\{ \int \int w(\vec{t}, s) \ln \left[ \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k) \right] d\vec{t} ds \right\}}, \tag{11}
\]
\[ w(\vec{t}, s) = \frac{\rho_{appr}(\vec{t}, s) \exp \left\{ \sum_{z_1, \ldots, z_n} v_{z_1, \ldots, z_n} \ln \left[ \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k) \right] \right\}}{\sum_{\vec{t}, s} \exp \left\{ \int \int w(\vec{t}, s) \ln \left[ \prod_{i=1}^{n} p(\vec{x}_i, z_i | \vec{t}, s, k) \right] d\vec{t} ds \right\}}, \tag{12}
\]
where \( \Sigma_{we} \) and \( \Sigma_v \) are the normalization constants (partition functions). The system of equations (11) can be simplified to:

\[ w(\vec{t}, s) = \frac{\rho_{appr}(\vec{t}, s) \prod_{i=1}^{n} W(\vec{x}_i | \vec{t}, s)}{\sum_{\vec{t}, s} \prod_{i=1}^{n} W(\vec{x}_i | \vec{t}, s)}, \]
\[ W(\vec{x} | \vec{t}, s) = \exp \left\{ \sum_{z_1} V_z(\vec{x}) \ln \left[ p(\vec{x}, z | \vec{t}, s, k) \right] \right\}, \]
\[ v_{z_1, \ldots, z_n} = \prod_{i=1}^{n} V_z(\vec{x}_i), \tag{12}
\]
\[ V_z(\vec{x}) = \frac{1}{\Sigma_v} \exp \left\{ \int \int w(\vec{t}, s) \ln \left[ \prod_{i=1}^{n} p(\vec{x}, z | \vec{t}, s, k) \right] d\vec{t} ds \right\}, \]
and takes the final explicit form when \( p(\vec{x}, z | \vec{t}, s, k) \) is substituted into it from (5):

\[
w(\vec{t}, s) = \frac{s^{2n} \rho_{apr}(\vec{t}, s)}{\sum_{\vec{w}}} \prod_{i=1}^{n} W(\vec{x}_i | \vec{t}, s),
\]

\[
W(\vec{x} | \vec{t}, s) = \exp \left\{ -\frac{1}{2} \sum_{i=1}^{N_n} V_z(\vec{x}) Q_{k,z} \left( s\vec{x} + \vec{t} \right) \right\}, \quad (13)
\]

\[
v_{z_1, \ldots, z_n} = \prod_{i=1}^{n} V_{z_i}(\vec{x}_i),
\]

\[
V_z(\vec{x}) = \frac{p_{k,z} \sqrt{\det \{ A_{k,z} \}}}{\Sigma_{V}} \times \exp \left\{ -\frac{1}{2} \int \int w(\vec{t}, s) Q_{k,z} \left( s\vec{x} + \vec{t} \right) d\vec{t} ds \right\},
\]

Sometimes, with a suitable choice of proper \( \rho_{apr}(\vec{t}, s) \), conjugate, for example, to \( s^{2n} \prod_{i=1}^{n} W(\vec{x}_i | \vec{t}, s) \), system of equations (13) can be reduced to (nonlinear) algebraic system and solved by well-known methods. However, it is possible to go along a different way – assuming that the number of \( \Delta \) parameter \( k \) the sizes of the mixture components, and the distribution of \( \vec{t}, s \) arguments are distributed independently and, for example, the \( w \) characteristic sizes) – let us try to find it in the asymptotic behavior before we find this solution, let us make a couple of simplifications, also related to asymptotics \( n \rightarrow 1 \).

Namely, with respect to \( \rho_{apr}(\vec{t}, s) \), let us assume that its arguments are distributed independently and, for example, the distribution of \( \vec{t} \) is well approximated by a Gaussian one with zero mean and a standard deviation \( \Delta_t \), that exceeds at least the sizes of the mixture components, and the distribution of \( s \) is well approximated by a gamma distribution with shape parameter \( k \), mean \( \bar{s} \) and, correspondingly, standard deviation \( \Delta_s = \bar{s}/\sqrt{k} \). We note that in the asymptotic case \( n \gg 1 \), the product \( s^{2n} \rho_{apr}(\vec{t}, s) \) in (13) will be well approximated by the old Gaussian distribution for \( \vec{t} \) multiplied by also Gaussian distribution with the mean \( \bar{s}n \sim 2n\bar{s}/k \) and standard deviation \( \Delta_{s,n} \sim \sqrt{2n\Delta_s}/\sqrt{k} \) for the \( s \). In these approximations, the logarithm of \( w(\vec{t}, s) \) (13) becomes quadratic in the parameters \( \vec{t}, s \), so its extremum can be easily found from the following linear system:

\[
P = \begin{pmatrix}
    \text{Tr} \left[ \sum_{i=1}^{N_n} \gamma_z A_{k,z} B_i \right] \\
    \frac{1}{\Sigma_{n,z}} \\
    \left[ \text{Tr} \left[ \sum_{i=1}^{N_n} \gamma_z A_{k,z} X_i \right] \right]^T
\end{pmatrix} + \frac{1}{\Sigma_{n,z}} \left[ \sum_{i=1}^{N_n} \gamma_z A_{k,z} X_i \right] + \frac{1}{\bar{s}^2} E_2;
\]

\[
\sigma = \sum_{i=1}^{N_n} X_i^T \gamma_z A_{k,z} \bar{m}_{k,z} + \bar{S}_n \Delta_{n,z}^{-1} \bar{\sigma} = \left( \gamma_z \right)^T \left[ \sum_{i=1}^{N_n} \gamma_z A_{k,z} \bar{m}_{k,z} \right];
\]

Substituting \( w(\vec{t}, s) = \delta(\vec{t} - \vec{t}_m) \delta(s - s_m) \) into (13) we immediately obtain the following relation:

\[
V_z(\vec{x}) = \frac{p_{k,z} \sqrt{\det \{ A_{k,z} \}}}{\Sigma_{V}} \exp \left\{ -\frac{1}{2} Q_{k,z} \left( s\vec{x} + \vec{t}_m \right) \right\} = \frac{p(\vec{x}, z | \vec{t}_m, s_m, k)}{p(\vec{x} | \vec{t}_m, s_m, k)}. \quad (16)
\]

Thus, given an initial, neutral approximation for the parameters \( (\vec{t}, s) \), for example, putting \( \vec{t}(0) = \vec{0}, s(0) = 1 \), on the base of (16) we can find initial approximation to \( \{V_z(\vec{x})\} \) for all \( \{\vec{x}\} \) and all corresponding \( \{z\} \). Using them, we can compute by formulas (15) the matrix \( P \) and the right-hand side \( \{\sigma, \bar{\sigma}\} \) of (14). Getting the solutions of this liner system and putting them as the refinements \( \{\vec{t}(1), s(1)\} \) of the initial parameters, we can refine \( \{V_z(\vec{x})\} \) (16), etc. So, we get a recursive algorithm in a form very reminiscent of EM:

\[
E: \quad \text{for} \quad i \in \{1, \ldots, n\}, \quad z \in \{1, \ldots, N_k\} \quad \text{calculate}
\]

\[
V^{(j)}_z(\vec{x}_i) = \frac{p(\vec{x}_i, z | \vec{t}_{(j-1)}, s_{(j-1)}, k)}{\sum_{z=1}^{N_k} p(\vec{x}_i, z | \vec{t}_{(j-1)}, s_{(j-1)}, k)};
\]

\[
M: \quad \text{for} \quad z \in \{1, \ldots, N_k\} \quad \text{calculate}
\]

\[
\gamma_z = \sum_{i=1}^{n} V^{(j)}_z(\vec{x}_i); \quad \gamma_z = \frac{1}{N_k} \sum_{i=1}^{n} V^{(j)}_z(\vec{x}_i) \vec{x}_i \quad \text{find} \quad P, \quad \sigma, \quad \bar{\sigma} \quad \text{and solve}
\]

\[
P \left( \frac{s_m}{\vec{t}_m} \right) = \begin{pmatrix}
    \sigma \\
    \bar{\sigma}
\end{pmatrix} \quad (17)
\]

The constructed algorithm (17) recursively, with increasing number of indexed by \( j \) iterations, refines the parameters \( (\vec{t}_{(j)}, s_{(j)}) \). When the iterations begin to show signs of stabilization for any of the calculated values \( \vec{t}_{(j)}, s_{(j)} \), \( \{V^{(j)}_z(\vec{x}_i)\} \), the algorithm can be stopped. On the basis of these values full solutions (13) \( v_{z_1, \ldots, z_n} = \prod_{i=1}^{n} V^{(j)}_z(\vec{x}_i) \), \( w(\vec{t}, s) = \delta(\vec{t} - \vec{t}_{(j)}) \delta(s - s_{(j)}) \) can be found. But, in fact, it is not needed. It turns out that the expression for the free energy \( F_{k,vw} \) (\( \{\vec{x}_i\} \)) (8), which is the
goal of the computation, depends only on the estimates \( \tilde{t}_{(j)}, s_{(j)} \). Indeed, substituting the solutions written above in (8) we obtain:

\[
F_{k,vw}(\{\tilde{x}_i\}) = \sum_{z_1,\ldots,z_n} v_{z_1,\ldots,z_n} \times \\
\times \ln \left[ \rho_{apr}(\tilde{t}_{(j)}, s_{(j)}) \prod_{i=1}^{n} p(\tilde{x}_i | \tilde{t}_{(j)}, s_{(j)}, k) \right] = (18)
\]

where it is taken into account that information \( \langle \ln \left[ w(\tilde{t}, s) \right] \rangle_w \) for the singular distribution \( w(\tilde{t}, s) \) is zero (at list after finite approximation). The value of \( \wp \) under the logarithm in (18) is the joint distribution of the recorded PPP data \( \{\tilde{x}_i\} \) and the current estimates of the parameters \( (\tilde{t}, s) \). Using all these data \( \ln \wp \) can be easily calculated starting directly from the original definitions (see (4)), it has a transparent interpretation and, therefore, as a criterion controlling the execution of iterations of the algorithm (17), \( \ln \wp \) is the best choose.

IV. CONCLUSIONS

In this paper a complete solution of the problem of Poison process intensity identification is presented. For the implementation of the solution, a computational scheme is proposed. This scheme is very close in structure to the popular in machine learning EM algorithm. Since many technical aspects of the computer implementation of EM algorithms are now well-developed [6], one can expect that the implementation of the proposed solution will be also quick and effective.

REFERENCES

A new mathematical method for automated identification of neurons on microscopic images

I.B. Gurevich, V.V. Yashina, A.A. Vladimirov
Department of Mathematical and Applied Problems of Image Analysis
Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences
Moscow, Russian Federation
igourevi@ccas.ru, werayashina@gmail.com, vladimirov@shkal.math.msu.su

A.T. Tleubaev, A.M. Ospanov
Faculty of Computational Mathematics and Cybernetics
Moscow State University
Moscow, Russian Federation
adil.tleubayev.3.14159@gmail.com, ayat.ospanov@gmail.com

Abstract—This work is devoted to the formulation and solution of mathematical problems associated with the development of a method that ensures the identification of images of neurons on microscopic images (MIN) of brain sections of experimental animals with a specified accuracy. The images are color photographs of the treated brain tissue of animals. The mathematical apparatus developed by the authors of the chapter for the analysis of MIN is based on the use of algorithms of the mathematical theory of pattern recognition and image analysis, information theory and methods of mathematical statistics. The developed method allows to find and allocate informative objects on microscopic images of brain slices while setting constraints on the shape, size, topology and smoothness characteristics of the boundaries of objects.

Keywords—image processing, neuron detection, biomedical imaging, automation of scientific research, mathematical morphology

1. INTRODUCTION

One of the most important tasks of neuroscience is the development of experimental models of neurodegenerative diseases (NDD) (creation of models of neuronal death in certain parts of the brain). These models will be used to develop new methods and technologies for the diagnosis and treatment of neurodegenerative diseases.

Model development could have been much faster and would have been more cost effective in reducing the time and material costs of morphological research by automating and optimizing the methods of processing and analyzing experimental material based on the use of modern algebraic and mathematical theories of image analysis and pattern recognition. It is shown that automation of the analysis of microscopic images of neurons (MIN) and their outgrowths allow to reduce material costs by an order of magnitude, and time costs by two orders of magnitude [10][11].

Analysis of the MIN is a rather complex multifactorial task. The solution of this problem will lead to the elimination of various kinds of problems related to the construction of NDD models, the automation of morphological studies, the analysis of experimental materials and other tasks.

Typically, the process of detecting and isolating neurons on histological microscopic images includes the following main steps: preprocessing, segmentation and classification. The need for preliminary processing of MIN is caused by the presence of noise, low resolution of MIN and heterogeneity of contrast associated with errors in staining of slices. To improve the quality of MIN, standard image processing operations are usually used – anti-aliasing, reverse convolution, morphological filtration and some others [14], [16].

In published works devoted to the automatic or semi-automatic allocation of neurons on MIN, segmentation algorithms mainly rely on algorithms based on threshold processing (e.g., [2]), morphological operations (e.g., Potts models [15]), the method of watersheds [17] or active contours [12]. Subsequent classification is carried out, for example, using Bayesian procedures, analysis of the main components or methods of machine learning [1].

An important stage in the analysis of MIN of brain slices is the morphological characterization of the extracted neurons. In recent years, a wide range of specific morphological parameters has been determined for an effective mathematical characterization of the neurons morphology, including their nuclei and outgrowths [14].

The authors of the report previously developed a method for identifying neurons, presented in the article. This method is based on the use of methods of mathematical morphology. The filtration and normalization methods were applied to the image, then the stage of binarization of the image was carried out. The binarized image was segmented, calculation of histograms of the oriented gradients (HOG-descriptors) was carried out for each segment. Classification of segments into neuron/background was performed using the support vector machine (SVM) with linear kernel [3].

The method developed earlier [10] showed good results for the currently presented class of images of neurons. The

This work was supported in part by the Russian Foundation for Basic Research (project no. 17-07-01482).
preparation of the brain tissues of experimental animals is a laborious process, and not always the same. During the experiments, five groups of original images were distinguished, visually different from each other. For certain groups of images, the previously implemented method showed unacceptable results.

This article presents a new automatic method for neurons extraction on microscopic images of dopaminergic (DA-ergic) neurons of brain sections of experimental animals. The method is based on the joint use of the mathematical theory of pattern recognition and image analysis, information theory and methods of mathematical statistics. The paper describes the results of evaluating the effectiveness of the method and comparing its work with the one presented earlier [10].

The work consists of the following sections: 1) Problem statement; 2) The method for identification of neurons on MIN; 3) Description of the features based on the moments; 4) Results; 5) Conclusion.

II. PROBLEM STATEMENT

The main task of the work is the extraction of neurons on digital microscopic images of dopaminergic (DA-ergic) neurons of experimental animals’ brain slices.

For this work we used images of substantia nigra pars compacta (SNC) of brain slices of mice (visually different images obtained in different experiments are shown in Fig. 1, Fig. 2). Images were obtained at the Laboratory of hormonal regulation of the FSUE Institute of Developmental Biology of RAS named by N.I. Koltsov.

MIN, depending on the group, have either 4080 by 3072 pixels or 2040 by 1536 pixels (the resolution of the images is the same).

Detection and extraction of neurons on two-dimensional microscopic images of brain sections are complicated by the following factors [9]:

- Expect neurons, there are extra non-neuron objects on the MIN (dirt particles, coloring errors, tissue folds, blood vessels, synapses, axons and other parts of neuronal connections);
- Connections between neurons (thin threads in large numbers) form a complex texture, on the background of which the extraction of neurons becomes more complicated;
- Neurons on MIN can vary very much in size and shape;
- Neurons can be damaged during preparation, which affects their shape, and, as a consequence, leads to a large number of different cells on the slice;
- Neurons can “stick together” or overlap;
- MIN contains areas of the brain with different quantitative distribution of neurons per unit area;
- Microscopic images of different sections of the brain vary greatly in contrast and brightness.

The developed method is designed to identify a lot of small informative extended objects with a clearly visible brightness inclusion of the oval shape on microscopic images of the front sections of SNC. The result of the method is the extracted image areas containing neurons.

A new mathematical method for identifying neurons should work equally effectively on different groups of images.

III. THE METHOD FOR IDENTIFICATION OF NEURONS ON MIN.

The theoretical basis for the development of methods for analyzing MIN was the descriptive approach to the analysis and understanding of images by I.B. Gurevich [10]. In accordance with his concept, mathematical methods are represented in the form of specialized algorithmic schemes. Such schemes include the basic stages of extracting information from images: 1) preliminary processing (improving the quality of images, removing irrelevant details and fragments, statistical and logical filtering); 2) image analysis (detection of objects and selection of their outlines, segmentation, extraction of features describing the composition and content of the image, search, extraction and construction of individual objects that reflect the specific features of the analyzed images, etc.); 3) constructing the description of the objects represented on the analyzed images; 4) classification of images and objects on them; 5) recognition.

Algorithmic scheme that implements the developed method of object extraction and identification includes the following stages.

Stage 1. Preprocessing of MIN.
Whilst developing a new method for identifying neurons, the work of the previously constructed method was analyzed. The main drawback of the method was the stage of binarization: some of the neurons were identified as a background; many background areas were identified as potential neurons; the boundary of a neuron was incorrectly defined – background parts were added in its component of connectivity. Segmentation based on the extraction of connected areas of a binary image, provided the allocation of about 70% of neurons for the next stage of classification. As a result of this, a large number of errors of the Type I and Type II were made at the recognition stage.

In the new method, the image preprocessing was not carried out in general. In the particular case of describing the objects represented in the analyzed images, the method was compared on the source images and on the normalized images using HOG descriptors.

Stage 2. Analysis of MIN.

At the MIN analysis stage, a sliding window method was used to extract areas containing objects. The original image was divided by a sliding window into square areas of different scales: 24, 36, 48 and 60 pixels. The step of the window shift is: 6, 9, 12 and 15 pixels. An example of an extraction for the neuron classifier using the sliding window method is shown in Figures 3, 4.

Stage 3. Construction of the description of the objects represented on the analyzed images.

To describe the objects represented in the analyzed images, the following widely used in the field of pattern recognition and image analysis specific features were extracted:

- The intensity of the color channels of the image;
- HOG descriptor – histogram of oriented gradients. The algorithm for obtaining these characteristics based on the division of the image into a certain number of zones, followed by the calculation of the predominant direction of the gradient in each zone [5];
- Histogram of local binary patterns (LBP) [6];

Fig. 3. Examples of “neuron” objects

Fig. 4. Example of “background” objects

- Features obtained from the pixel brightness with help of principal component analysis (PCA) procedure [13];
- Features based on moments (specific features developed to solve this problem).

Features based on moments are constructed in accordance with the concept that rectangular areas of the image are considered as elements of spaces defined on the corresponding rectangles by functions. As the characteristics, the coefficients of the expansion of these functions are used for the bases of the corresponding spaces.

More specifically, the area with the pixel \((a + 1, b + 1)\) in the upper left corner and the pixel \((a + n, b + n)\) in the lower right corner was given by a set of moments

\[
M_k = \int_a^{a+n} \int_b^{b+n} \rho_k \left( \frac{x-a}{n}, \frac{y-b}{n} \right) f(x,y) dx dy, \quad k \in \mathbb{N}
\]

where \(f\) is a function which values on all possible squares of the species \((a - 1, a) \times (b - 1, b)\) coincide with the brightness of the pixels \((a, b)\) of the original image. These characteristics are described in more detail in the next section.

Stage 4. Classification of images and objects presented on them.

At the stage of classification, the following algorithms were used:

- K-nearest neighbors algorithm (kNN) is a metric algorithm for automatic classification of objects. The main principle of the method of the nearest neighbors is that the object is assigned to the class that is the most common among the neighbors of this element [18];
- Random forest – the algorithm of machine learning, consisting in the use of a committee (ensemble) of decision trees. The decision tree is a classifier represented in the form of a tree [4];
- XGBoost (Extreme Gradient Boosting) [7];
- Logistic regression – the method of linear classification that allows estimating the a posteriori probabilities of
Step 5. Recognition.

The classification allowed to choose the most successful pair: features + classifier, which was applied to the initial images for the identification of neurons.

IV. DESCRIPTION OF THE FEATURES BASED ON THE MOMENTS

From a mathematical point of view, a monochrome rectangular image of size \( n \times m \) admits of a finite Borel measure as given on the set \([0, n] \times [0, m]\) for which the value of the measure of rectangles of the form \([a,1, a] \times [b,1, b]\) gives the brightness of the corresponding pixels. Taking into account the usual limitations for permissible brightness in applications, the class of mathematical objects that model images can be further narrowed from the space of measures to the space of non-negative square integrable functions. In particular, assuming that the brightness distribution is uniform within each pixel given by the space \( L_2([0,1] \times [0,1]) \), bearing in mind that any square carries out the standard orthogonalization procedure, we obtain a piecewise constant function of the class \( \text{class of mathematical objects that model images can be further narrowed from the space of measures to the space of non-negative square integrable functions. In particular, assuming that the brightness distribution is uniform within each pixel given by the space } L_2([0,1] \times [0,1]) = \{ x(\cdot) \in L_2([0,1] \times [0,1]) : \text{measurable} \} \).

Let's consider a question on obtaining a set of characteristics of the image hypothetically suitable for the purposes of recognition. We confine ourselves here to the case of the unit square \([0,1] \times [0,1]\), bearing in mind that any square image can be reduced to an image on the unit square by means of a similarity transformation.

It is well known that the system of polynomials \( \{ x^{k} y^{l} \} \) is fundamental in the space \( L_2([0,1] \times [0,1]) \). Correspondingly, by ordering the indicated system according to the lexicographic order of pairs of the form \( \{ k \times i, l \times i \} \) and carrying out the standard orthogonalization procedure, we obtain an orthonormal basis \( \{ \rho_k \} \) of the space \( L_2([0,1] \times [0,1]) \). The vectors of this basis have the following form:

\[
\rho_1(x, y) = \sqrt{12} \left( x - \frac{1}{2} \right), \\
\rho_2(x, y) = \sqrt{12} \left( y - \frac{1}{2} \right), \\
\rho_3(x, y) = 180 \left( x^2 - x - \frac{11}{6} \right), \\
\rho_4(x, y) = 12 \left( xy - \frac{x}{2} - \frac{y}{2} + \frac{1}{4} \right), \\
\rho_5(x, y) = 180 \left( y^2 - \frac{y}{2} - \frac{11}{6} \right), \\
\rho_6(x, y) = 2800 \left( x^3 - \frac{3x^2}{2} - \frac{27x}{5} + \frac{59}{20} \right), \\
\rho_7(x, y) = 2160 \left( x^2y - \frac{x^2}{2} - xy - \frac{x}{2} - \frac{11y}{6} + \frac{11}{12} \right), \\
\rho_8(x, y) = 2160 \left( xy^2 - \frac{y^2}{2} - xy - \frac{y}{2} - \frac{11x}{6} + \frac{11}{12} \right),
\]

\[
\rho_9(x, y) = \sqrt{2800} \left( y^3 - \frac{3y^2}{2} - \frac{27y}{5} + \frac{59}{20} \right), \\
\rho_{10}(x, y) = 210 \left( x^4 - 2x^3 - \frac{75x^2}{7} + \frac{82x}{7} - \frac{179}{70} \right), \\
\rho_{11}(x, y) = \sqrt{33600} \left( x^3y - \frac{x^3}{2} + \frac{3x^2y}{4} - \frac{27xy}{5} + \frac{27x}{10} + \frac{59y}{20} - \frac{59}{40} \right), \\
\rho_{12}(x, y) = 180 \left( x^2y^2 - x^2y - \frac{11x^2}{6} - xy^2 + xy + \frac{11x}{6} - \frac{11y^2}{6} + \frac{11y}{6} - \frac{23}{36} \right), \\
\rho_{13}(x, y) = \sqrt{33600} \left( xy^3 - \frac{3xy^2}{2} + \frac{3y^2}{4} - \frac{27xy}{5} + \frac{27y}{10} + \frac{59x}{20} - \frac{59}{40} \right), \\
\rho_{14}(x, y) = 210 \left( y^4 - 2y^3 - \frac{75y^2}{7} + \frac{82y}{7} - \frac{179}{70} \right).
\]

Functional series \( \sum_{k=0}^{\infty} c_k \rho_k \), where \( c_k \) are coordinates shown next

\[
c_k = \int_0^1 \int_0^1 f(x, y) \rho_k(x, y) dx dy,
\]

converge in \( L_2([0,1] \times [0,1]) \) space to the original function \( f \), and therefore the sequence \( \{ c_k \}_{k=0}^\infty \) gives in the aggregate a complete description of the original image. For the purposes of recognition, this sequence as a whole, however, is certainly excessive. The nature of the subject area (problems associated with the detection of structures visually close to the rings) makes it possible to make an a priori assumption about the possibility of restricting ourselves to considering a set of coordinates corresponding to basic polynomials of order \( \leq 4 \), that is, a finite list \( \{ c_k \}_{k=0}^{14} \). Indeed, the distribution of brightness with a minimum value on the ellipse

\[
(X - X_0)^T A (X - X_0) = 1, \quad X = \begin{pmatrix} x \\ y \end{pmatrix}, \quad X_0 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}
\]

where \( A \) is arbitrarily fixed positive \( 2 \times 2 \)-matrix, is given by a fourth-order polynomial of the form

\[
f(x, y) = [(X - X_0)^T A (X - X_0)]^2
\]

It was this set of parameters that was used in the described method. More precisely, the basis was a slightly modified set of values

\[
\zeta_k = \frac{c_k}{\sqrt{0.01 \cdot c_0^2 + \sum_{i=1}^{14} c_i^2}}, \quad k = 1, 14
\]

(1)
in the construction of which the level of average brightness and the overall contrast of the original image was leveled.

The required coordinates $c_k$ attributed to the model square $[0,1] \times [0,1]$ selected within an arbitrary image $f \in L_2([0,n] \times [0,m])$ of section with side $N$ and the coordinates of the upper left corner $(a, b)$ obviously satisfy the equation

$$c_k = \frac{1}{N^2} \int_{a}^{a+N} \int_{b}^{b+N} \rho_k \left( \frac{x - a}{N}, \frac{y - b}{N} \right) f(x, y) \, dx \, dy$$

Accordingly, they can be calculated as linear combinations of moments of the form

$$M_{k\ell}(a, b, N) = \int_{0}^{a} \int_{0}^{b} x^k y^l f(x, y) \, dx \, dy$$

(2)

with coefficients representing some polynomials in the parameters $a, b$ and $N^{-1}$. Therefore, the question of the effective computation of the coordinates $c_k$ reduces to the question of the effective computation of the moments $M_{k\ell}(a, b, N)$. The natural solution of the last question is to use a modified integral representation of the image under which each pixel $(a, b)$ is put in correspondence set of values

$$M_{k\ell}(a, b) = \int_{0}^{a} \int_{0}^{b} x^k y^l f(x, y) \, dx \, dy, \quad k + l \leq 4.$$

In this case, moments (2) can be quickly calculated by formulas

$$M_{k\ell}(a, b, N) = M_{k\ell}(a + N, b + N) + M_{k\ell}(a, b) - M_{k\ell}(a + N, b) - M_{k\ell}(a, b + N)$$

It should be noted that the choice made by us as the basis functions of just the polynomials is not in the last instance due to the possibility of determining the corresponding characteristics of arbitrary square sections on the basis of fast operations on some common integral representation of the original image.

**V. RESULTS**

Table 1 shows the results of comparing the work of the method with different features and classifiers. In the columns of the table the used features are shown, in the rows the classifier and the results of using this classifier with the features are indicated in pairs. The cells have the following values: AUC-ROC (the area under the ROC curve), II type error, I type error.

The pair of moments based features and random forest classifier showed the best results. This pair showed the best AUC-ROC score, low value of the type I error and acceptable values of the type II error. Further, in the identification task, the method based on this pair of features and classifier was used.

Fig. 5 shows a picture with manually marked neurons. Fig. 6 is a snapshot obtained after the application of the neuron identification method. On average, the method identifies about 72% of all neurons per image. Approximately 30% of the identification result is the background or tissue objects, not neurons.

The method most often identifies neurons of medium size, but the main part of errors are made on small and large sized

**TABLE I. Comparison of classifiers (In the table cell there are three numbers: AUC-ROC, II type error, I type error from top to bottom, respectively.) Columns “S” and “N” mean “Source” and “Normalized” respectively.)**

<table>
<thead>
<tr>
<th>The intensity of the color channels</th>
<th>HOG</th>
<th>LBP</th>
<th>PCA</th>
<th>Moments</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.4945</td>
<td>0.92540</td>
<td>0.89688</td>
<td>0.82754</td>
<td>0.95046</td>
</tr>
<tr>
<td>0.05087</td>
<td>0.02585</td>
<td>0.01167</td>
<td>0.26939</td>
<td>0.05087</td>
</tr>
<tr>
<td>0.18407</td>
<td>0.28606</td>
<td>0.38723</td>
<td>0.21475</td>
<td>0.17827</td>
</tr>
</tbody>
</table>

**Random forest**

<table>
<thead>
<tr>
<th></th>
<th>S</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.97749</td>
<td>0.95263</td>
<td>0.94688</td>
</tr>
<tr>
<td>0.12010</td>
<td>0.06422</td>
<td>0.06588</td>
</tr>
<tr>
<td>0.04892</td>
<td>0.15837</td>
<td>0.18325</td>
</tr>
</tbody>
</table>

**XGBoost**

<table>
<thead>
<tr>
<th></th>
<th>S</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.97559</td>
<td>0.95581</td>
<td>0.94951</td>
</tr>
<tr>
<td>0.08256</td>
<td>0.07673</td>
<td>0.07673</td>
</tr>
<tr>
<td>0.07794</td>
<td>0.13681</td>
<td>0.15174</td>
</tr>
</tbody>
</table>

**Logistic regression**

<table>
<thead>
<tr>
<th></th>
<th>S</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.94078</td>
<td>0.96292</td>
<td>0.94795</td>
</tr>
<tr>
<td>0.11009</td>
<td>0.06922</td>
<td>0.07506</td>
</tr>
<tr>
<td>0.14013</td>
<td>0.12106</td>
<td>0.15091</td>
</tr>
</tbody>
</table>

Fig. 5. Image with manually marked neurons

Fig. 6. Image with identified neurons
neurons. It is planned to expand the sliding window size range in the future.

In a number of cases, the method identifies objects that are very similar to neurons as neurons, but they have not been labeled. Clustering of objects extracted by the sliding window on MIN was carried out. It turned out that some marked neurons are very similar to background objects containing clots of tissue. Numerous meetings with experts also showed that the expert doctor sometimes does not know with 100% accuracy whether this or that object belongs to neurons.

VI. CONCLUSIONS

A new method for identifying neurons on microscopic images was described in this work. The conducted experiments showed that the method shows not quite acceptable results. In the future, it is planned to improve the results by increasing the training sample when new images are received. Results improvement are also planned by increasing the window size range for the sliding window, which will allow better identification of small and large-sized windows. It is also intended to use the results of cluster analysis to develop multi-algorithmic classifiers, which are specific for different classes of neurons.

REFERENCES

New Mathematical Methods for Automation of Angiographic Image Analysis of the Human Fundus

I.B. Gurevich, V.V. Yashina  
Department of Mathematical and Applied Problems of  
Image Analysis  
Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences  
Moscow, Russian Federation  
igourevi@ccas.ru, werayashina@gmail.com

A.M. Ospanov, A.T. Tleubaev  
Faculty of Computational Mathematics and Cybernetics  
Lomonosov Moscow State University  
Moscow, Russian Federation  
ayat.ospanov@gmail.com, adil.tleubayev.3.14159@gmail.com

A.A. Fedorov  
Laboratory of Fundamental Research in Ophthalmology  
Scientific Research Institute of Eye Diseases  
Moscow, Russian Federation  
gdn@mail.ru

Abstract—new mathematical methods for automatic angiographic image analysis of the human fundus are presented. The two problems were selected for the solution: a) retinal vasculature analysis; b) detection of ischemic zones. The initial images are colored photographs of the retinal vasculature of the human fundus with a brightly lit part in the center and a blackout along the edges. A mathematical model which characterizes the density of the retinal vasculature (the number of capillary loops per unit area), the average diameter of the vessels that gives an idea of the degree of the retina vascularization in normal state and vascular pathology of the vision organ has been developed. The problem of determining the area and topography of the ischemia zone (non-perfusion of vessels or hemorrhages) has been solved. The obtained data allow predicting the degree of visual disturbances at thrombosis of the retinal vessels, as well as the dynamics of the ischemic zones’ size allow to estimate the effectiveness of treatment conducted by ophthalmologists. The proposed approaches are based on the combined use of the techniques of image processing and mathematical morphology. The program implementation of the developed methods and results of the performance evaluation are presented. Computer experiments and expert estimation of ophthalmologists confirmed the feasibility of the processing and analysis automation of angiographic images using the methods developed.

Keywords—automated angiographic image analysis, biomedical imaging, fundus blood vessel detection, retinal vasculature detection, ischemia detection, automation of scientific research, mathematical morphology

I. INTRODUCTION

The work is dedicated to:

1) the formulation and solution of problems of mathematical analysis of images associated with the development of methods that provide automatic analysis of angiographic images of the human fundus and detection of the vascular bed and ischemic zones on them;
2) software implementation of the developed methods;
3) research of the developed methods and their software implementation in computational experiments.

An important direction in the modeling of diseases and their diagnosis is the acquisition, storage, processing and analysis of data extracted from digital images. Image is one of the most informative and widespread forms of presentation, transmission and storage of information and is now actively used as a means of presenting the results of biological and clinical research in the main sections of medical science and practical medicine.

Fundus fluorescein angiography is an important lifetime study of the state of retinal and choroidal circulation for demonstrating of some eye disease process.

Fundus angiograms allow us to judge the state of the retinal and choroidal vascular beds: their architectonics, the density of the vascular bed, the patency of the vessels, their occlusion and the presence of avascular (ischemic) zones, the dynamics of the area of ischemic sites, depending on the treatment. In addition, the state of vascular permeability, increased in newly formed vessels (for example, with diabetes) can be judged by the presence of "leakage" areas of fluorescein.

To assess the fundus angiographic picture, it is necessary to develop a mathematical model that characterizes the density of the vascular bed (the number of capillary loops per unit area),

This work was supported in part by the Russian Foundation for Basic Research (projects no. 17-07-01482, 18-57-00013)
the average diameter of the vessels, which gives an idea of the vascularity of the retina in normal state and vascular pathology of the vision organ. In patients with thrombosis of the retinal vessels, ischemia zones (nonperfusion of vessels or hemorrhages) are formed, by the area and topography of which the extent of visual disturbances and by the dynamics of the zones’ sizes the treatment effectiveness can be predicted.

Manual analysis of angiographic images of the human fundus is a labor-intensive process that involves training and the experience of the operator. Investigation of the given images would progress much faster and be more economically efficient (its time and materials costs could be reduced) by automating and optimizing the corresponding image-processing and analysis methods based on recent advances in the mathematical theories of image analysis and pattern recognition.

To automate the processing and analysis of biology and medicine research results, mathematical and information approaches are widely used. The current state of these theories makes it possible to design special-purpose algorithms and software complexes that enable automatic information extraction from biomedical images, as well as providing a substantial basis for optimal diagnostic decision-making by practicing physicians and medical researchers in the interactive and automatic modes.

The authors of the article have been developing methods for automating the analysis of various biomedical images for a decade [3-6].

This article presents automatic methods for identifying the vascular bed and ischemic zones on angiographic images of the human fundus (Fig. 1). The proposed approaches are based on the combined use of the techniques of image processing, mathematical morphology and the analysis of specifics of angiographic images.

The paper describes the software implementation of the developed methods and the results of assessing the effectiveness of their work.

In the "Formulation of tasks" section we present the tasks for which methods were developed and analyze the given data. The classification of known methods of vascular isolation is given and the main requirements to the methods being developed.

In the section "The method of retinal vasculature detection", a description of a new mathematical method for the human fundus angiographic images analysis automation for evaluating the state of the retinal vasculature is described, as well as a step-by-step description of the developed method.

In the section "The method of ischemic zones detection", the problem of analyzing human fundus angiographic images for evaluation the density and size of ischemic zones is described.

The "Results" section presents the software implementation of the developed method and the created user interface. An analysis of the results obtained during the work of the developed method is presented.

---

II. FORMULATION OF TASKS

The initial images are color photographs of the human fundus retinal vasculature with a brightly lit part in the center and a blackout along the edges (Fig. 1).

Images visually display the shades of gray. Vessels are a branched network of blood channels emerging from the center of the optic nerve disk. Penetrating into the eyeball, fluorescein first contrasts (stains white) with choroidal vessels (with a faster blood flow), and further with retinal vessels: successively arterial vessels, capillaries and veins. Ischemia zones are dark areas with no vessels. Due to the fact that the ischemia zones are avascular zones, the fluorescein solution does not enter the tissue of this zone, thus these areas remain darkened.

The main tasks that were solved in this work are: 1) automatic detection of all links of the retinal vasculature on the human fundus angiographic images to calculate the density and the average diameter of the vessels; 2) automatic detection of ischemic zones and calculation of their required morphological characteristics.

The results of the developed methods should be: 1) binary images with detected vessels for visual analysis by ophthalmologists; 2) morphometric characteristics associated with the allocated vessels: the density, the average diameter, the standard deviation of the diameter and the total length of the vessels; 3) binary images of the detected ischemic zone and the volume of this region in μm.

The theoretical and methodological basis for automating the processing, analysis and evaluation of experimental data is the mathematical theory of pattern recognition and image analysis.

The methods used for identifying the vessels in the two-dimensional (2-D) images are generally based on the local image, taking into account the specific properties of the vascular segments. Two main approaches are normally considered for the identification of general vascular segments, associated with two different strategies for classifying each pixel as belonging to a vessel or not: pixel processing-based methods and tracking methods. The first class is divided to kernel-based and classifier-based methods by several authors [11], the second class is divided into rule-based and supervised methods mentioned in [10].
Pixel processing-based methods frequently use a two-step approach. The first step is an enhancement procedure, usually a convolution operator, with the main purpose of selecting an initial set of pixels to be further validated as vessels in the second step. Example of a solution using pixel processing-based methods is given below.

The concept of matched filter detection was proposed by Chaudhuri et al. in [1]. In this method, the authors use 12 rotated versions of a 2-D Gaussian shaped template for searching vessel segments along all possible directions. For every pixel, the maximum response to these kernels is retained. As each convolution kernel is modified by subtracting its mean value, the expected filter output is ideally zero in the background, and the resulting image can be easily thresholded to produce a binary representation of the retinal vasculature.

Tracking methods start by locating, manually or automatically, the vessel points used for tracing the vasculature, by measuring some local image properties. These methods, that only evaluate the pixels close to the initial positions, are also mentioned as exploratory algorithms. The algorithms are normally implemented as a single-pass operation, where the extraction of image features and the recognition of the vessel structure are simultaneously executed. Most of the methods reported in the literature use Gaussian functions to characterize the vessel profile. In [12], the authors report an algorithm that is initiated by the definition of the starting and ending points and is automatically followed by a matched filter for locating the vessel boundaries, tracking the midline and extracting parameters of clinical interest.

The methods proposed in this work are based on the combined use of image processing and image analysis methods using operations of mathematical morphology. The developed automation methods are represented in the form of specialized algorithmic schemes [3-6]. Algorithmic schemes are used for the program realization of the methods that they model. A description of the method is given in the next chapter.

III. THE METHOD OF RETINAL VASCULATURE DETECTION

A general description of the algorithmic scheme that models the method of retinal vasculature detection is given below and includes three stages of extracting information from images: image preprocessing, image analysis (vessel selection), and calculation of vessel morphometric characteristics.

A. Image preprocessing

Step A.1 «Brightness and color correction of the images». The original image (Fig. 1) is given in a monochromatic form, but with three-color channels. Therefore, this image is converted to a single-channel image in grayscale using equation (1).

\[
I = 0.2989 * R + 0.5870 * G + 0.1140 * B
\]  

(1)

Where: R, G, B are red, green and blue channels of the image consequently.

Step A.2 «Image normalization». For the further convenience of working with the image, it is normalized to a segment [0, 1].

Step A.3 «Logarithmic correction of the image». The camera illuminates only the rounded area, so the edges of the image are low contrast and require lighting. To lighten the dark areas, a logarithmic correction is applied: each pixel of the image is processed according to the formula (2):

\[
Output = gain \cdot \log(I + Input)
\]  

(2)

Where: gain is scaling factor, “Input” is the input image pixels, and “Output” is the output image pixels.

Step A.4 «Image denoising». The original image is quite noisy and Gauss filter is used to reduce the noise [11] (Fig. 2). The Gaussian filter also contributes well to the allocation of boundaries, because reduces the sensitivity of the boundary detector to noise.

Step A.5 «Contrast increase». As small vessels are very thin structures and usually present low local contrast, their segmentation is a difficult task. To improve the discrimination between these thin vessels and the background noise, the normalized image is processed with a set of line detection filters [2], corresponding to the four orientations (horizontal, vertical, 45° and 135°). The set of convolution kernels used in this operation is shown in Fig. 3.

![Fig. 2. Preprocessed image](image-url)

![Fig. 3. Kernels, which each respond to lines at the particular orientation](image-url)

![Fig. 4. Left: Detected edges; right: Processed image](image-url)
After the convolution with these kernels, the image edges were found (Fig. 4, left) and added to the image processed in the previous step (Fig. 4, right).

Step A.6 «Image binarization». To the resulting image, the Sauvola binarization is applied [10]. Since the image is inhomogeneous in terms of illumination, adaptive binarization is well suited. For this image we used the Sauvola binarization. (Fig. 5).

B. Image analysis (vessel selection)

To identify the objects of interest, which are vessels, it is necessary to analyze the specificity of angiographic images: the appearance of small unrelated components due to the unevenness of the illumination of the preparation, and the appearance of a plot of nerve endings that are not vessels.

Step B.1 «Area based noise objects removal». Along with the vessels, noise (small components) is also emitted due to the heterogeneity of the illumination. These components are removed from the image by their area (Fig. 6).

Step B.2 «Compactness based noise objects removal». After the cleaning in the previous step, there are components that are not vessels in the resulting image. These components can be relatively large noise elements, and clots of nerve endings. All these elements are removed by the compactness threshold (3). Because these objects are less elongated and closer to a circle, then we can choose such a threshold without the risk of loss of blood vessels. The result in comparison with the original image is shown in Fig. 7.

\[
\text{Compactness} = \frac{\text{Perimeter}^2}{\text{Area}} \quad (3)
\]

Fig. 7. Left: Original image; right: Resulting image

Step B.3 «Skeleton construction». We construct a skeleton for the selected vessels for further measurement of their width [9]. Fig. 8 shows the skeleton of a certain region of the image and Fig. 9 show superimposed skeleton on the image with vessels.

C. Calculation of the morphometric characteristics of blood vessels

At this step, it is necessary to calculate the density, the average diameter, the standard deviation of the diameter and the total length of the vessels.

Step C.1 «Counting the density of the vessels». The ratio of the extracted pixels in step B.2, referring to the retina vasculature, to the total area of the original image is the value by which we can judge the density of the vessels.

Step C.2 «Counting the diameter of blood vessels». The diameters of the vessels on each point of the skeleton can be measured with the skeleton image. To do this, we need to perform the following operations on each pixel of the skeleton: a) construct a circle of a radius R (initial approximation is R=1); b) if a pixel of the edge of the vessel falls into the circle, then it is assumed that the width of the vessel is equal to the diameter of the circle (the pixels are converted to μm); c) if there is no edge pixel in the circle, then increase the radius by 1 and repeat from step a). By the known diameter of the vessels at each point of the skeleton, the statistical characteristics are calculated: the mean diameter and the standard deviation.

Step C.3 «Counting the length of the vasculature». At a known scale of the original image, the length of the constructed skeleton in pixels is translated into the length of the vasculature in μm.

IV. THE METHOD OF ISCHEMIA ZONE DETECTION

A general description of the algorithmic scheme of method for ischemia zones detection also includes three stages of extracting information from images.

A. Image preprocessing

Initial preprocessing of angiographic images in the method is carried out similarly to the previous method.

Step A.0 «Area of interest selection». To search for ischemia zones, the area with 17% cut edges of the image is
examined (calculated empirically). This reduction of the image is done because during the process of making fundus angiographic images, the edges of the image are poorly illuminated (due to the roundness of the eyeball) and prevent the identification of ischemia zones.

Steps A.1 – A.4 Watch section “The method of retinal vasculature detection”.

B. Image analysis (ischemia zone selection)

Step B.1 «Pixel intensity histogram building». The image from the previous steps is normalized to the segment of [0, 1]. We will divide the segment for 256 bins and build a histogram for these sub-sections (Fig. 10).

Step B.2 «Pixel intensity histogram analysis». Theoretically, considering that the ischemia zones are the darkest areas in the image, the first local maxima on the histogram will separate these areas from the other illuminated areas. The validity of this division was tested on a sample of several images. It has been experimentally proved that this extremum is indeed a threshold for finding ischemic zones. As a result of this processing of the original image, the result on Fig. 11 is obtained.

The area of the ischemia zone is found, but it is necessary to take into account the special case of the vessels overlapping on the ischemia zones on images that are in reality on another plane, but visible on the image. The following steps are aimed at obtaining a result that takes into account this feature.

Step B.3 “Vessel overlaying on the ischemia zones”. Ischemia zones are in a different plane relative to the vessels, so it is necessary to take into account (add) ischemic zones intersecting with the vessels. We fill the ischemia zone, intersecting with the vessels. In Fig. 12 shows the superposition of the vessels on ischemia zones.

Step B.4 “Expansion of the ischemia zone to the intersection with the vessels”. To fill in these areas, we will apply the dilatation operation to expand the ischemia zone to the intersection with the vessels (Fig. 13).

Step B.5 “Finding vessels in the ischemia zone”. After expansion, we obtain a region intersecting with the vessels. We find the intersections of the enlarged zone with the vessels and add it to the original image with a selected ischemia zone (Fig. 14).

Step B.6 “Filling of ischemia zones with vessels”. We obtain areas of ischemia zones filled with vessels (Fig. 15).

Step B.7 “Selected area of ischemia zone”. Applying the operation of morphological close and reconstruction to this image, we fill the areas so that there are no zones of intersection with the vessels (Fig. 16). Empirically a balance was found between the intensity of morphological operations and the correct allocation of ischemic zones.

C. Calculation of the morphometric characteristics of ischemia zones

At a known scale of the original image, the number of pixels entering the ischemia zone is translated into the area of the ischemic zone in μm².

V. Results

The computational experiments carried out showed that the developed methods for the retinal vasculature and ischemia zones detection on the human fundus angiographic images

Fig. 10. Histogram of pixel intensity

Fig. 11. Ischemia zone

Fig. 12. Ischemia zone with vessels

Fig. 13. Expanded ischemia zone

Fig. 14. Ischemia zone expanded

Fig. 15. Refined ischemia zone
provide results that allow to identify and evaluate the course of treatment of diagnosed diseases.

Discussion and interpretation of the results were carried out jointly with experts-opthalmologists. It is established that the accuracy of the method, at least, is not inferior in accuracy to the results of visual processing by experts-opthalmologists. At the same time, it is necessary to take into account the following: a) processing of a single image in automatic mode takes no more than a second, depending on the size of the processed image, and processing by an expert ophthalmologist takes at least one hour; b) the possibility of automatic processing will reduce the number of samples needed to make a diagnostic decision and choose the tactics of treatment.

Ophthalmologists have provided about 20 angiographic images. Computational experiments on the obtained data showed that the method is workable and allows to obtain results that are not worse, and are much more complete than are achieved by the ophthalmologist at visual processing. Automation also reduces the processing time of the image by 3 orders.

The main results of the work: a) a new mathematical method for identifying the vascular bed on images of fluorescein angiograms of the human fundus; b) a new mathematical method for ischemia zone detection and area measurement on human fundus angiographic images; c) software implementation of the developed methods; d) the results of computational experiments, confirming the efficiency and effectiveness of the developed method and software.

The software implementation was performed on the Python version 3 using the “scikit-image” library. The interface was also written in the Python version 3 using the Qt library. The implemented program is cross-platform and runs on any computer that supports Python 3.

The methods were transferred to the Laboratory of Fundamental Research in Ophthalmology of the Federal State Budget Scientific Institution “Scientific Research Institute of Eye Diseases” (led by AA Fedorov), were tested in it and used to automate scientific research in ophthalmology.

VI. CONCLUSIONS

Separate attempts to solve the problems in the literature are described, however, there are no standard software tools for automating the analysis of diagnostics of angiographic images in the public domain.

This work was carried out in order to provide researchers ophthalmologists and clinicians from "Scientific Research Institute of Eye Diseases” an open software system for solving problems of retinal vasculature analysis and detection of ischemic zones. This goal has been fully achieved. Software is used on a regular basis in this institute for automated scientific research in the field of ophthalmology. Paper authors plan to develop the ophthalmology complex including new tasks of automated ophthalmic image analysis.

REFERENCES

Intellectual Information Technology for Symbol Extraction from Ill-structured Graphical Documents

Nizhny Novgorod State University. N.I. Lobachevsky
Nizhny Novgorod,
Russian Federation

Abstract — the algorithms for the formation of effective, feature-resistant characteristics and decisive rules for recognizing objects represented on large-format raster images with a poorly formalized description of objects are considered. The question of the influence of the binarization method of the initial raster description on the quality of the subsequent automatic recognition of objects is considered.

Keywords — raster images, models of the description of raster data, automatic recognition, criterion of classification, binarization of raster images, filtration of the raster image.

I. INTRODUCTION.

At present, the problem of creating electronic archives of large-format graphic documents (topographic and navigational charts, design documentation, technical drawings, electrical circuits, etc.) remains urgent. In this case, the source is usually graphic images on paper, and the creation of digital documents must be provided in terms of the relevant problem area. In order to automate the technology for entering large format, complex structured graphic documents (LFGD) in recent years, various information technologies based on heuristic procedural methods and also on the basis of methods of recognition with the teacher that are effective for a limited set of objects with a tight restriction on size and orientation have been proposed. However, a significant class of LFGD is performed manually, with a poorly formalized outline of objects, arbitrary orientation, arbitrary size of symbols, the presence of a large number of interobject overlaps, junctions and intersections, forming a subclass of LFGD with a poorly formalized description of objects (PFGD). Therefore, the task of further increasing the intellectualization of the input of such documents is difficult because of the low level of recognition of objects with PFGD.

In the process of initial analysis of documents related to PFGD, the characteristic features inherent in this class of documents are revealed, namely: with PFGD-type engineering drawings, schemes, plans for floor-building buildings are documents of mass type; as a rule, this class of documents is executed on paper medium of medium and poor quality; plans and schemes lack a rigid standard and often do not follow the rules for drawing individual graphic signs; depending on the quality of the carrier with the PFGD and the correct choice of the digitization parameters, the degree of noisiness of the bitmap image of the graphic document (BIDAD) can vary within wide limits; Metric accuracy of graphic signs also varies a wide range of values.

In this regard, it is extremely difficult to create a set of standards for recognition systems with the teacher, which leads to a decrease in the quality of automatic recognition of the objects of the original image, which ultimately affects the effectiveness of the entire technological chain of creating electronic archives with PFGD.

One of the main problems that arise in solving the problem of pattern recognition and significantly affecting both the recognition efficiency, on the one hand, and its quality, on the other, is the choice of a system of effective classification characteristics.

In algorithms of recognition of the text various mathematical methods are used: structural frames, single transformations, projective histograms, geometrical moments, Fourier's descriptors, wavelet-transformation, main component, templates etc. [1]. Their efficiency is various and depends on conditions of applications. For example, at automatic recognition of noisy objects the method of the geometrical moments has proved to be from the best party, but it is reached by rather large volume of calculations made for each point of the image. For elimination of this negative property it is offered to use the method based on Fourier's descriptors [2]. Unlike a method of the geometrical moments by means of a method of descriptors of Fourier calculation is conducted only for image contour points. Sample systems [3] will transform the image of a separate symbol to a raster, normalizing at the same time his sizes, thickness of a stroke and a tilt angle; and then compare him to all templates which are available in the database of system and choose a template with the smallest quantity of the points differing on an arrangement in a raster from the entrance image. Advantage of sample systems is the small sensitivity to defects of the press, and a shortcoming – practical inability to distinguish symbols at their unfamiliar tracing. In the priznakovikh systems [4] the average image of each symbol is submitted as an object in space of signs which values are calculated at recognition of the entrance image. The received vector is compared with reference and the image corresponds to the most suitable of them. Priznakov systems adapt to tracing better, but are more sensitive to defects of the press. In structural systems [5] an object is described in terms of parts of a letter and ratios between them. For example, the system knows that the letter "P" consists of a vertical stick
and, this stick located in the top part, to the right of her, the figure similar to a semicircle. The system analyzes not the initial image of a letter, but her contour which is calculated before the letter begins to be distinguished. A lack of structural systems is slower work and high sensitivity to the defects of the image breaking the elements making a letter.

II. STATEMENT OF THE PROBLEM

Especially with PFGD it is that, in spite of the deviation from the regulatory requirements for image objects, they have some form of a stylized representation. This means that it is possible to create a set of object standards based on working with low-level models of graphic images that are maximally adapted for this class of documents, taking into account the strong dependence of the classification characteristics on the characteristics of the input data. In the future, as new documents are received, this set can be supplemented accordingly.

As initial documents 24-bit raster images of texts in English and also the tables of hydrographic shooting located strictly horizontally (fig. 1) have been taken.

![Figure 1](image-url)

Fig. 1. Examples of source images: a) a hydrographic survey plate of good quality; b) a medium-sized hydrographic survey plate; c) the scanned text

III. LOW-LEVEL STRUCTURED MODEL FOR DESCRIBING THE BIGD

Let BIGD is represented in a binary pixel format, while the brightness function of pixels has the form:

\[ f(x, y) = \begin{cases} 
255, & \text{background pixel;} \\
0, & \text{no background pixel.} 
\end{cases} \]

The non-derivative element of the model is a one-dimensional cluster of connected pixels of a given color, represented by a set of numbers \( S = (t, B, E, C) \), where: \( t \) - row number of the raster; \( B, E \) - coordinates of the beginning and the end of the connected single-color pixel region along the raster line; \( C \) - the color code of the connected pixels forming the stroke. For binary raster images (BR1), a stroke is described by a triple of type \( S = (t, B, E) \), where: \( t \) - row number of the raster; \( B, E \) are the coordinates of the beginning and end of the stroke, defined by the relations:

\[ B = x \land f(x - 1, y) \land \overline{f}(x, y), \quad E = x \land \overline{f}(x, y) \land f(x + 1, y). \]

The set of strokes for a given raster line \( t \) determines the line of strokes \( SH = \{ S_i \} (i = 1, 2, \ldots m \) , where \( S_i \) - stroke.

The description of the raster using a set of lines of strokes is unambiguous. Note that the barcode format for representing and storing raster data is 10–20 times more compact than pixel data.

Topological properties of a stroke \( S = (t, B, E) \) are defined in terms of its connection with the primes of adjacent rows. Two strokes \( S_t \) and \( S_{t+1} \) adjacent rows \( t \) and \( t + 1 \) are connected if

\[ (B_{t+1} \geq B_t) \land (B_t \leq E_t) \lor (E_{t+1} \geq B_t) \land (E_t \leq E_{t+1}) \]

A connected prime is a collection of \( H = \{ S, sw_p, sw_e \}, sw_p, sw_e \) - number of connected strokes with strokes of adjacent rows, and \( t \) - number of the current line of raster strokes. Then \( sw_p = 0 \), if in the previous line \( p = t - 1 \) there is not a single stroke for which the condition

\[ (B_p \geq B_t) \land (B_t \leq E_t) \lor (E_p \geq B_t) \land (E_p \leq E_t), \]

otherwise, \( sw_p = \mu_p/\mu_t \) - the multiplicity of the present condition, that is, about \( e \) number of connected lines of the previous line. Similarly, \( sw_e = 0 \), if in the next line \( s = t + 1 \) there is not a single stroke for which the condition is fulfilled:

\[ (B_s \geq B_t) \land (B_t \leq E_t) \lor (E_s \geq B_t) \land (E_s \leq E_t), \]

otherwise, \( sw_e = \mu_e/\mu_t \) - multiplicity of this condition, it is the number of associated strokes of the subsequent line. Here \( (B_t, E_t) \) - the coordinates of the beginning and the end of the subject under investigation to the connection of the prime, \( (B_p, E_p) \) - coordinates of the beginning and end of the current stroke of the previous adjacent line of strokes, \( (B_s, E_s) \) - coordinates of the beginning and end of the current stroke of the next adjacent line of strokes. By superposing the values of \( sw_p \) and \( sw_e \), the following classification of graphical situations can be introduced. In the current line:

- an isolated bar (Si): \( sw_p = 0 \land sw_e = 0 \);
- stroke of the beginning of the raster object (Sb): \( sw_p = 0 \land sw_e = 1 \);
- stroke of the end of the raster object (Se): \( sw_p = 1 \land sw_e = 0 \);
- the stroke of merging raster objects (Sm): \( sw_p > 1 \);
- bar splitting of raster objects (Ss): \( sw_e > 1 \);
- the stroke of merging and splitting of raster objects (Sms): \( sw_p > 1 \land sw_e > 1 \).

**Raster simple object (RSO)** – a cluster of connected strokes that does not contain graphical situations such as fusion and splitting, that is, for any cluster stroke, condition \( sw_p < 2 \land sw_e < 2 \)

**Raster composite object (RCO)** – class of connected primes, for each stroke of which the following condition is satisfied: \( sw_p > 0 \lor sw_e > 0 \), and for any two elements (dashes or pixels), there is at least one pixel trajectory connecting them, consisting of connected adjacent elements [13].
IV. METHODS AND ALGORITHMS OF THE SOLUTION OF A TASK

In the work [6], moment characteristics were introduced that have found wide application in many problems of processing digital images [7].

Moment is the characteristic of the image contour, combined (summed) with all pixels of the contour.

The moment \( m_{p,q} \) for a continuous image \( f(x,y) \) is defined as:
\[
m_{p,q} = \overline{\int \int x^p y^q f(x,y) dx dy},
\]
in the case of a discrete bitmap:
\[
m_{p,q} = \sum \sum x^p y^q f(x,y),
\]
where \( p, q \) – is the power at which the corresponding component is taken in conjunction with the other mapped ones. If the function \( f(x,y) \) is treated as a binary image, then the moment \( m_{00} \) computes the area of the black pixels of this image.

To calculate characteristics that are invariant with respect to a shift, it is necessary to center with respect to the mean values:
\[
x = -\frac{m_{10}}{m_{00}}, \quad y = -\frac{m_{01}}{m_{00}}
\]
the center of gravity of the figure.

The central moment is defined as:
\[
\mu_{p,q} = \sum \sum (x-\bar{x})^p (y-\bar{y})^q f(x,y).
\]

The central moments are expressed in terms of the usual ones, using the formula:
\[
\mu_{p,q} = \frac{p!}{n!} \sum \sum C_c^p C_s^i (\bar{x})^{p-i} (\bar{y})^{q-j} m_{ij},
\]
where
\[
C_k^h = \frac{n!}{k!(n-k)!}.
\]

With the help of central moments, characteristics invariant to the rotation of the image (moment invariants) are determined. Momentary invariants are image attributes constructed on the basis of power moments and describing the silhouette of some object, they are invariant to affine transformations of the image, they do not depend on orientation, viewing angle and other changes, they are also insensitive to various geometric transformations (shift, rotation, scaling) [8]. This determines their effectiveness when used as signs in the task of detecting and recognizing objects of unknown orientation on the image.

Having a set of characteristics, we can determine the following affine transformations of moment invariants, which are invariant under translations, rotations, and also by stretching and contraction:
\[
I^1 = \frac{\mu_{20} + \mu_{02}}{m_{00}^2}, \quad I^2 = \frac{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}{m_{00}^4},
\]
\[
I^3 = \frac{(\mu_{30} - 3\mu_{21})^2 + (3\mu_{21} - \mu_{30})^2}{m_{00}^5}.
\]

\( I^4 = \frac{(\mu_{30} + \mu_{21})^2 + (\mu_{21} + \mu_{30})^2}{m_{00}^5} \), which are used as signs of classification.

Sings of classification. For each object we will record the following characteristics:
1. The ratio of the sides of the rectangle described around the symbol with the sides parallel to the coordinate axes (aspect ratio);
2. The ratio of the area of the symbol to the area of the rectangle described around the symbol with sides parallel to the coordinate axes (fill percentage);
3. The values of the moment invariants \( I^i - I^j \);
4. The number of internal areas (IA);
5. Number of special strokes: \( Sb, Se, Sm, Ss \);
6. Number of RSOs.

The set of features considered is not final and can be extended [14].

The number of internal areas allows you to immediately divide all objects into 3 groups, i.e. objects in which: \( IA = 0 \) (K, X, S, I, 2, etc.), \( IA = 1 \) (R, O, P, 4, 6, 9, etc.) and \( IA = 2 \) (B, 8 ). We will take into account that "B" is the only one among symbols English alphabet, a numeral 8 - the only figure among the Arabs, having \( IA = 2 \).

Further object recognition can be performed within these groups.

The number of IA is a sufficiently effective sign, while the aspect ratio and the percentage of occupancy are more resistant to different distortions of the recognized data. The number of RSOs, as well as their parameters, characterize each character quite well. For example, the symbol "A" - 6 RSO, the symbol "K" - 5 RSO

Converting the original BIGD into grayscale. Given that the data source is different graphic documents, the main colors of their representation will be white (background color) and black (the color of objects). However, initial scanning of the original document in monochrome mode, can lead to significant distortions, because quite a small amount of information is represented by pure black and white colors. To minimize such distortions, you need to gradually reduce the color depth by converting the color of all pixels of the image to the desired range. Initially, we scan the LFCD with a color depth of 24 bits/pixel. This scanning mode is redundant, but allows subsequent program binarization of BIGD with minimization of possible distortions. At the first stage, we convert the full-color BIGD into shades of gray. This is done by applying to each pixel of the original BIGD a transformation of the form:

\[
Y = 0.299 R + 0.587 G + 0.144 B,
\]
in which R, G, B - red, green and blue component the colors of the original BIGD, respectively, and \( Y \) is the resulting brightness in shades of gray.
Binarization of BIGD. It is obvious that the quality of the discriminant characteristics formed, determined by the low-level structured model of the BRI description, directly depends on the quality of the preprocessing operations and the binarization of the original halftone raster image. The fact is that an unsuccessful choice of the binarization threshold leads to the appearance of various noises, which contribute significantly to a decrease in the quality of the final recognition result. Among the main obstacles can be identified presence on the objects recognized by the BRI: unauthorized tears or adhesions, the appearance of isolated voids inside objects, the presence of "snow" - isolated regions of small size of coherent non-background pixels, the ruggedness of the edges of objects.

In the course of the research, the influence of different methods of binarization of BIGD on the final result of object recognition was studied.

Binarization of BIGD by the Otsu method. In the Otsu method, a normalized image histogram is used to calculate the threshold value [9]. The histogram is constructed from the values of \( p_i = n_i / N \) where \( N \) is the total number of pixels of the image, \( n_i \) - number of pixels with brightness level \( i \) (0 ≤ \( i \) ≤ \( L \)). The histogram allows to divide all pixels of the image into two classes: "useful" and background ones, using the threshold value of the brightness level \( t \) (\( t \) is an integer value 0 ≤ \( t \) ≤ \( L \)). To each class there correspond the relative frequencies \( o_0 \) and \( o_1 \):

\[
o_0(t) = \frac{1}{L} \sum_{i=1}^{L} p_i, \quad o_1(t) = \frac{1}{L} \sum_{i=t+1}^{L} p_i = 1 - o_0(t)
\]

The average levels for each of the two image classes are:

\[
\mu_0(t) = \frac{1}{L} \sum_{i=0}^{t} o_0(t) i_p, \quad \mu_1(t) = \frac{1}{L} \sum_{i=t+1}^{L} o_1(t) i_p.
\]

The essence of Otsu's method is to expose the threshold between classes so that each of them is the most "dense". The Otsu method defines a threshold that reduces variance within a class, which is defined as the weighted sum of variances of two classes:

\[
\delta_0^2(t) = o_0(t) \delta_1^2(t) + o_1(t) \delta_2^2(t).
\]

Next, the maximum value of the image quality estimation is calculated into two parts:

\[
\eta(t) = \max \left( \frac{\sigma_y^2(t)}{\sigma_G(t)} \right) \sigma_y^2(t) = o_0(t) o_1(t) (\mu_1(t) - \mu_0(t))^2.
\]

The threshold brightness values of the pixels were estimated from the histogram.

In Fig. 2 shows a fragment of the image of the hydrographic survey plate, processed by the method of binarization with double restriction.

Comparing results of binarization (Fig 2, right and Fig 3) shows that in this class of documents the method of binarization with double constraint eliminates deficiencies of binarization by the Otsu method.

Correction of a tilt angle of a raster. As strokes are allocated along horizontal or vertical lines of a raster, the values of signs received with their use directly depend on orientation of the image.

It should be noted that the values of discriminant signs determined by the shaped description (quantity of strokes of merge/splitting, a relative positioning of special strokes, the aspect relation) are very sensitive to a tilt angle of the recognizable raster. It is experimentally established that at an inclination of the recognizable raster on a corner, bigger ±5 to the horizon, the quality of recognition significantly worsens.

The result of binarization with double restriction.

In Fig. 2 on the left shows a fragment of a successfully binarized image of a hydrographic survey plate using the Otsu method.

The result of processing has an obvious drawback: when processing objects that have different dimensions, but close colors, as a result of the binarization Otsu are obtained dark spots (Fig. 2, right).

Binarization of BIGD with double restriction. In this case, the pixels that enter the range \([t_1, t_2] \) - become non-background – \( f(x,y)=0 \), which do not enter - background – \( f(x,y)=255 \) [10].

![Image](image.png)

**Fig. 2.** left: a good result of Otsu's binarization; right: unsuccessful choice of Otsu binarization threshold.

The threshold brightness values of the pixels were estimated from the histogram.

In Fig. 3 shows a fragment of the image of the hydrographic survey plate, processed by the method of binarization with double restriction.

Comparing results of binarization (Fig 2, right and Fig 3) shows that in this class of documents the method of binarization with double constraint eliminates deficiencies of binarization by the Otsu method.

**Correction of a tilt angle of a raster.** As strokes are allocated along horizontal or vertical lines of a raster, the values of signs received with their use directly depend on orientation of the image.

It should be noted that the values of discriminant signs determined by the shaped description (quantity of strokes of merge/splitting, a relative positioning of special strokes, the aspect relation) are very sensitive to a tilt angle of the recognizable raster. It is experimentally established that at an inclination of the recognizable raster on a corner, bigger ±5 to the horizon, the quality of recognition significantly worsens.

In this regard there is a need of correction of a tilt angle to the horizon of the recognized text.

We will define the main requirements to an algorithm of definition of a tilt angle of the text to the horizon:

- speed (operating time of an algorithm shouldn't depend on content of the image);
- accuracy (small size of a systematic mistake);
- the tilt angle can be defined as locally – for some part of the image, and for all document in general;

In the review [11] several various methods of definition of a tilt angle of the document are offered. The developed algorithm, combines some of them and is adapted to a shaped format that reduces the number of necessary calculations. For text tilt angle assessment in work the device of the vertical
projective histogram [12] representing the one-dimensional massif of the dimension equal to number of lines of a raster where each his cell keeps the number not of background pixels in the corresponding line of the image is used.

Let there be a binary raster image which lines are strictly horizontal (θ=0), W x H pixels in size where W and H – respectively the pixel width and height of the image. Let \( W(\theta) \), \( H(\theta) \) – respectively the pixel width and height of the same image, which lines located at an angle \( \theta \) to the horizon. Then

\[
S(\theta) = \sum_{j=1}^{W(\theta)} P(i, j)
\]

– the total number not of background pixels in a line of a raster with number i at turn of the initial image on a corner \( \theta \), and

\[
S(\theta) = \sum_{j=1}^{W(\theta)} S^2(i, \theta)
\]

– total function of an angle of rotation \( \theta \).

Example of the \( S(\theta) \) function, rated on the maximum value, received on the real graphic document turned on a corner about 10 (fig. 4).

![S(\theta)](image)

Fig. 4. The \( S(Q) \) function, rated on the maximum value, received on the real graphic document

Stage of image preparation. At this stage, images are divided into connected regions not background pixels assigned with the unique label and described rectangular region around it. Note that this operation is the most time-consuming. During the execution of this stage, the object filtering of the BRI is performed, the BRI is converted to a barcode format, and the internal areas within the component are selected.

Algorithm of object filtering of the source binary image. Note that, in general, the characteristics determined by the low-level model of BRI are not sufficiently stable. Various kinds of interference on the BRI lead to some changes in the dashed description, as a result of which additional splitting strokes, mergers, RSOs and internal areas can be highlighted. Therefore, effective use of the bar-graph description for automatic object recognition requires pre-filtration of the BRI.

The input of the algorithm is supplied by the BRI with recognizable objects. Input parameters are the color code of the noise to be removed and the dimensions of the rectangle described around the interference with sides parallel to the coordinate axes, as well as the percentage of its occupancy by the pixels of the interference color. The work of the algorithm is based on the allocation of the connected components of the original BRI, consisting of pixels with a color equal to the specified one, followed by the removal (repainting into an alternative color) of those connected regions that satisfy the interference parameters in terms of their geometric characteristics. An important advantage of this algorithm is the fact that the specified filtering does not introduce any distortions into those connected areas that are not recognized by the algorithm as noise.

Algorithm for automatic character recognition. The input of the algorithm is the vector of attributes of the connected object. At the output, the recognized symbol of the alphabet of the reference base or an empty symbol. The algorithm introduced the notion of a penalty - a positive value proportional to the ratio of the distance to the nearest interval boundary to the length of this interval. For characteristics formed by a dashed description: if the value of the characteristic is less than the value of the lower bound of the interval, then immediately a penalty is imposed that is greater than the threshold value and the class is removed from further consideration.

Initially, we form an array in size by the number of classes. In the cycle by the tag number:

1. For each class, a penalty is added to the corresponding array cell, if the value of the characteristic is not within the allowed range for the class.
2. If the penalty for an object for a class exceeds the threshold, then the class is excluded from further consideration.
3. If the examination was only one class, the corresponding symbol is returned as the result of recognition.
4. If all classes have been eliminated, the null character is returned as a result of the algorithm.

If, after completion of the cycle there are a few classes, fives are not exceeded threshold, as a result returns the character corresponding to the class with a minimum fine.

Algorithm for automatic assembly inscriptions. The input to the algorithm enters a set of descriptions of the recognized objects. At the exit description of structure formed within Labeled output thematic database. Assembling is performed by using clustering techniques. Another object is added to the current cluster if it misses the \( e \)-neighborhood about the center of gravity of the cluster core and the angle of inclination of the vector connecting the center of gravity and cluster nucleus candidate object to add to the cluster from the horizontal does not exceed a predetermined value.

V. RESULTS

According to the training text sample was formed by 5 standards per character. The analysis concluded that the values of \( I^1 \) and \( I^2 \) is sufficient to distinguish classes among all groups of similar values \( I^1 \) and \( I^2 \) are necessary for fine classification within the group. In general, values of moment invariants are stable enough and therefore recognition begins with them. These symptoms are more likely to rule out a
variety of classes, not related to the object under consideration.

In practice, after the use of the torque indication remains 2 - 5 possible classes, which may belong to a recognizable object. For a final decision on referring to a specific object class uses the attributes derived from the description of the bar. This takes into account the experimentally ascertained fact that the individual features derived for bar-description, are sufficiently resistant. For example, the character "M" may never have less than three Se, two Sb and Ss, one Sm and four RSO. This is true for any of recognizable objects, provided that they are presented by consistent raster objects. This feature allows to increase the time efficiency of recognition procedures.

The use of the algorithm for character recognition of real PFGD made it possible to achieve quality recognition at the level of 93 - 96%.

References

Workshop: Lognormality Principle and its Applications

Session 1: 7 papers
Thursday May 17, 2018: 10:00 - 12:30

Roberto Parisi, Antonio Parziale and Angelo Marcelli

*Some observations on lognormality and motor control in handwriting*

Simon Pierre Boyogueno Bidias, Jean Pierre David, Yvon Savaria and Réjean Plamondon

*On the use of Interval Arithmetic to Bound Delta Lognormal Rapid Human Movements Models*

Saad Chidami, Mylène Archambault-Caron and Réjean Plamondon

*The Delta-Lognormal model in 2.5D*

Roman Schindler, Manuel Bouillon, Réjean Plamondon and Andreas Fischer

*Extending the Sigma-Lognormal Model of the Kinematic Theory to Three Dimensions*

Nadir Faci, Simon Pierre Boyogueno Bidias, Réjean Plamondon and Nicolas Bergeron

*A New Experimental Set-up To Run Neuromuscular Tests*

Luis A. Leiva, Daniel Martin-Albo and Réjean Plamondon

*Gesture Synthesis for Human-Computer Interaction*

Daniel Berio, Frederic Fol Leymarie and Réjean Plamondon

*Kinematic Reconstruction of Calligraphic Traces from Shape Features*

Session 2: 5 papers
Thursday May 17, 2018: 1:30 PM - 3:10 PM

Miguel A Ferrer, Moises Diaz, Cristina Carmona-Duarte and Rejean Plamondon

Improving on-line signature skillfulness

Ruben Vera-Rodriguez, Ruben Tolosana, Javier Hernandez-Ortega, Aythami Morales, Julian Fierrez and Javier Ortega-Garcia

*Modeling the Complexity of Biomechanical Tasks using the Lognormality Principle: Applications to Signature Recognition and Touch-screen Children Detection*

Rémi Céline, Jean Vaillant, Nagau Jimmy and Réjean Plamondon

*Can the Sigma-lognormal Modeling Help to Monitor Child Graphomotor Skill Progress?*
Patricia Laniel, Nadir Faci, Réjean Plamondon, Miriam H. Beauchamp and Bruno Gauthier

Kinematic Analysis of Fast Pen Strokes in Children with ADHD using the Sigma-lognormal Model

Nadir Faci, Naddley Désiré, Miriam H. Beauchamp, Isabelle Gagnon and Réjean Plamondon

Lognormality in Children With Mild Traumatic Brain Injury: A Preliminary Pilot Study

Cristina Carmona-Duarte, Miguel Ferrer, Pedro Gómez-Vilda, Arend. W. A. Van Gemmert and Réjean Plamondon

A common framework to evaluate Parkinson’s disease in voice and handwriting

Alexandra Nadeau, Karina Lebel, Joan Carbo, Arnaud Bore, Ovidiu Lungu, Julien Doyon, Hung Nguyen, Christian Duval, Patrick Boissy, Nicole Vincent, Zoi Kapoula and Réjean Plamondon

Exploiting the Lognormality Principle: Three Ongoing Projects
Some observations on lognormality and motor control in handwriting

Roberto Parisi  
DIEM- Natural Computation Lab  
University of Salerno  
Fisciano (SA), ITALY  
r.parisi12@studenti.unisa.it

Antonio Parziale  
DIEM- Natural Computation Lab  
University of Salerno  
Fisciano (SA), ITALY  
anparziale@unisa.it

Angelo Marcelli  
DIEM- Natural Computation Lab  
University of Salerno  
Fisciano (SA), ITALY  
amarcelli@unisa.it

Abstract— Lognormality has proven to be an effective way for handwriting modeling. It assumes that handwriting is a time superimposition of a sequence of commands issued by the central nervous system, each command producing a stroke, i.e. a movement with a lognormal velocity profile. Motor control theories, however, suggest that handwriting movements result from both central and local control, thus assuming that some movements of the sequence may not be the results of an explicit command issued by the central nervous system. In the light of those observations, we present an algorithm for segmenting handwriting movements into strokes, each of which corresponds to a command issued by the central nervous system, while disregarding those that may depend on local control. Experiments on handwriting samples show that the proposed algorithm detects the same number of strokes across multiple executions of an handwriting task by the same subject, and this set of strokes provides also a good reconstruction of the movement.

Keywords— handwriting generation model, handwriting learning, handwriting production, action plan, stroke segmentation.

I. INTRODUCTION

Lognormal distributions have proven to be very effective in describing natural and cultural phenomena resulting from the accumulation of many small percentage changes. Lognormal distributions describe biological and neural systems [1-4] as well as social and economic behaviors [5, 6], just to mention a few.

In case of rapid human movements, such as reaching or pointing, it has been observed that their velocity profiles also exhibit a lognormal distribution, and the investigation to model the source of this behaviour has lead to the development of the kinematic theory of rapid human movement [7-9]. According to such a theory, these movements, called strokes, result by issuing an impulse, modelling the central nervous system, that goes thru a network of interconnected modules, modelling the muscular system. The observed lognormal distribution of the velocity profile, thus, results by the accumulation of the responses of the basic modules according to the network architecture.

In case of complex human movements, such as handwriting, the theory mentioned before suggests that they can be thought as the time superimposition of strokes, each of which results from a command generated by the central nervous system for reaching a target point and exhibits a lognormal velocity profile [10]. To support this view, several algorithms have been proposed to perform stroke segmentation, i.e. to identify within a handwriting movement its stroke and estimate the values of the parameter describing each of them, as prescribed by the theory [11, 12]. Along the years, many experiments in many different domains have shown that the proposed algorithms exhibits very good performance in extracting the strokes embedded into the movements and estimating their parameters in such a way that the reconstructed velocity profile resembles very closely the actual one [13, 14].

Motor learning and execution studies aimed at modeling the human neuromuscular system, on the other hand, have shown that the representation of complex movements such as handwriting is stored in two different forms. The former, called effector-independent, is represented by the sequence of target points, each representing the position to which each elementary movement is aimed at reaching. The latter, called effector-dependent, is represented by the commands to the specific neuromuscular system that is recruited for executing the movement [15, 16]. More recently, we have suggested that the implementation of a movement through the interaction between the central nervous system and the musculoskeletal system can be interpreted as the realization of a motor plan stored in the brain [17]. The idea is that the practice of a certain movement over time allows to create a compact representation of a complex movement that, in the final stages of learning, is stored as a succession of elementary motor commands describing the motor plan. Accordingly, after a movement has been learned, i.e. when the subject executing the movement is no longer conscious of the elementary movements it is composed of, the variability observed in repeated executions may be ascribed to the neuromuscular system executing the movement. Further investigations into movement execution have shown that the actual movements result from the interaction between the central nervous system, the spinal cord, the muscles and the proprioceptive receptors [18]. In a nutshell, to initiate the movement, the central nervous system sends commands to recruit the muscles and to set the forces they have to exert on the bones they are connected to, while, during
execution, the spinal cord modulates such command depending on the information received by the proprioceptive receptors in order to keep the execution as close as possible to the learned one. The effects of those modulations are therefore the source of the observed variability, and they should not be considered as the results of commands stored in the motor plan.

Following these observations, we propose an algorithm that aims at segmenting handwriting movements into the sequence of strokes that encode the motor plan, i.e. those that most likely correspond to the actual commands issued by the central nervous system for handwriting execution. By definition, such motor plan is independent from the variability affecting different executions by a subject. Therefore, the desired representation should be stable with respect to the number of elementary movements that constitute the description of each handwritten sample.

In the next section we describe the algorithm building blocks and provide the rationale and the structure of their implementation. Section 3 reports the results of a set of experiments to evaluate the performance of the proposed algorithm in reconstructing the velocity profiles of repeated executions of a handwriting task. In addition to the performance indices that are commonly used in the literature, we introduce and motivate the use of the stability of the number of strokes, which has been neglected in other studies. Eventually, in the conclusion we summarize the experimental findings and discuss the inside the proposed stability measure offers for handwriting modeling and its application.

II. MOTION PLAN EXTRACTOR

The fine motor control required by handwriting is achieved by learning how to produce the desired movement under different conditions, and this learning is driven by the minimization of the metabolic energy. As mentioned in the Introduction, during handwriting the spinal cord can generate movements that are needed in order to keep the ongoing execution as close as possible to the learned one. Therefore, they are expected to result in small changes on the dynamics and kinematics of the ongoing movement, and the actual strategy adopted for producing the desired changes is the one that requires the smallest amount of metabolic energy to be executed.

Among the lognormal models that have been proposed in the literature, the Sigma-Lognormal Model [11] is particularly suitable for describing what can be considered the motor plan, i.e. a time superimposition of strokes aimed at reaching a suitable for describing what can be considered the motor plan, or whose duration s are shorter than the durations of the movements generated by the actual commands of the motor plan. We implement those criteria by comparing the peak velocities generated by the spinal cord should correspond to either velocity peaks whose amplitudes are much smaller than the amplitudes of the velocity peaks corresponding to the movements generated by the actual commands of the motor plan, or whose durations are shorter than the durations of the movements generated by the actual commands of the motor plan. We implement those criteria by comparing the peak amplitudes and their durations with two thresholds, denoted by \( T_{vel} \) and \( T_{short} \), respectively, and ignoring the peaks of the velocity profile that meets the criteria above in the next steps of the algorithm.

The values of the thresholds are expressed as a percentage of the maximum peak amplitude within the whole signal in case of \( T_{vel} \), and as percentage of the average duration of peaks for \( T_{short} \), being the duration of a peak measured as the time difference between the minima of the velocity profile before and after the peak.

A. Stroke identification

At this stage, the handwriting sample velocity signal is analyzed searching for peaks, each of which corresponds to a basic movement, which will be described using the Sigma-Lognormal model. According to our model, movements generated by the spinal cord should correspond to either velocity peaks whose amplitudes are much smaller than the amplitudes of the velocity peaks corresponding to the movements generated by the actual commands of the motor plan, or whose durations are shorter than the durations of the movements generated by the actual commands of the motor plan. We implement those criteria by comparing the peak amplitudes and their durations with two thresholds, denoted by \( T_{vel} \) and \( T_{short} \), respectively, and ignoring the peaks of the velocity profile that meets the criteria above in the next steps of the algorithm.

B. Parameter estimation

In our implementation of this stage we adopted a simplified version of the procedure used by RX0 extractor to obtain the lognormal parameters that describe each elementary movement velocity profile. In contrast with RX0, that attempts to reduce the complexity of the search by bounding the values of the lognormal parameters depending on the profile of the peak, we
adopt a brute force approach to estimate the optimal lognormal parameters \([t_0, D, \mu, \sigma]\): for each \(\sigma\) value within the interval \([0, 2]\) we search for the best fitting lognormal profile with respect to the sample velocity profile. The best fitting strokes are those that locally optimize the SNR value between the original velocity signal and the actual reconstruction within the time interval corresponding to the peak duration, as previously defined.

A lognormal signal is reconstructed by using the equations (20a)-(20e) and (22)-(24) from [11], that are reported below for the convenience of the reader:

\[
a_{\text{flex-left}} = 1.5\sigma^2 + \sigma \sqrt{0.25\sigma^2 + 1} \\
a_{\text{flex-right}} = 1.5\sigma^2 - \sigma \sqrt{0.25\sigma^2 + 1} \\
D = \sqrt{2\pi v_{\text{peak}} e^\mu \sigma e^{-\frac{a_{\text{flex-right}}}{\sigma^2}}} a_{\text{flex-right}} \\
t_0 = t_{\text{min-left}} - e^\mu e^{-a_{\text{flex-right}}} \\
\mu = \ln \left( e^{-a_{\text{flex-right}}/e^{a_{\text{flex-left}}}} \right)
\]

(5)

C. Local optimization

Local optimization is performed after each lognormal parameter estimation to improve reconstruction quality. A non-linear least-square optimization method is applied first to \([t_0, D, \mu, \sigma]\) to improve velocity reconstruction and then to \(\theta_x\) and \(\theta_y\), to improve x-y signal reconstruction. In contrast, RX0 local optimization applies the non-linear least-square optimization method to \(\mu, \sigma, \theta_x\) and \(\theta_y\), first and then to \(t_0\) and \(D\), thus focusing mainly on improving the reconstruction of the velocity profile. Our choice follows from the assumption that in the motor plan the amount of time superimposition between successive strokes is learned at the same time as the other parameters of the movements are learned, and eventually the brain learns how to configure the musculoskeletal system for achieving the desired movement. On the contrary, the local optimization performed by RX0 assumes that the learning of all the parameters of the basic movements are learned first, and then their time superimposition is learned.

At last, but not least, after each lognormal has been fitted within the sample, the portion of the velocity profile fitted by that lognormal is removed from the velocity signal but the corresponding portion of the trajectory is not, as instead in the case of RX0, and the procedure is iterated until no more lognormal can be fitted. Our choice follows from the same observations that motivated our implementation of the local optimization stage. Angle estimation and global optimization phases, eventually, have the same implementation as in the RX0 algorithm.

III. PERFORMANCE EVALUATION

As mentioned in the introduction, the proposed algorithm aims at extracting from the handwriting the movements that correspond to the execution of the previously learned motor program stored in the brain, discriminating them from those arising by commands sent by the spinal cord in response to proprioceptive feedbacks. As at the end of the learning the effector-independent representation of motor program is the sequence of target points, and therefore the number of strokes between them, each execution of that motor program should result from the composition of that number of strokes. As the ground truth, i.e. the number of strokes actually encoded into the motor program, is not available, we consider the stability of the number of strokes embedded into the motor plan to evaluate the performance of the proposed method in providing a plausible motor plan. The similarity between the dynamics and the kinematics of movements obtained by executing the motor plan provided by the MPE algorithm and the original one is adopted for evaluating to plausibility of the estimated motor plan parameters as the commands for generating the actual movement.

The experiments involved 22 subjects, 14 males and 8 females, whose age ranges in the interval 18-30 years, with a mean value of 23.45 and a standard deviation of 3.07. All of them volunteered to take part in the experiment and expressed their formal consent.

During the experiment, each subject was requested to reproduce 10 times the handwriting pattern in Fig. 2. Such a pattern, or very similar ones, have been used in many experiments on handwriting generation modeling as well as for validating neuro-computational modeling of handwriting [19-23], because it is reasonable to believe that its motor plan has been learned by the subjects involved in the experiments, independently of their individual characteristics, being at the same time fairly complex to allow a quantitative evaluation of the estimated motor program parameters.

The handwriting samples were collected by using an ink-and-paper WACOM Intuos 2 digitizing tablet with a 100 Hz sampling rate to record the handwriting movements. We adopted an ink-and-paper digitizing tablet as we aimed at extracting the strokes corresponding to the motor program of the subject, that most likely were learned under the same condition, so as to avoid as much as possible the influence of the spinal cord during the movement execution due to unexpected proprioceptive feedbacks that may arise in different writing condition, as it would have been the case by using a stylus-and-screen digitizing tablet.

Figure 3 shows, for every subject and for every sample of that subject, the number of stroke found by our algorithm. For the sake of comparison, we also report the numbers provided by the RX0 algorithm. Over the entire data set, the mean number of strokes is 8.88 (standard deviation 1.44) and 14.25 (standard deviation 2.80) for MPE and RX0, respectively.
Figure 4 reports, for every subject and for every sample of that subject, the SNR computed by comparing the velocity profiles reconstructed by MPE and the original ones. As in the previous case, the figure reports also the values in case of the signals reconstructed by using the RX0 algorithm. Over the entire data set, the mean SNR value is 16.58 (standard deviation 5.50) and 19.60 (standard deviation 1.48) for MPE and RX0, respectively.

Figure 5 reports, for each subject involved in the experimentation, the mean RMSE and the standard error of the mean computed by comparing the xy-trajectories reconstructed by MPE or RX0, and the original ones. Over the entire data set, the mean RMSE is 0.18 (standard deviation 0.29) and 0.07 (standard deviation 0.03) for MPE and RX0, respectively.

The results reported in the figures were obtained by setting $T_{vel}$ to 10% of the maximum of velocity profiles, while $T_{short}$ was set to 10% of the average durations of the peaks.

IV. DISCUSSION AND CONCLUSIONS

The research reported in this paper was motivated by our interest in developing handwriting analysis tools for early diagnosis of neurodegenerative diseases. As most of the diseases we are interested in produces difficulties in complex motor task learning and execution, it is of paramount importance to be able to understand in which way they affect the motor plan, and how the difficulties they produce progress along time. Effective modeling of such difficulties by computational tools that incorporate the features of the neuromuscular system may lead to better understanding of those pathologies, as well as provide insight for early diagnosis.

In this framework, we have proposed a stroke segmentation algorithm that attempts to discriminate between elementary movements that results from the execution of the stored motor program and other movements that are generated by the spinal cord in reaction to proprioceptive feedbacks, as suggested by motor control studies on learning and execution of voluntary movements.

The proposed algorithm builds upon another algorithm proposed in the literature within the framework of the kinematic theory of rapid human movements for decomposing the handwriting movements into elementary movements, but, in contrast with such a theory, assumes that some of the elementary movements are originated by the spinal cord and therefore do not correspond to a command embedded into the representation of the motor plan stored in the brain.

The results of the experiment illustrated in figure 3 show that the MPE algorithm produces very promising results, as it provides the same number of strokes for the large majority of, if not for all, the repetitions of the task by the subject, supporting the claim that there is one motor plan controlling the execution of the movement. The comparison of these
numbers with those provided by the RX0 algorithm, moreover, supports our claim that not every stroke in the actual movement correspond to the execution of a command embedded into the motor program.

Eventually, the results reported in figures 4 and 5 show that in the large majority of the cases, both the dynamics and the kinematics of the movement produced by executing the motor plan provided by the MPE algorithm resemble very closely those of the original movement. The comparison of the overall performance on the entire data set between MPE and RX0 also supports our claim that the modulation of the command performed by the spinal cord to keep the ongoing movement as close as possible to the learned one produces small changes with respect to the learned one, as removing those changes from the actual movement does not deteriorate significantly the characteristics of the reconstructed movement with respect to the one reconstructed by including all of them.

The results of the experiments show that, even in the case when the same number of strokes is extracted across many handwriting samples of the same subject, the values of the parameters of the Sigma-Lognormal strokes are not the same across the repetitions, as one would expect if they were multiple executions of the motor plan learned by the subject. Therefore, our future investigations will aimed at exploiting the handwriting generation model we have described in the Introduction to obtain the set of parameter values representing the actual commands embedded into the motor plan.

ACKNOWLEDGMENT

The work reported in this paper was partially supported by the grant “Bando PRIN 2015 - Progetto HAND” from the Italian Ministero dell’Istruzione, dell’Università e della Ricerca.

REFERENCES


Fig. 4 SNR values obtained by comparing the reconstructed velocity profile and the original ones. Note that user S01 wrote 9 patterns instead of 10.
Fig. 5 Mean RMSE value with the standard error of the mean obtained by comparing the reconstructed trajectories and the original ones.
On the use of Interval Arithmetic to Bound Delta-Lognormal Rapid Human Movements Models

Simon Pierre
Boygoueno Bidias
Electrical engineering
Polytechnique Montréal
Montréal, Canada
simon-pierre.boyogueno-bidias@polymtl.ca

Jean-Pierre David
Electrical engineering
Polytechnique Montréal
Montréal, Canada
JPDavid@polymtl.ca

Yvon Savaria
Electrical engineering
Polytechnique Montréal
Montréal, Canada
yvon.savaria@polymtl.ca

Réjean Plamondon
Electrical engineering
Polytechnique Montréal
Montréal, Canada
rejean.plamondon@polymtl.ca

Abstract. This paper proposes an algorithm based on interval arithmetic to construct envelope models that contain all possible parameter values of a delta-lognormal function that can match an observed human movement. In the context of the Kinematic Theory of rapid human movements, these parameters represent the basic timing properties of a neuromuscular system, as well as the control strategies used to produce a rapid human movement. The basic idea behind the proposed algorithm is to exploit the isotonicity and monotonicity over an interval to sharply bound the range of the delta-lognormal function, within a finite space. The new algorithm is described, tested (under various handwritten strokes) and compared with another algorithm previously developed for the same purpose. The numerical results show that the proposed algorithm gives better results in terms of speed and accuracy and provides new tools for the basic study of a single stroke as a primitive for understanding rapid human movements, according to the Kinematic Theory.

Keywords—bounding model, monotonicity, isotonicity, interval arithmetic, delta-lognormal, kinematic theory.

I. INTRODUCTION:

Human movements can be very complex, and their study in terms of motor control has been intensively used in many fields of research. In neurosciences, handwritten strokes constitute a specific class of rapid human movements, notably used to study neurodegenerative processes such as those involved in Parkinson and Alzheimer’s diseases [1], [2]. They have also been used in cybersecurity applications [3], as well as for the rehabilitation of patients having suffered from cerebrovascular accidents [4], [5], [6]. In this context, the need to study the elementary properties of a single stroke for a realistic analysis becomes a key step to understand how the motor control system accomplishes complex movements. The Kinematic Theory is commonly used to model rapid human movements [7], [8], [9]. It present handwritten strokes as primitives that contain information on both the motor control and the neuromuscular system. The study of such strokes relies on the extraction of parameters that characterize their velocity profile [10], [8].

To meet this need, three deterministic [11], [12], [13] algorithms have been developed in our laboratory in order to extract the lognormal parameters derived from handwriting velocity profiles. These algorithms have successfully led, for example, to applications for recognition of signatures and handwriting. Although the parameter extractor developed in [8], [9], [10] show good data fitting, none of these prior work can guarantee the optimality of the solution obtained. Such a requirement is not always critical for some applications, as excellent curve fitting is sufficient for many pattern recognition tasks. However, globally optimal solutions are important in biomedical applications where a confidence interval on the value of an extracted parameter is necessary in developing a benchmark, or a diagnostic. To address this need, a first branch and bound algorithm was developed in [14].

This algorithm is based on building a tree where each node represents a possible solution of the problem considered. The nodes are delta-lognormal envelopes, built by studying the variations of the delta-lognormal function, according to its seven parameters. Each subset or child node is created by subdividing the parent node into three children. The dimension according to which a parent node is subdivided is selected depending on whether the variables that minimize the area of the parent delta-lognormal envelope. The tree thus constructed is traversed in a way that we prune non-promising solutions base on a monotonicity test. In summary, the branch and bound algorithm proposed in [14] is based on a delta-lognormal envelope that bounds all the possible delta-lognormal parameters within a subspace, and then computes a bound on the error between the signal and the nearest envelope. This method, despite the fact that it clearly circumscribed the challenge, suffers for lower bound problems (in some cases the computation of the lower bound of the envelope does not hold) and requires very long processing times.

The goal of this paper is to propose a new algorithm to relieve the bounding problem of the extractor proposed in [14]. The proposed method is used to bound the delta-lognormal parameters representing velocity profile data. Its effectiveness is demonstrated with ideal and experimental velocity profiles.

The rest of the paper is organized as follows. Section 1 reviews the delta-lognormal equation and the modelling tools that will be used to bound the objective function. Section 2 introduces some theoretical aspects of interval arithmetic and proposes an algorithm for computing enclosures of the bound of the objective function. Section 3 presents numerical results and performance comparisons with [14]. Section 4 discusses the importance of the proposed algorithm, and finally Section 5...
proposes possible future research and summarizes the main contributions of this paper.

II. THE DELTA-LOGNORMAL MODEL

The Kinematic Theory of rapid human movements, from which the delta-lognormal model is derived, considers the trajectory of a pen tip as a single stroke having a delta-lognormal velocity profile.

\[
\Delta \Lambda(t; p) = D_1 \Lambda(t; t_0, \mu_1, \sigma_1) - D_2 \Lambda(t; t_0, \mu_2, \sigma_2)
\]

Where

\[
\Lambda(t; t_0, \mu, \sigma) = \begin{cases} 
\frac{1}{\sigma \sqrt{2\pi(t-t_0)}} \exp \left( \frac{-(\ln(t-t_0)-\mu)^2}{2\sigma^2} \right), & \text{for } t < t_0, \\
0, & \text{elsewhere.}
\end{cases}
\]

with \( \sigma > 0 ; \mu, t_0 \in \mathbb{R} \)

is the lognormal probability density function. In this theory, a stroke is produced by a synergy of two neuromuscular systems: the agonist, acting in the direction of the movement and the antagonist, acting in the opposite direction. The agonist and antagonist are lognormal convolved with their corresponding input commands, represented by the subscript 1 and 2 in equation (1) respectively. The subtraction of the two systems responses produces the velocity profile \( \Delta \Lambda(t; p) \) described by the seven parameters \( p \) shown in equation (3).

\[
p = [t_0, D_1, \mu_1, \sigma_1, D_2, \mu_2, \sigma_2]
\]

In (3), \( t_0 \) represents the system activation time, \( \mu_1, \sigma_1, \mu_2, \sigma_2 \) correspond to the timing properties (on a logarithmic scale) of the two neuromuscular commands, in reaction to their input amplitudes \( D_1 > 0 \) and \( D_2 > 0 \) respectively.

To bound the model parameters from velocity data, we need to find the bounding box \( P \) that contains all the possible seven parameters \( p \), which fit the original velocity movement (collected from synthetic human movements) most accurately. This problem can be achieved by building an envelope to the nonlinear least square function \( f \) (4) that maximizes the signal to noise ratio SNR (5).

\[
f = \int (v(t) - \Delta \Lambda(t; p))^2 dt
\]

\[
SNR = 10 \log \left( \frac{\int v^2(t) dt}{\int f} \right)
\]

III. INTERVAL ARITHMETIC:

Before formulating the proposed algorithm, some definitions and notations must be established. Consider the following problem:

Bound \( f(x) \) subject to \( x \in X \), where \( f \) is an \( n \) dimensional continuously differentiable function \( \mathbb{R}^n \rightarrow \mathbb{R} \) and \( X \subseteq \mathbb{R}^n \) is an \( n \) dimensional interval vector.

We denote by \( I = [a, b] = \{x \in \mathbb{R} : a \leq x \leq b\} \) the set of real numbers and one dimensional closed intervals. In this paper, an interval will be denoted with a variable in upper case and the set of values contained in the interval, in lower case. Thus \( X = [\underline{X}, \overline{X}] \) is an interval variable where \( \underline{X} \) and \( \overline{X} \) represent the left and right end points of \( X \) respectively. The width of \( X \) is defined and denoted by \( w(X) = \overline{X} - \underline{X} \) and the midpoint is given by \( m(X) = \frac{1}{2}(\overline{X} + \underline{X}) \).

We denote by \( X^I = (X_1, X_2, ..., X_n) \) the set of \( n \) dimensional interval vectors or box such that:

\[
X^I = (X_1, X_2, ..., X_n) = ([\underline{X}_1, \overline{X}_1], [\underline{X}_2, \overline{X}_2], ..., [\underline{X}_n, \overline{X}_n]).
\]

The width and the midpoint of an interval vector are defined as:

\[
w(X^I) = \max_i w(X^I_i)
\]

and

\[
m(X^I) = (m(X_1), m(X_2), ..., m(X_n)).
\]

We call \( F \) an inclusion function of \( f : \mathbb{R}^n \rightarrow \mathbb{R} \), the interval extension of \( f \), such that for \( x \in X \Rightarrow f(x) \subseteq F(X) \). \( F(X) \) is the real range of \( f \) on \( X \) and \( f(X) \subseteq F(X) \). Now that we have these definitions and notations, it is possible to use interval arithmetic [15], [16] to bound a function defined by a mathematical expression such as the delta-lognormal function.

A. Interval arithmetic operations:

In this section, we summarize some principles of interval arithmetic needed to derive the bounding operation of the objective function stated in (4). For more details, the reader can consult [15], [16].

Definition 1: the natural interval extension of a given function \( f(x_1,x_2,...,x_n) \) of \( n \) variables is given by the interval function \( F(X_1,X_2,...,X_n) \), which is obtained by replacing the real variable \( x \) with the corresponding interval variable \( X \).

Example 1: The natural interval extension of

\[
f(x, y) = (1 - x)^2 + 100(y - x^2)^2
\]

\[
g(w, z) = \exp \left( \frac{w + z}{z^2 + w} \right)
\]

are:

\[
F(X,Y) = (1 - X)^2 + 100(Y - X^2)^2
\]

\[
G(W, Z) = \exp \left( \frac{W + Z}{Z^2 + W} \right)
\]

Where \( X, Y, W \) and \( Z \) are intervals. If we assign for example \( X = [-1, 2], Y = [-1, 3], W = [0, 2], Z = [-1, 1] \) we have:

\[
F([-1, 2], [-1, 3]) = (1 - [-1, 2])^2 + 100([-1, 3] - [-1, 2]^2) = [0, 2504]
\]

\[
G([0, 2], [-1, 1]) = \exp \left( \frac{[-1, 3]}{[0, 1] \cup [-1, 1]} \right)
\]

\[
= \exp([-1, 3]) = [\exp(-1), \exp(3)]
\]
It is of interest that, in these examples:

- In (8), if we take any interval, say \( X' = [-1; 1] \subset X \), the value of \( F(X', Y) = [0, 904] \) is contained in \( F(X, Y) \).
- This important property is called inclusion isotonicity.
- In the right-hand side of equation (9), we use the monotonicity of the exponential function to get the result.
- This property is also very important for bounding operations and is called inclusion monotonicity.

To this end, the following fundamental theorem of interval arithmetic can be stated:

**Fundamental Theorem.** Let \( F(x_1, x_2, ..., x_n) \) be the natural interval extension of \( f(x_1, x_2, ..., x_n) \) then

\[
f(X_1, X_2, ..., X_n) \subseteq F(X_1, X_2, ..., X_n).
\]

And for all intervals,

\[
Y_k \subset X_k \text{ for } k = 1, ..., n, f(Y_1, Y_2, ..., Y_n) \subseteq F(Y_1, Y_2, ..., Y_n),
\]

where

\[
f(Y_1, Y_2, ..., Y_n) = \{ f(x_1, x_2, ..., x_n) : x_k \in Y_k \text{ for } k = 1, ..., n \}.
\]

This theorem due to Moore [15] was extended by Hansen [16]. It states that, the evaluation of a function in any point inside a closed interval where both the function and variable are defined, is guaranteed to produce a result contained in the natural interval extension of the function. Its proof can be found in [16]. This fundamental theorem shows how easy it is to bound the range of a function, and makes possible the resolution of the bounding problem stated in Section 1. In the next subsection, we will apply this theorem to the delta-lognormal function as a specific sequence of interval arithmetic operations to bound the range of equation (4) within an interval vector.

**B. Algorithm for computing the upper and lower bound of the delta-lognormal function.**

To compute a bound of equation (4), we first define the bounding space of its variables as:

\[
p^1 = \{ [\mu_x, \mu_{\sigma}], [\mu_y, \mu_{\alpha}], [\mu_{\sigma}, \mu_{\alpha}], [\mu_{\sigma}, \mu], [\mu_{\alpha}, \mu], [\mu, \mu_{\alpha}], [\mu, \mu_{\sigma}] \} \tag{10}
\]

Then, through the Fundamental Theorem we apply the natural interval extension on \( f \). This yields:

\[
f(P^1) \in F(P^1) = [f(P^1), \tilde{f}(P^1)]
\]

\[
= [\int [f(t) - \Delta(t; P^1)]^2 dt \tag{11}
\]

As provided by the theorem and the natural interval extension, the result of the right-hand side of equation (11) is an interval \([\tilde{f}(P^1), \tilde{f}(P^1)]\) that bounds \( f \). To compute that result, we break down the last term of equation (11), into a unique finite sequence of interval arithmetic operations presented below as Algorithm 1.

**Algorithm 1:** Eval_Lognormal \([\mu, \mu_{\alpha}], [\mu_x, \mu_{\sigma}], [\mu_y, \mu_{\alpha}], [\mu_{\sigma}, \mu], [\mu, \mu_{\alpha}], [\mu, \mu_{\sigma}], t] \)

1. Set the constant variable const=\( \sqrt{2\pi} \);
2. Compute equation (2) using the following list of expressions (‘;’ are used as separators for brevity).

\[
T_1 = \left[ \mu_x, \mu_{\sigma} \right]^2; \quad T_2 = t \left[ t_0, t_0 \right];
\]

\[
T_3 = \ln(T_2); \quad T_4 = T_3 - \left[ \mu_y, \mu_{\alpha} \right];
\]

\[
T_5 = T_3^2; \quad T_6 = \frac{t_0}{T_4}; \quad T_7 = 0.5 \times T_6;
\]

\[
T_8 = -T_7; \quad T_9 = \exp(T_8);
\]

\[
T_{10} = \text{const} \times \left[ \mu_y, \mu_{\alpha} \right]; \quad T_{11} = T_2 \times T_{10};
\]

\[
T_{12} = \frac{t_0}{T_{11}}.
\]

3. return \( T_{12} \);

This algorithm starts by expressing \( f \) as a code list, where each arithmetic function (\( \pm, \times, \div \), etc.) and unary elementary functions of the type (exp, ln, ... ) are expressed with only one unique variable in their respective expressions. In this way, we ensure that the desirable properties (inclusion isotonicity, etc.) of the Fundamental Theorem are satisfied. Another advantage of using this method is the sharpness of the bound that results from this calculation process. In summary, Algorithm 1 takes as input each lognormal parameter \( P_i \) and the sample time \( t \). From steps 1 to 3, the delta-lognormal function is computed by calling Eval_Lognormal for each lognormal.

**IV. TESTING THE ALGORITHM UNDER IDEAL CONDITIONS**

The algorithm was tested through two experiments:

- The ability of the resulting envelope to maintain, in a defined finite space, all the enclosed spaces of this set.
- Its ability to enhance the performance of the extractor in [14], when used as a bounding operation. In this test we do not take into account the dependencies between the different parameters. Each parameter space was set according to its domain.

**A. Tests conditions**

The experiments performed to validate Algorithm 1 use the database proposed in [12] and [14]. This database is composed of 7000 delta-lognormal curves, grouped into categories. These ideal curves were generated automatically using parameter sets derived from human handwriting movements, previously extracted using the IIX algorithm presented in [12]. The choice to use synthetic data for the test is justified by the fact that the exact values of the parameters of the speed profiles to be tested are known. Therefore, the extraction algorithm can be validated based on its ability to find the same values as in the truth table. To this end, we chose randomly 300 ideal delta-lognormal parameters from class Ca0 and Ca2 (see Fig 1). Each ideal velocity profile, constructed from the ideal delta-lognormal parameters (by evaluating equation 1), was sampled at 200Hz to simulate the data collected with a digitizer. In the second experiment, we replaced the boundary operations contained in the branch and bound algorithm proposed in [14] by Algorithm 1. The resulting speed profiles were then fed to this reconstructed algorithm to try to recover the true parameters value.
We carried out tests using a 3.60 GHz Intel core (i7) with 32 Go of RAM. The implementation was done in C++ using our custom interval arithmetic library based on [15] and [17]. All computations were performed using rounded interval arithmetic. 

B. Results and discussion

For the first experiment, Table I presents the search region which is restricted, for simplicity, to a finite box where the solution can be found.

TABLE I. THE INITIAL BOX

<table>
<thead>
<tr>
<th>Parameters</th>
<th>( t_0 )</th>
<th>( D_1 )</th>
<th>( \mu_1 )</th>
<th>( \sigma_1 )</th>
<th>( D_2 )</th>
<th>( \mu_2 )</th>
<th>( \sigma_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.0001</td>
<td>50</td>
<td>-5</td>
<td>0.001</td>
<td>10</td>
<td>-5</td>
<td>0.001</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.9</td>
<td>500</td>
<td>0.005</td>
<td>1.0</td>
<td>300</td>
<td>0.005</td>
<td>0.9</td>
</tr>
</tbody>
</table>

As shown in Fig 2, thick black lines represent the envelope produced by Algorithm 1 with the bounding box values of Table 1. Inside this envelope, more than 3000 delta-lognormal curves are shown as thin gray lines. The parameters of these curves have been generated by splitting the bounding box into sub-boxes such that \( P_k \subset \mathbb{P} \) for \( k=1 \ldots n \). As can be seen, the envelope bounds the possible values of equation (2) inside the given space quite well, since no curve extends beyond these borders.

Table II and III present the main statistics of the absolute error on each estimated parameter. As can be seen for the 300 tested curves, the errors on the estimates of \( t_0 \) and \( D_1 \) are on average \( 10^{-9} \) and \( 10^{-6} \) respectively. We can also observe that the new algorithm always finds not only the true value within an accuracy of \( \varepsilon = 10^{-6} \) but a confidence interval around the true value (Table III). This interval contains all the possible solutions satisfying at least the ending criteria. Thus, if we decide to generate a solution with a smaller error (up to the limit of numerical precision), we just need to continue the search from that bounding box. This is an advantage of using interval arithmetic. All the results compiled in this table were obtained for a signal to noise ratio of more than 100dB.

Fig.3 presents an example in which the delta-lognormal parameters of a real human generated velocity profile of a pen-tip movement are extracted using Algorithm 1 combined with [14]. As can be seen in the middle trace of Fig.3, the extracted velocity profile fits quite well (to an SNR=31dB) the human speed profile of the pen-tip movement shown in the top trace of Fig.3. The bottom trace in Fig.3 shows the extracted velocity profile decomposed into two neuromuscular systems acting in direct opposition as predicted by the delta-lognormal model. The agonist and the antagonist are respectively plotted in the figure by dashed and thin lines around the continuous line movement.
Table IV shows a performance comparison in terms of processing time and number of boxes explored for the proposed algorithm and the one proposed in [15]. In [15], the algorithm was executed in parallel, on a 12-CPU node grid. As can be seen in Table IV below, this algorithm requires a run time of about 50s to perform 11 iterations of its main loop, while the one we propose requires only 702ms. On the other hand, the number of boxes needed to find a solution of the desired quality can reach one million with the previous algorithm, while ours requires at most 70 thousand boxes to extract such a solution.

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Boxes</td>
<td>Min 800000</td>
</tr>
<tr>
<td>Processing time (s)</td>
<td>50</td>
</tr>
</tbody>
</table>

V. CONCLUSION

In this paper we have shown that, with a good bounding method applied to the delta-lognormal function, when embedded in a branch and bound algorithm, the parameters of delta-lognormal velocity profiles can be extracted with less computational costs. The effectiveness of the proposed bounding method is quite remarkable. The number of evaluations was reduced 12-fold to obtain results of a same target quality. This algorithm exploits the natural interval extension and the fundamental theorem of interval arithmetic to compute the bounding operations of the delta-lognormal function. Although the proposed algorithm is promising in its ability to recover a global solution in a processing time much smaller than a previously reported method, more work remains to be done. The proposed algorithm should be tested on all classes of velocity profiles that the delta-lognormal expression can model, and it should be more thoroughly tested on real human movement data.

REFERENCES

The Delta-Lognormal model in 2.5D

Saad Chidami  
Dept. of Electrical Engineering  
Polytechnique Montréal  
Montréal, Canada  
saad.chidami@polymtl.ca

Mylène Archambault-Caron  
Dept. of Electrical Engineering  
Polytechnique Montréal  
Montréal, Canada  
mylene.archambault-caron@polymtl.ca

Réjean Plamondon  
Dept. of Electrical Engineering  
Polytechnique Montréal  
Montréal, Canada  
rejean.plamondon@polymtl.ca

Abstract—In the context of the Kinematic Theory of Rapid Human Movement, this paper investigates the extension of the Delta-Lognormal model from two dimensions (2D) to three dimensions (3D). As an intermediate goal, a 2.5D parameter extraction method is developed and preliminary results are discussed.

Index Terms—Kinematic Theory of rapid human movements, Delta-Lognormal model, 3 Dimensions, Velocity profiles, Parameter estimation

I. INTRODUCTION

The Delta-Lognormal model [1], [2], [3], [4], have been used in many applications dealing with 2D data, mostly collected using digitizers. It has been used to validate the Kinematic Theory of rapid human movements in EMG experiments to demonstrate the proportionality between the cumulative time delays of response of the different muscles as it propagates toward the effector following a command [5] as well as in EEG experiments to predict the time occurrence of an evoked potential associated with a motor command using the parameter \( t_0 \) [6]. Moreover, it has been used in basic experiments dealing with the definition of a movement primitive [7], [8], with speed-accuracy trade-offs [9], [10], as well as to study aging [11], study the impact of brain stroke risk factors on human movements [12], [13].

Over the years, it has been integrated into a basic methodology [14] to evaluate the neuromuscular control of subjects using various reaction time experiments (simple, choice, speed-accuracy trade-offs) based on auditory or visual stimuli using a digitizer to collect the pen tip data from which the velocity profiles could be computed and processed with parameter extraction algorithms [15], [16]. All these tests involve 2D movements in a plane, which reduce the set of upper arm movement that can be studied with the Delta-Lognormal model. Some preliminary studies have extended the model to 3D [17], which makes the parameter extraction more complex. In this paper we explore a way to extend these psychometric tests to any pointing movement in 3D, using what we will call here after a 2.5D parameter extraction processing where we assume that every 3D short and fast hand stroke is a "limb restricted" 2D movement on a 3D coordinates plane. In the next section, we present the experimental set up and protocol and sections III and IV are devoted to experimental results and data analysis.

In 2D, the Delta-Lognormal model describes the speed profile of fast and straight movements [1]. Indeed, it represents the synergetic activation of two neuromuscular systems when an action is executed, one is agonist and the other is antagonist to the action. This action can be represented by the parameters \( D_1 \) and \( D_2 \), which represent the amplitudes of the input commands of the agonist and antagonist neuromuscular systems, as well as by \( t_0 \), the time of occurrence of the muscular command. In addition, the agonist and antagonist systems are additionally represented by the parameters \( \mu_1, \sigma_1, \mu_2, \sigma_2 \), where \( \mu_i \) represent the logtime delay, and \( \sigma_i \) represent logresponse time of the neuromuscular system.

The result of the synergy of those neuromuscular systems, is a rapid movement which the speed profile over time can be written as a Delta-Lognormal equation:

\[
|v(t - t_0)| = D_1 \Lambda_1(t; t_0, \mu_1, \sigma_1) - D_2 \Lambda_2(t; t_0, \mu_2, \sigma_2)
\]

where \( \Lambda \) is expressed as:

\[
\Lambda_i(t; t_0, \mu_i, \sigma_i) = \frac{1}{\sigma_i \sqrt{2\pi(t - t_0)}} e^{-\frac{\ln[(t - t_0) - \mu_i]^2}{2\sigma_i^2}}
\]

II. EXPERIMENTAL SETUP AND PROTOCOL

In order to assess that, the Delta-Lognormal model can be used to investigate simple and fast human movements in 3D, the velocity profile of such movement must be fitted by (1).

A. Data Acquisition

Data acquisition was performed using a Vicon system. This system consists of six high-speed synchronized cameras which frame rate was set to 200 frames per second. The system’s image processing module returns the three-dimensional coordinates of a target as a function of time using an integrated triangulation algorithm. The system is calibrated before each data acquisition to ensure a very small measurement error (0.25mm).

In this paper, we focus on the study on the human upper limbs and record trajectories of short hand movements using a target whose grip handling is similar to a pencil. As we are investigating the Delta-Lognormal model in 3D, all the recorded movements are similar to rapid “pencil” strokes in 3D space.
A Type II Chebyshev Filter of a 11th-order with 16Hz cut-off frequency was applied to the x, y and z coordinates extracted from the recording system (Fig. 1). The tangential velocity is calculated with (3) after preprocessing data with a derivative filter (Fig. 2).

\[ v_t = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2} \] (3)

Fig. 2. Velocity profile of a 3D hand movement.

C. Principal Component Analysis and 3D to 2D transformation

As we can observe Fig. 1, the trajectory of the movement is slightly curved, and we can assume that this curvature is an intrinsic property of the human neuro-motor response. We can also presume the origin of this curvature which can be the consequence of the wrist rotation or the elbow rotation for short and rapid hand movement.

From this last hypothesis, we assume that every 3D short and fast hand stroke is in fact a “limb restricted” 2D movement on a 3D coordinates plane. To find the least errors plane approximation, we use a Principal Component Analysis (PCA). This statistical method allows us to extract the best projection plan for each hand movement trajectory.

\[ N = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \]

The 3D space orientation of this vector can be defined by its angles of rotation relative to the z axis and the x axis. In order for this plane to be parallel to the xy plane, it is necessary to transform the \( N \) vector in coordinates such as:

\[ N_p = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \]

For this, we can define the angle of rotation with respect to z and a vector on the xy plane such as:

\[ \theta_z = \cos^{-1} \left( \frac{n_y}{\sqrt{n_y^2 + n_z^2}} \right) \] (4)

Then, the rotation matrix to correct the rotation around the z axis is defined by:

\[ R_z = \begin{bmatrix} \cos(-\theta_z) & \sin(-\theta_z) & 0 \\ -\sin(-\theta_z) & \cos(-\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \]

And the angle of rotation around the x axis:

\[ \theta_x = \cos^{-1} \left( \frac{n_z}{\sqrt{n_x^2 + n_z^2}} \right) \] (5)

The rotation matrix to correct the rotation around the x axis is defined by:

\[ R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-\theta_x) & \sin(-\theta_x) \\ 0 & -\sin(-\theta_x) & \cos(-\theta_x) \end{bmatrix} \]

And finally \( N_p \) can be solved by:

\[ N_p = R_z R_x N = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \]

As can be seen in Fig. 3, the rotations around the z axis and the x axis of the plan make it possible to obtain a new trajectory parallel to the xy plane. The new coordinates of the trajectory can be considered as 2D since we can neglect the z coordinates because they are constant. The new transformed 2D coordinates are called \( uw \). To reconstruct the signal, the new calculated parameters \( \theta_x \) and \( \theta_z \) are added to the 3D Delta-Lognormal model.

D. Delta-Lognormal parameters extraction

Extracting Delta-Lognormal parameters from a pencil stroke only requires the tangential velocity. Thus the IIX extractor algorithm [15] has thus been used to extract the parameters from the 3D tangential velocity and the transformed 2D tangential velocity. The results will be discussed later.
E. Signal Reconstruction

In order to complete the 3D modeling of fast and simple human upper limb movements, it is necessary to extract two other parameters $\theta_s$ and $\theta_e$, respectively as the start and the ending angles of the 2D transformed movement trajectory. These parameters are added to the 9 parameters $D$, $t_0$, $\mu$, $\sigma$ and finally $\theta_x$ and $\theta_z$. These parameters are extracted using 2D trajectories; the starting angle is extracted from the first half sampling points of the trajectory (removing a first portion that could correspond to a break time). A linear regression is applied on this portion of trajectory and the start angle is calculated as the angle between the regression line and the $u$ axis. The ending angle is measured from the second half of the trajectory, the final portion is removed and is considered as the resting time. As for the starting angle, a linear regression is applied on this portion of the 2D trajectory and the ending angle is calculated as the angle between the regression line and the $u$ axis.

The reconstruction of the signal or the 2D trajectory is similar to the algorithm used by Djioua and Plamondon [15]:

$$
\begin{bmatrix}
u_u(t) \\
v_w(t)
\end{bmatrix} = v(t) \begin{bmatrix} \cos \theta(t) \\ \sin \theta(t) \end{bmatrix}
$$

with:

$$
\theta(t) = \theta_s + \frac{\theta_e + \theta_s}{D_1 - D_2} \int_0^t v(\tau)d\tau
$$

and $v(t)$ the tangential velocity reconstructed from the extracted Delta-Lognormal parameters. To map those 2D coordinates in 3D, we just need to apply the inverse rotation matrix. Since the transformation matrix is the one that transforms 3D motion into 2D, the inverse matrix allows the transformation of 2D motion into 3D:

$$
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} = \begin{bmatrix}
u \\
w \\
1
\end{bmatrix} M_{rotation}^{-1}
$$

Fig. 3. Resulting transformation of the velocity profile.

III. Experimental Results

Since the IIX extractor algorithm only require the tangential velocity, in a first step, we had to determine if the 3D to 2D transformation did not alter the tangential velocity of the signal. In fact, we proved that the dimension reduction had no effect on the tangential velocity, such that 2D tools can be used in a 3D analysis.

Fig. 4 shows the superposition of the tangential velocities of a typical 3D collected data and its 2D transformed data respectively raw data fig. 4.a) and filtered data fig. 4.b). For raw data, small differences are apparent, mostly due to the attenuation of the peaks that one could attribute to shaky hand movement (worst case scenario). Indeed, on the 2D transformed data, these peaks are slightly attenuated. When the low-pass filter is applied on the data, we can conclude that the 3D to 2D transformation is not removing informations from the tangential velocity.

The Delta-Lognormal parameters were extracted from the 3D trajectory or the 2D transformed data and a signal noise ratio (SNR) is calculated with (6), where $v_i$ are $v_x$, $v_y$ and $v_z$. And where $v_{1m}$, $v_{2m}$, $v_{3m}$ and $v_{4m}$ the velocity generated
by the reconstruction.

$$SNR = 10 \log \left( \frac{\int [v_x^2(t) + v_y^2(t) + v_z^2(t)]dt}{\int \sum (v_{i}^2(t) - v_{im}^2) } \right)$$  \hspace{2cm} (6)$$

As preliminary tests, a set of 10 simple fast hand movements, from the same person, without elbow restriction was recorded. Then 8 simple fast hand movements, with elbow on a table were recorded. Finally, 13 more complex figures, in a triangular shape, were recorded. The results in terms of SNR are illustrated in fig. 1 and fig. II.

TABLE I

<table>
<thead>
<tr>
<th>Type of stroke</th>
<th>Statistic analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>unrestricted</td>
<td>28.33</td>
</tr>
<tr>
<td>restricted</td>
<td>26.6</td>
</tr>
<tr>
<td>triangle</td>
<td>28.8</td>
</tr>
</tbody>
</table>

TABLE II

<table>
<thead>
<tr>
<th>Type of stroke</th>
<th>Statistic analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>unrestricted</td>
<td>28.14</td>
</tr>
<tr>
<td>restricted</td>
<td>26.92</td>
</tr>
<tr>
<td>triangle</td>
<td>28.64</td>
</tr>
</tbody>
</table>

To illustrate these results, a typical example of the 3D reconstructed signal is illustrated in the fig. 5

Fig. 5. Simple hand stroke and its reconstruction.

IV. DATA ANALYSIS

The hypothesis that the Delta-Lognormal model can be applied to simple 3D hand movement has been validated in this study. We can also conclude that the transformation of 3D to 2D by using a PCA to project the trajectory on a plane does not alter significantly the tangential velocity. The SNR values (range from 26 to 28) obtained from this 2.5D method are equivalent to native 2D strokes analysis. According to our preliminary results, there are no statistical differences between the extraction of Delta-Lognormal parameters from the 3D data or the 2D transformed data. However, it was observed, when analyzing curved rapid hand stroke that the parameter extraction from the tangential velocity was giving a high SNR, but the reconstruction was incorrect. An additional curvature parameter will be needed in the future for a better signal reconstruction or use a more complex model like the Sigma-Lognormal, but the present methodology allows to study various restricted 3D movements in simple psychophysical experiment under realistic conditions.

ACKNOWLEDGMENTS

This work was supported by grants to Saad Chidami and Mylène Archambault-Caron from the UPIR fellowship program of École Polytechnique and by NSERC grant RGPIN-2015-06409 to Réjean Plamondon.

REFERENCES


Extending the Sigma-Lognormal Model of the Kinematic Theory to Three Dimensions

Roman Schindler*, Manuel Bouillon*, Réjean Plamondon† and Andreas Fischer*‡

*DIUF department, DIVA group, University of Fribourg, Fribourg, Switzerland
†Laboratoire Scribens, École Polytechnique de Montréal, Montréal, Québec H3T 1J4, Canada
‡iCoSys Institute, University of Applied Sciences Western Switzerland, Fribourg, Switzerland
Email: {roman.schindler,manuel.bouillon,andreas.fischer}@unifr.ch, rejean.plamondon@polymtl.ca

Abstract—The Kinematic Theory of rapid human movements and its Sigma-Lognormal model enables to model human gestures, in particular complex handwriting patterns such as words, signatures and free gestures. This paper investigates the extension of the theory and its Sigma-Lognormal model from two dimensions to three, taking into account new acquisition modalities (motion capture), multiple subjects, and unconstrained motions. Despite the increased complexity and the new acquisition modalities, we demonstrate that the Sigma-Lognormal model can be successfully generalized to describe 3D human movements. Starting from the 2D model, we replace circular with spherical motions to derive a representation of unconstrained human movements with a new 3D Sigma-Lognormal Model. First experiments show a high reconstruction quality with an average signal-to-noise ratio (SNR) of 18.52 dB on the HDM05 dataset. Gesture recognition using dynamic time warping (DTW) achieves similar recognition accuracies when using original and reconstructed gestures, which confirms the high quality of the proposed model.

Keywords—Kinematic Theory of rapid human movements, Sigma-Lognormal model, trajectory reconstruction, 3D motion recognition, dynamic time warping

I. INTRODUCTION

The Kinematic Theory of rapid human movements is used to analyze human movements as a process depending on the neuromuscular parameters of the human body. Applied to handwriting, it can be used to study writer expertise [1], [2], to verify genuine signatures [3], [4], or to synthesize artificial handwriting [5], [6], to name just a few applications.

In this paper, we investigate the extension of the Sigma-Lognormal model to three dimensions to analyze 3D human movements. This modeling of 3D motions with the Kinematic Theory of rapid human movements could potentially lead to a wide new range of applications including, for example, neuromuscular disorder analysis, movement synthesis for robotics and computer games, and person identification by gait analysis.

We pursue a natural extension of the model by replacing circular with spherical motions for individual lognormal strokes (which are hypothesized to act along a pivot). While the estimation of the lognormal parameters from the velocity profile remains the same as in 2D, we integrate polar angles in addition to azimuthal angles and adapt the estimation of the angular parameters, accordingly.

The resulting 3D Sigma-Lognormal model is empirically tested on the HDM05 motion capture database, which contains different motion classes, such as walking, dancing, kicking, etc. We reconstruct the movements with our proposed model and assess the quality of the reconstruction with a signal-to-noise ratio (SNR). Furthermore, we conduct a classification experiment based on dynamic time warping (DTW), both, with the original and with the reconstructed trajectory, as an additional assessment of the reconstruction quality.

The remainder of this paper is organized as follows. Section II presents the Kinematic Theory of rapid human movements and the 2D Sigma-Lognormal model. Section III introduces the equations of the proposed 3D extension. In Section IV, we present the results of our empirical evaluation. Section V concludes and discusses future work.

II. KINEMATIC THEORY OF RAPID HUMAN MOVEMENTS

This section briefly presents the Kinematic Theory of rapid human movements with the original Sigma-Lognormal model [7] in two dimensions, which we will extend to three dimensions in the next section.

A. The Sigma-Lognormal Model

The kinematic theory postulates that any movement is the combination of movement primitives, so-called strokes, with lognormal speed. Those strokes are initiated at time $t_0$ in the central nervous system with an intended distance $D$. They are then actuated with a log time delay $\mu$ and a log response time $\sigma$. The Kinematic Theory formulates the speed of a stroke as

$$|\dot{v}(t)| = \frac{D}{\sqrt{2\pi \cdot \sigma(t-t_0)}} \exp\left(-\frac{[\ln(t-t_0)-\mu]^2}{2\sigma^2}\right).$$

Assuming the movement acts along a pivot, the angular position of each stroke is given by:

$$\theta(t) = \theta_s + \frac{\theta_e - \theta_s}{D} \int_0^t |\dot{v}(\tau)|d\tau,$$

where $\theta_s$ is the start angle and $\theta_e$ is the end angle.

Simplest rapid movements are composed of two strokes, the strokes of the agonist and of the antagonist action needed to execute the movement. Those two strokes are combined according to the Delta-Lognormal model ($\Delta\Lambda$). More complex movements in two dimensions are described as the sum of the different strokes according to the Sigma-Lognormal model.
In the general case, 2D movements (like handwriting) are described as a vector sum of strokes

\[ \vec{v}(t) = \sum_{i=1}^{n} \vec{v}_i(t) , \]

where \( n \) is the number of strokes.

Accordingly, the velocity and position of the movement in \( x- \) and \( y- \) direction are

\[ v_x(t) = \sum_{i=1}^{n} |\vec{v}_i(t)| \cos(\theta_i(t)) , \]

\[ v_y(t) = \sum_{i=1}^{n} |\vec{v}_i(t)| \sin(\theta_i(t)) , \]

\[ x(t) = \int_{0}^{t} v_x(\tau) d\tau , \]

\[ y(t) = \int_{0}^{t} v_y(\tau) d\tau . \]

In summary, a 2D movement can be represented as a combination of lognormal strokes \((l_1, \ldots, l_n)\) with six parameters each:

\[ l = (t_0, D, \mu, \sigma, \theta_s, \theta_e) . \]

For more details about the Kinematic Theory of rapid human movements, we refer the reader to [9]–[12].

B. Stroke extraction and parameters estimation

In order to represent a movement with the Sigma-Lognormal model, the input is first preprocessed, and then the different strokes are extracted with their respective parameters.

The preprocessing is minimal, but required to enable the correct modeling of the whole movement. The movement is artificially stopped at the beginning and at the end of the motion by artificially holding the respective positions for 200ms [3]. Reducing the speed at the border of the trajectory to zero improves the extraction of the first and the last lognormal stroke. If the movements have been acquired with different devices, it is recommended to interpolate the input trajectories at a common sampling rate, i.e. 200Hz [3]. If necessary, a Chebyshev filter can also be applied to remove high-frequency components if some noise was introduced by the acquisition device.

From the input trajectory, we get the observed velocity \( \vec{v}_o(t) \), and then three steps are applied to extract the different strokes as illustrated in Figure 1. First, the local minima and maxima of the speed profile \( |\vec{v}_o(t)| \) are used to detect the biggest stroke \( l \). Second, the parameters \( l = (t_0, D, \mu, \sigma, \theta_s, \theta_e) \) of this stroke are estimated based on an initial analytical solution using the Robust XZERO algorithm [13]. These initial solutions are then refined by means of non-linear least squares curve fitting. Third, the extracted stroke is added to the estimated model and its estimated velocity \( \vec{v}_e(t) \) is subtracted from the observed velocity \( \vec{v}_o(t) \). This three-step process is repeated until the signal-to-noise ratio (SNR) cannot be further improved.

C. Model quality assessment

The quality of the reconstructed movement can be used to assess the quality of the model by means of a signal-to-noise ratio (SNR) between the observed movement \( \vec{v}_o(t) \) and the reconstructed movement \( \vec{v}_e(t) \)

\[ SNR = 10 \cdot \log \left( \frac{\int_{t_s}^{t_e} |\vec{v}_o(\tau)|^2 d\tau}{\int_{t_s}^{t_e} |\vec{v}_o(\tau) - \vec{v}_e(\tau)|^2 d\tau} \right) , \]

where \( t_s \) is the starting time and \( t_e \) the ending time of the movement.

III. EXTENSION TO THREE DIMENSIONS

The Kinematic Theory of rapid human movements assumes that strokes act along a pivot. The strokes can be described in the two dimensional plane with the distance to the origin \( \rho \) and one angle \( \theta \). In order to model human motions in three dimensions, an additional angle \( \phi \) is required as shown in Figure 2.

---

**Image 1.** Workflow of the Sigma Lognormal parameters estimation process. Illustration from O’Reilly [3]

**Image 2.** Illustration of the movement 3D model with \( \phi(t) \) in addition to the 2D representation with \( \theta(t) \) and \( \rho = |\vec{v}(t)| \).
When adding a third dimension, the velocity value from Equation 1 does not change, but the angular position of Equation 2 now depends on a second angle $\phi$

$$\phi(t) = \phi_s + \frac{\phi_s - \phi_e}{D} \int_0^t |\ddot{v}(\tau)| \, d\tau .$$

(10)

The velocity Equations 4 and 5 become

$$v_x(t) = \sum_{i=1}^M |\ddot{v}_i(t)| \sin(\phi_i(t)) \cos(\theta_i(t)) ,$$

(11)

$$v_y(t) = \sum_{i=1}^M |\ddot{v}_i(t)| \sin(\phi_i(t)) \sin(\theta_i(t)) ,$$

(12)

and the velocity in the z-direction is

$$v_z(t) = \sum_{i=1}^M |\ddot{v}_i(t)| \cos(\phi_i(t)) .$$

(13)

The position of the movement in x- and y-direction in Equations 6 and 7 do not change, and the position in z-direction is

$$z(t) = \int_0^t v_z(\tau) \, d\tau .$$

(14)

In summary, a 3D movement can be represented as a combination of lognormal strokes $(l_1, \ldots, l_n)$ with eight parameters each

$$l = (t_0, D, \mu, \sigma, \theta_s, \theta_e, \phi_s, \phi_e) .$$

(15)

The estimation process of Figure 1 does not change, only the angle estimation step is extended to estimate the two new parameters. The new angle parameters $\phi_s$ and $\phi_e$ can be estimated in a similar way as the original parameters $\theta_s$ and $\theta_e$ with

$$\phi_{s,n}(t) = \arccos \frac{v_{o,z}(t)}{v_o(t)} ,$$

(16)

$$\Delta\phi = \frac{\phi_{s,n}(t_{2,n}) - \phi_{s,n}(t_{2,n})}{l(t_3) - l(t_2)} ,$$

(17)

$$\phi_s = \phi_{s,n}(t_{2,n}) - \Delta\phi(l(t_3) - l(t_1)) ,$$

(18)

$$\phi_e = \phi_{s,n}(t_{2,n}) - \Delta\phi(l(t_3) - l(t_3)) ,$$

(19)

where $t_i$ are the times of the points $p_i$ as follows:

$p_1$ Lognormal stroke beginning
$p_2$ First inflexion point
$p_3$ Local maximum of the lognormal stroke
$p_4$ Second inflexion point
$p_5$ Lognormal stroke ending

IV. Experiments and Results

In this section, we present the results of our empirical evaluation conducted with the proposed 3D extension of the Sigma-Lognormal model. First, we measure the signal-to-noise ratio (SNR) on the HDM05 dataset to assess the model quality of the 3D gestures. Second, we investigate the impact of using synthetic gestures for a dynamic time warping (DTW) classifier to explore the potential for applications in the field of 3D action recognition.

### A. Dataset

For evaluating the 3D Sigma-Lognormal model, we use the HDM05 dataset [14]. HDM05 is a motion capture (mocap) dataset that contains the trajectories of various points on the skeleton that was recorded with a suit containing more than 40 markers. The input data are the 3D trajectories of the suit markers recorded at 120Hz, which allows to evaluate the 3D Sigma-Lognormal model with a high precision. The dataset is composed of roughly 100 classes that were performed 10 to 50 times by 5 subjects, amounting to 1,457 samples in total.

From the skeleton data, we compute the trajectories of wrists and ankles relative to shoulders and hips, respectively, which yield four limb trajectories [15]. Those limb vectors are then normalized by the limb length to get a motion representation that is independent of the morphology of the subject [16]. This allows to get a simplified representation of the skeleton, that is independent of the position of the skeleton but that preserves the main characteristics of the movements. We also use 11 selected actions [17] for our experiments (deposit floor, elbow to knee, grab high, hop both legs, jog, kick forward, lie down floor, rotate both arms backward, sneak, squat, throw basketball), resulting in 249 samples.

Each limb trajectory is represented as a sequence of $(x, y, z)$ coordinates, from which we can extract the sequences of velocities $(v_x, v_y, v_z)$ and accelerations $(a_x, a_y, a_z)$, which gives a total of 36 features. All those features are computed with second order regression [18], and they are normalized by a z-score normalization over all sampling points [4].

### B. Trajectory Reconstruction Quality

To evaluate the quality of the representation of the 3D motions with the Sigma-Lognormal model, we measure the signal-to-noise ratio (SNR) between the original and reconstructed trajectories [7]. To reconstruct trajectories, we extract the Sigma-Lognormal model parameters from the 36 features of the input samples, and then we try to reconstruct the motions (with the 36 features) from the Sigma-Lognormal model parameters.

Figure 3 shows the original (in blue) and reconstructed (in red) trajectories of the wrists and ankles of a kick motion sample. Visually, the original and reconstructed trajectories look very much alike. Figure 4 shows the original (in blue) and reconstructed (in red) velocity profiles of the four limbs of the same kick motion sample. Again, the reconstruction is very close to the original.

In order to quantify the quality of the reconstruction, we measure the SNR between the original and reconstructed trajectories [7]. Figure 5 presents the SNR distribution for all the four reconstructed trajectories of the 249 samples (hence 996 reconstructions). The mean SNR is $18.52dB$ (with a standard deviation of $4.09dB$) which suggests a high quality
of the 3D Sigma Lognormal modeling. The general quality of the 3D reconstructions is as good as the quality of the 2D reconstructions, which we can obtain by discarding the $z$—dimension, as shown in Table I.

### C. Motion Recognition Results

In order to further investigate the quality of the proposed 3D Sigma-Lognormal model, we use reconstructed movements to perform action recognition on the HDM05 dataset. We use dynamic time warping (DTW) to compute a distance between two movements. To avoid unusual warping paths and to speed up the computation, a Sakoe-Chiba band [22] is employed.

The classification of a test sample is done by computing the DTW distance to a set of reference samples and predicting the class of the nearest reference sample.

First, we study the importance of the position, velocity and acceleration profiles. Table II presents the recognition accuracy obtained when using the original movements of four subjects as reference samples and those of the remaining subject as test samples. The average accuracy and the standard deviation (SD) over the five experiments are indicated. The best results are obtained when using only the velocity profile. In fact, adding either the position or acceleration (or both) yield lower recognition accuracies. Since the velocity profile performed best, we proceeded using only 12 velocity features (3 dimensions and 4 limbs).

Next, we compare our classification results with other state-of-the-art methods. Table III shows the results of the proposed velocity-based DTW classifier on the same train/test split that is typically used in the literature [17]. The results indicate that our classifier is able to reach state-of-the-art performance.

Finally, we compare the recognition accuracy obtained when using reconstructed movements as reference samples,

---

**Table II.** Comparison of the different combinations of input data (four subjects as references and one subject for testing).

<table>
<thead>
<tr>
<th>DTW input data</th>
<th>Accuracy (%)</th>
<th>SD (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>93.61</td>
<td>4.14</td>
</tr>
<tr>
<td>Velocity</td>
<td><strong>96.44</strong></td>
<td>4.54</td>
</tr>
<tr>
<td>Acceleration</td>
<td>91.51</td>
<td>5.56</td>
</tr>
<tr>
<td>Position + Velocity</td>
<td>95.35</td>
<td>3.39</td>
</tr>
<tr>
<td>Velocity + Acceleration</td>
<td>95.76</td>
<td>3.33</td>
</tr>
<tr>
<td>Position + Acceleration</td>
<td>95.43</td>
<td>2.81</td>
</tr>
<tr>
<td>Position + Velocity + Acceleration</td>
<td>95.34</td>
<td>4.17</td>
</tr>
</tbody>
</table>

---

**Table III.** Comparison of recognition accuracy on the HDM05 benchmark (using the proposed train-test split [17]).

<table>
<thead>
<tr>
<th></th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cov3DJ + SVM [19]</td>
<td>95.41</td>
</tr>
<tr>
<td>HOD + SVM [20]</td>
<td>97.27</td>
</tr>
<tr>
<td>BIPOD + SVM [21]</td>
<td>96.70</td>
</tr>
<tr>
<td>HIF3D + SVM + Level = 2 [15]</td>
<td>98.17</td>
</tr>
<tr>
<td>Our approach (DTW + velocity profile)</td>
<td><strong>97.25</strong></td>
</tr>
</tbody>
</table>

---

Fig. 3. Original trajectories (blue) and reconstructed (red) trajectories of wrists and ankles in a kick motion sample.

Fig. 4. Original (blue) and reconstructed (red) velocity profiles of right wrist, left wrist, right ankle and left ankle of the kick motion of Figure 3.

Fig. 5. Distribution of signal-to-noise ratio (SNR) values of the reconstructed 3D trajectories of the HDM05 dataset (996 samples).

Copyright © 2018 by CENPARMI, Concordia University
while still testing on the original test samples. We use five-fold cross-subject splitting and vary the number of subjects whose movements are used as reference sequences. The samples of all remaining subjects are used for testing. Each setting is run five times, accordingly, and the results are averaged. Table IV shows the results obtained when using original movements as reference samples or when using reconstructed movements (always using original samples for testing). The accuracy obtained with reconstructed samples is very similar to the one obtained with the original samples and a paired t-test shows no significant difference between the two sets of results ($p > 0.05$, n.s.). This result confirms the high model quality of the proposed 3D Sigma-Lognormal model.

### V. Conclusion

In this paper, we presented a 3D extension of the Sigma-Lognormal model to represent unconstrained 3D human movements. First experiments show a good model quality, as demonstrated by a high signal-to-noise ratio (SNR) of reconstructed motions and the fact that similar classification results have been achieved with original and reconstructed reference samples.

The best classification results were achieved using only the velocity profile. This observation is consistent with the underlying hypothesis of the Kinematic Theory that the velocity is the main control variable used by the central nervous system to plan and execute a movement. It is also consistent with the basic property of the velocity vector, which is tangent to the trajectory, allowing learning by observing [23].

A promising line of future research is the synthesis of 3D movements, which is expected to support classification systems with artificial but realistic training data. In general, there is a wide range of applications that could emerge from modeling 3D motions with the Kinematic Theory in biometrics, biomedicine, and robotics.

### Acknowledgments

This work has been supported by the NSERC-Canada Grant RGPIN-2015-06409 to R. Plamondon, by the European Union H2020 program with the grant number 731861, and by the National Natural Science Foundation of China with the grant number 61650110512.

### References


**A New Experimental Set-up To Run Neuromuscular Tests**

Nadir Faci  
*Dept. of Electrical Engineering*  
*École Polytechnique Montréal*  
Montreal, Canada  
nadir.fac@polymtl.ca

Simon Pierre Boyogueno Bidias, Ph.D.  
*Dept. of Electrical Engineering*  
*École Polytechnique Montréal*  
Montreal, Canada  
simon-pierre.boyogueno-bidias@polymtl.ca

Réjean Plamondon, Ph.D.  
*Dept. of Electrical Engineering*  
*École Polytechnique Montréal*  
Montreal, Canada  
rejean.plamondon@polymtl.ca

Nicolas Bergeron  
*Scribens Project Inc.*  
Montreal, Canada  
nicolas@scribensproject.com

---

**Abstract**—This article proposes an integrated and updated version of the human writing movement acquisition system Sign@medic. Based on the Wacom Cintiq technology, this new system takes advantage of all the features available in this tablet to integrate the visual and audio stimulus generator of the Sign@medic software into the tablet. The resulting embedded system which combined the acquisition system and the audio-visual stimulus in a unique tablet environment, allows users to work directly on the screen of the Cintiq 13HD tablet. The methodology, as well as a comparison between the old and new system is described and presented. This new system provides an efficient and affordable tool for researchers involved in various domains of pattern recognition and artificial intelligence, relying on studies of human movements.

**Keywords**— Sigma-Lognormal model, Kinematic Theory of rapid human movement, kinematic analysis, fine motor control, kinematic analysis, handwriting

I. INTRODUCTION

Human movement can be very complex, and their studies in different areas of medicine and human science has been intensively used. In neurosciences handwriting strokes, which constitute a specific class of rapid human movement, are used to study neurodegenerative processes involved by diseases such as Parkinson and Alzheimer’s [1] [2]. Moreover the early diagnosis of cerebral lesions appears to be possible by detecting slight deviations from the norm which are not evident by simple visual inspection.

Many of these examples emphasize the need of building robust and easily workable equipment for movement analysis. Among the various sets present in the literature and the market, the Sign@medic acquisition system developed in the Scribens laboratory has been found to be successful in acquiring handwriting movement most accurately. According to the Kinematic Theory of rapid human movement these strokes constitute the elementary movement necessary to understand how the motor control deal in performing complex movement. Over the last fifteen years, all the neuromuscular studies performed at Scribens laboratory have been done using the Sign@medic acquisition devices, based on Wacom Intuos2 technology under Windows XP operating system. This was done to make sure that we could get comparative results from one experiments to another. Such a system is no longer adequate and is about to be unsupported by the manufacturers.

This paper proposes an integrated and updated version of the acquisition device Sign@medic based on Wacom Cintiq technology. This new system takes advantage of all the recent innovations offered by Wacom Cintiq 13HD technology to provide a versatile and comfortable acquisition system. Allowing the users to work directly on the screen of this compact tablet. Excellent color and resolution, professional ergonomics and pressure-sensitive control have all come together in this affordable model.

The rest of the paper is organized as follows: section 2 reviews the old acquisition device and its main features. Section 3 introduces some gap of the old equipment and proposed a method to relieve these deficiencies as well as some new features accompanying the new system. The global system including the lognormal parameter extraction will also be presented. Section 4 presents numerical results and performance comparisons with the previous system. Section 5 summarizes the main contributions of this paper and possible future research.

II. DESCRIPTION OF THE PREVIOUS SYSTEM

Fig.1 represents the bloc diagram of the old acquisition system. It is composed of the homemade software Sign@medic, which managed the synchronization between the tablet digitizer and the stimulator. A Wacom Intuos2 is used as the tablet digitizer to record the coordinates of the pen tip movement, while the subject is executing a handwriting movement. This tablet records the data at a sampling rate of 200 Hz with a spatial resolution of 100 lines per millimetre. Each movement performed on the tablet follows a test protocol based on different neuromuscular tests [3]. Guide sheets are placed under the transparent plastic fold of the tablet to reduce friction and indicate the starting point and the target zone to subjects. The beginning of each test is given by the stimulator, a proprietary design. Two types of stimulus are used: a visual and an audio. The visual stimulus is displayed by a matrix of 8x10 light-emitting diodes (LED) with a delay of 1 ms [3]. The auditory stimulus is a 1 KHz beep with duration of 500 ms. The stimulator is linked to the computer by a serial to parallel converters. As it can be seen in Fig.1, the system takes a handwriting movement as an input and return information about that movement, including the Cartesian coordinates of the stiletto, the pressure, etc. Despite its effectiveness, this system...
suffers from compatibility with new Windows operating system.

Fig. 1. The block Diagram of the old acquisition system

The parallel port used by the stimulus generator is outdated, so the stimulus device cannot be connected to modern computers anymore. Moreover, the stimulus generator is cumbersome. Furthermore and the Wacom Intuos2 tablet is no longer supported by the manufacturer. These problems point the need to update the old system so it would become compatible with modern computers.

III. DESCRIPTION OF THE NEW SYSTEM

A. Transfer Process

The goal of this project is to update the old acquisition system such that it will be ergonomic and compatible with the Windows 10 operating system. To achieve this goal, we have designed a graphical interface that displays the different functionality of the visual stimulator on a Wacom Cintiq 13HD tablet. In addition, the various neuromuscular tests present in our database have to be included. The interface is implemented in C++ using the Microsoft foundation class library and the communication is done by our custom software employing Wacom drivers and libraries. The interface is divided in three sections, two sides sections displaying the visual stimulator and a middle section used to display the guide sheet and the stroke. Since the Cintiq does not have a transparent plastic fold as the Intuos2, the guide sheets were resized to fit into the active area of the Cintiq 13HD. Those guide sheets were included directly as a C++ function in the software. A new button is added in the interface to switch between right-handed and left-handed guide sheets. Finally, a new feature exploiting the features offered by the Cintiq 13HD is added to the software. It consists of showing, in real time, on the screen of the tablet the stroke being executed by a subject.

B. Description of the acquisition system

As presented in Fig. 2, the new acquisition system is a compact Update version of the old one. This system includes the stimulus generator in the Sign@medic software and will use the Wacom Cintiq 13HD as the stimulator tools. This make the overall system composed of only a computer containing the homemade software, and the Wacom Cintiq 13HD tablet. In addition, the tablet screen acts as an additional monitor for the computer. The visual stimulus is displayed on the tablet, at a response rate of 25 ms. In regard of the audio stimulus, the speakers connected to the computer emits an audio cue at 1 KHz during 500 ms. The stroke made by the participant is displayed on the tablet, so that the operator could see his movement. The Wacom Cintiq 13HD record data at 200 Hz with 200 lines per millimetre spatial resolution. The new system take as an input a handwriting movement and deliver as an output a data file including the information about the movement.

C. Description of the global system

Fig. 2 presents the global system which is composed of the new acquisition software and the lognormal parameters extractors [4] [5] [6]. Those extractors were developed by the Scríbens laboratory following the Kinematic Theory of rapid movement [7] [8] [9] [10] [11]. The global system is able to acquire data related to fast stroke movement executed by a participant. After that, it allows extraction of the lognormal parameters by using the different extractors. In conclusion, the global system takes a fast stroke movement as an input and gives lognormal parameters as an output.

IV. COMPARATIVE EVALUATION

A. Advantages and disadvantages of the new system compared to the old one

The major advantage of the new system, is that all the material everything is integrated in the software. Foremost, the guide sheets are integrated on the software and they can be changed automatically while changing the test. There is also the option to choose between right-handed and left-handed guide sheet. The stimulator generator is also integrated in the software. The device used before to generate stimulation is now outdated and it is not needed anymore. Moreover, the stroke trajectory can be seen on the screen of the tablet while being executed. This feature will help the operator to judge if the stroke is well executed. After that, the operator can decide to reject it or not. Finally, the fact that the stimulus is not displayed on the previous hardware device anymore, and only displayed on the tablet, should give more focus to the participant. Indeed, the participant will only look at the tablet screen and will not need to move their head to see what is happening with the stimulator device anymore. The participant will only look at the tablet because all they need is to execute the stroke displayed on the tablet. The new tablet has a better spatial resolution than the old one.

On the other hand, the new system presents some disadvantage as compared to the old one. The stimulator device had a 1 ms response time for displaying the visual and audio stimulus. With the new system, the tablet has response rate of 25 ms to display the visual cue. In the case of the sound cue, the response rate depends of the computer to which the tablet is connected. Knowing that human reaction to visual and auditory stimuli is respectively around 247 ms (18.54) and 228 ms (16.49) for young adults [12] [13] [14], the difference between the response rate of the new and the old system should not influence the result. The limit that it put on the experimental protocol is that the resolution for observing changes in neutomotor control, for example the changes in $t_0$ will have to be greater than 25 ms instead of 1ms to be observed.
B. Methodology to compare the two systems

To compare the two systems, we requested a subject to draw fast strokes movement for five days in a row. Each system was connected to different computers. The computer used for the new system is slower. Each day, the subject had to draw 30 valid strokes for the four following tests: visual stimulus test on the new system, auditory stimulus test on the new system, visual stimulus test on the old system and auditory stimulus test on the old system. The visual stimulus test was always executed before the audio one. Each day, we changed the order, the first day, the tests were executed on the new system before doing it on the old one. The second day, the tests were executed on the old system before doing it on the new one. The same process was done the other days. A new guide sheet was used to perform those tests. Fig. 3 and 4 respectively depict the user interface and the new guide sheet used to perform the test while the visual cue is displayed on the tablet screen. The user interface allows you to select different tests and validate the stroke performed. It also displays several pieces of information, such as the current stroke, the previous stroke, the velocity and pressure profiles.

Fig. 3. The user interface displayed on the computer monitors shown to the user

Fig. 4. The interface displayed on the screen of the tablet including the guide sheet on the center with the fast stroke, and the visual cue on each side

C. The Sigma-Lognormal model

After recording the data, the Sigma-Lognormal model was used to extract the parameters from the original data. In this purpose, Script Studio was used [5] [15] [16] [17]. This software extract 8 parameters linked with the Sigma-Lognormal model. In addition, the SNR/NbLog was calculated. The first three parameters describe the global state of the neuromotor system [8] [9] [11] [18]:

- **SNR**: The signal-to-noise ratio between the original speed profile and the reconstructed speed profile, computed in decibels (dB). This is a measure of the quality of the sigma-lognormal reconstruction. Higher SNR indicates better reconstruction.
- **NbLog**: The number of lognormal functions required to reconstruct the signal. This parameter represents fluidity of movement by the participant. Higher nbLog, indicates less fluid movement.
- **SNR/NbLog**: SNR/nbLog is used as a performance criterion. The capacity to reconstruct a movement’s speed profile with lognormals can be interpreted as an indicator of the quality of motor control since the lognormal speed profile corresponds to perfect motor control [19]. Higher SNR/nbLog indicates that the movement is closer to ideal lognormal behavior predicted by the lognormality principle.

The other parameters are indexes of the internal functioning of the neuromotor system. The four following parameters represent the neuromotor action plan:

- **t₀**: Time required (in seconds) for the brain to produce a motor command. t₀ should not to be confused with the moment a sound cue is emitted (t = 0) nor with the response time (RT) measured by moment of movement onset. t₀ is the moment when a population of neurons sends a motor command. In other words, t₀ occurs after the start cue, given that the central nervous system must perceive the sound cue before sending the command, but before RT, given that a delay is necessary for the motor command to be transmitted and for enough muscle cells to be recruited to begin the movement effectively [18]. Longer t₀ are indicative of greater delays before a motor command is emitted and, therefore, longer time to perceive the stimulus and prepare the command. t₀ is the time required for the first motor command to be produced by the brain.
- **D**: The amplitude of the movement associated with each motor command, in millimeters. Greater D indicates greater amplitude of the movement associated with a command. D is the total amplitude of the movement.

The two following parameters are intrinsic characteristics of each lognormal and related to motor program action plan [11] [20].

- **θs**: The starting angular direction of each stroke in radian.
- **θe**: The ending angular direction of each stroke in radian.

The other parameters are related to motor program execution:

- **μ**: Logtime delay, the time required to reach the middle of movement distance on a logarithmic scale. It can be associated with the general response speed of the neuromuscular system to a command.
• σ: Logresponse time, the response time on a logarithmic scale of the neuromuscular system’s response to a command. Sigma (σ) is used to compute an estimate of movement duration, it also reflects the asymmetry of the lognormal.

D. Statistical analysis

After extracting the parameters, the data for each day were regrouped together by test. Indeed, all the data recorded with the new system using the visual stimulus were regrouped together by test. Indeed, all the data recorded with the new system using the visual stimulus were regrouped together, and so forth. A t-test for independent measures was conducted on the parameters extracted by Script Studio from all the strokes made. Bonferroni correction for multiple comparisons was used to determine if there is a significant difference. P-value has to be less than 0.00625 (α= 0.05/8 = 0.00625) to have a statistical significant difference between the two groups.

E. Preliminary results and Discussion

This preliminary test showed that there is a significant difference, with the Bonferroni correction, on the SNR between the two systems in both visual and audio stimulus. In both cases, the SNR mean is higher for the new system. This could be caused by the higher spatial resolution of the new Wacom Cintiq 13HD as compared to the Wacom Intuos2. The Dtot showed a significant difference, without the Bonferroni correction, between the two systems in both visual and audio stimulus. In both cases, the Dtot mean is higher for the new system. The subject stipulates that the Wacom Cintiq 13HD has a surface with less friction than the surface of the Wacom Intuos2. Drawing on the new tablet is smoother than the old one, so the subject have more difficulties to stop their arm in fine motor control. Further investigations are under way to clarify this point.

TABLE I. RESULT OF T-TEST FOR INDEPENDENT GROUP BETWEEN THE NEW AND THE OLD SYSTEM WITH THE VISUAL CUE

<table>
<thead>
<tr>
<th>Parameters</th>
<th>New system M(S.D)</th>
<th>Old system M(S.D)</th>
<th>t-value</th>
<th>df</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNR</td>
<td>30.44 (2.46)</td>
<td>29.61 (2.67)</td>
<td>2.87</td>
<td>309</td>
<td>0.0044*</td>
</tr>
<tr>
<td>NbLog</td>
<td>2.06 (0.36)</td>
<td>2.00 (0.63)</td>
<td>0.99</td>
<td>309</td>
<td>0.3225</td>
</tr>
<tr>
<td>SNR/NbLog</td>
<td>15.25 (3.15)</td>
<td>16.37 (5.48)</td>
<td>-2.20</td>
<td>309</td>
<td>0.0283*</td>
</tr>
<tr>
<td>t0</td>
<td>0.24 (0.16)</td>
<td>0.23 (0.16)</td>
<td>0.31</td>
<td>629</td>
<td>0.7592</td>
</tr>
<tr>
<td>t1</td>
<td>0.22 (0.07)</td>
<td>0.21 (0.10)</td>
<td>0.59</td>
<td>309</td>
<td>0.5564</td>
</tr>
<tr>
<td>D</td>
<td>118.18 (95.89)</td>
<td>165.24 (94.35)</td>
<td>0.39</td>
<td>629</td>
<td>0.6973</td>
</tr>
<tr>
<td>Dtot</td>
<td>192.92 (8.59)</td>
<td>191.02 (8.08)</td>
<td>2.01</td>
<td>309</td>
<td>0.0455*</td>
</tr>
<tr>
<td>mu</td>
<td>-1.54 (0.54)</td>
<td>-1.54 (0.56)</td>
<td>0.04</td>
<td>629</td>
<td>0.9721</td>
</tr>
<tr>
<td>sigma</td>
<td>0.24 (0.14)</td>
<td>0.23 (0.15)</td>
<td>0.84</td>
<td>629</td>
<td>0.4002</td>
</tr>
<tr>
<td>tetaD</td>
<td>-0.31 (4.58)</td>
<td>-0.73 (4.64)</td>
<td>1.15</td>
<td>629</td>
<td>0.2506</td>
</tr>
<tr>
<td>tetaf</td>
<td>-0.60 (4.58)</td>
<td>-1.08 (4.64)</td>
<td>1.30</td>
<td>629</td>
<td>0.1930</td>
</tr>
</tbody>
</table>

Significant at p < 0.05

TABLE II. RESULT OF T-TEST FOR INDEPENDENT GROUP BETWEEN THE NEW AND THE OLD SYSTEM WITH THE SOUND CUE

<table>
<thead>
<tr>
<th>Parameters</th>
<th>New system M(S.D)</th>
<th>Old system M(S.D)</th>
<th>t-value</th>
<th>df</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNR</td>
<td>30.99 (2.64)</td>
<td>29.19 (2.38)</td>
<td>6.28</td>
<td>306</td>
<td>0.0000*</td>
</tr>
<tr>
<td>NbLog</td>
<td>2.01 (0.41)</td>
<td>2.03 (0.53)</td>
<td>-0.37</td>
<td>306</td>
<td>0.7106</td>
</tr>
<tr>
<td>SNR/NbLog</td>
<td>16.06 (3.73)</td>
<td>15.35 (4.30)</td>
<td>1.55</td>
<td>306</td>
<td>0.1212</td>
</tr>
<tr>
<td>t0</td>
<td>0.22 (0.14)</td>
<td>0.18 (0.14)</td>
<td>3.64</td>
<td>621</td>
<td>0.0003*</td>
</tr>
<tr>
<td>t1</td>
<td>0.17 (0.06)</td>
<td>0.14 (0.07)</td>
<td>4.06</td>
<td>306</td>
<td>0.0001*</td>
</tr>
<tr>
<td>D</td>
<td>118.71 (96.14)</td>
<td>115.27 (94.71)</td>
<td>0.45</td>
<td>621</td>
<td>0.6528</td>
</tr>
<tr>
<td>Dtot</td>
<td>194.15 (9.03)</td>
<td>190.46 (9.12)</td>
<td>3.57</td>
<td>306</td>
<td>0.0004*</td>
</tr>
<tr>
<td>mu</td>
<td>-1.62 (0.51)</td>
<td>-1.66 (0.57)</td>
<td>0.97</td>
<td>621</td>
<td>0.3307</td>
</tr>
<tr>
<td>sigma</td>
<td>0.25 (0.12)</td>
<td>0.25 (0.14)</td>
<td>-0.54</td>
<td>621</td>
<td>0.5928</td>
</tr>
<tr>
<td>tetaD</td>
<td>-0.94 (4.29)</td>
<td>-0.01 (4.40)</td>
<td>-2.68</td>
<td>621</td>
<td>0.0075*</td>
</tr>
<tr>
<td>tetaf</td>
<td>-1.26 (4.29)</td>
<td>-0.30 (4.40)</td>
<td>-2.76</td>
<td>621</td>
<td>0.0060a</td>
</tr>
</tbody>
</table>

a Significant at p < 0.00625

V. CONCLUSION

This project consisted of transferring the old Sign@medic system based on a Wacom intuos2 tablet under Windows XP operating system to the new Wacom Cintiq 13HD to operate under windows 10. The new system has the advantage of including the external stimulus generator device and the guide sheets in the software. This make the system more compact, ergonomics and user-friendly. On the other side, the new system may be slower than the old one. Eventually for more concrete result, the two system will be tested by a drawing robot to remove the human impact. Finally, the proposed system can be used in different research projects dealing with the human neuromotor system. The system is also available for non-commercial use after signing non-disclosure agreements.

ACKNOWLEDGEMENTS

This project was supported by NSERC grant RGPIN-2015-06409 to Réjean Plamondon and MESI and Univar support to Scribens laboratory.

REFERENCES


ACKNOWLEDGEMENTS

This project was supported by NSERC grant RGPIN-2015-06409 to Réjean Plamondon and MESI and Univar support to Scribens laboratory.


Gesture Synthesis for Human-Computer Interaction

Luis A. Leiva
Sciling, SL
Valencia, Spain
name@sciling.com

Daniel Martín-Albo
Wiris Math, SL
Barcelona, Spain
dmas@wiris.com

Réjean Plamondon
École Polytechnique de Montréal
Montréal, QC, Canada
rejean.plamondon@polymtl.ca

Abstract—Gesture recognizers require a large pool of training data to achieve good accuracy. However, recruiting participants, data collection and labeling, etc. necessary for achieving this goal are usually time-consuming and expensive. Fortunately, the Kinematic Theory allows to easily bootstrap gesture generation. In this paper, we show that the synthesized gestures not only perform equally similar to gestures generated by human users but also they look and feel the same. Ultimately, this work benefits researchers and designers who wish to prototype gesture-driven applications.

Index Terms—Gesture Synthesis; Bootstrapping; Gesture Recognition; Strokes; Marks; Symbols; Unistrokes; Multistrokes; Multitouch; Kinematics; User Interfaces; Rapid Prototyping

I. INTRODUCTION

Gestures are increasingly becoming a predominant input modality in today’s graphical user interfaces (GUIs). Gesture interaction is possibly one of the most researched areas in Human-Computer Interaction (HCI), with a long history that started as early as 1960, with the Sketchpad project [40] and the RAND tablet [10]. Gestures can be mid-air (more prominent in gaming applications) or stroke based (more prominent in mobile applications). We are particularly interested in the latter type, since stroke gestures are becoming more and more relevant to mainstream products such as touchscreen-capable devices like smartphones and tablets.

Stroke gestures represent the movement trajectory of one or more contact points on a sensitive surface. Stroke gestures are sometimes also called “pen gestures”, “hand drawn marks”, “hand drawn gestures”, “hand markings”, or “markings” [45]. Stroke gestures tend to give richer perceptual cues to the user, to form an association between the shape of the gesture and the meaning of the command [5]. Stroke gestures also may improve the usability of UIs, by replacing standard shortcuts by more accessible triggers.

Today, stroke gestures are mostly used in consumer devices for executing simple actions, such as pinching a picture to zoom in/out, swiping to reveal an options menu, or panning to switch between apps. Nevertheless, stroke gestures are increasingly being incorporated to facilitate random access to smartphone contents, such as invoking a command hidden in an advanced settings menu or quickly searching for a friend’s email in the contacts list. Therefore, it is expected that stroke gestures will make a notable impact in consumers’ lives.

This paper agglutinates ideas and results previously published by the authors at scientific journals and conferences [21], [22], [23], [27].

In general, any application that is driven by gestures must rely on some recognition-based techniques. These techniques often require expert knowledge in pattern recognition or machine learning, something that is typically beyond the reach of many developers and UI designers. Furthermore, recruiting participants, data collection and labeling, etc. necessary for using these techniques are usually time-consuming and expensive. Thus, it is important to investigate how to empower developers to (1) quickly collect gesture samples and (2) create recognizers; both for improving GUI usage and user experience.

II. RELATED RESEARCH

We review core areas that resemble the most to our work: approaches to gesture recognition and gesture bootstrapping.

A. Gesture Recognition

Gesture recognition has its own roots in sketching and handwriting recognition [9], [11], [29], [38]. In HCI, most gesture recognizers for prototyping GUIs are based on the template matching (or instance-based) approach [20]: a query gesture is geometrically compared against a number of stored templates, using 1 nearest-neighbor for classification and either Euclidean distance or a Mean Square Error (MSE) score as dissimilarity measures. Template matchers are a very viable and a relatively simple solution for recognizing gestures, and can be adapted to personalized user gestures.

Popular examples of these template-based recognizers among the HCI literature are part of the so-called $ family: $1 [44], $N [3], and their newer versions Protractor [24] and $N-Protractor [4], respectively. More recently, Vatavu [42] introduced $P, a sequential-agnostic recognizer where strokes are treated as a cloud of 2D points, discarding thus stroke number, order, and direction.

B. Gesture Boostrapping

Example-based approaches like GRANDMA [38], Agate [18], or Gesture Studio [26] allow developers to create and test gestures by recording examples. There are a number of similar systems tailoring end-users, like EventHurdle [16], A CAPpella [12], or GestIT [39]. They support designers’ explorative prototyping through programming by demonstration environments. Another strand of research is aimed at simplifying the process of designing gesture sets. For example, Gesture Script [25], Gesture
Mark [31], Gestalt [32], or CrowdLearner [2]. Finally, we can find a number of competing systems aimed at creating synthetic 3D gestures as a means to improve gesture recognizers, including e.g. MAGIC [6], [17] and Gesture Follower [8].

Overall, training data is the key factor to build a competitive gesture recognizer, for which most of the previously reviewed approaches have contributed to generate their own, without having to recruit participants and perform time-consuming user evaluations. They also have contributed to decreasing the number of iterations needed to build a fast and stable gesture recognition interface. However, there is no evidence that any of the previous works can produce human-like samples. Further, artificially generated samples usually perform poorly since they do not illustrate sufficient variation required for high-quality training [1], [36], [37].

**III. SYNTHESIZING GESTURES**

Many models have been proposed to study human movement production; e.g. [7], [41], [30], [43], among which the Kinematic Theory [33] provides a well-established and solid framework for the study of the production of human movements. This framework takes into account different psychophysiological features, such as the neuromuscular response time, and has been shown to outperform many other approaches [34]. The Sigma-Lognormal (ΣΛ) model [35] is the latest instantiation of this framework, and very recently has been used to explore gesture recognition.

At a high-level representation, the Kinematic Theory assumes that a complex handwritten trace (such as a stroke gesture) is composed of a series of primitives (circular arcs) connecting a sequence of virtual targets. This series of primitives conform the “action plan” of the user, which is fed through the neuromuscular network to produce a trajectory that leaves a handwritten trace.

Under this framework, each gesture primitive is modeled according to a lognormal function of their velocity profile, defined by a set of central parameters \((D, l_0, \theta)\) and peripheral parameters \((\mu, \sigma)\) [33]. Then, an extractor computes the parameter values that best explain the observed velocity profiles [28]. Once the gesture primitives are modeled, perturbations can be added to the model parameters in order to produce different gesture variations [22]:

\[ p_i^* = p_i + n_{p_i} \]  

where \(p_i = \{ \mu_i, \sigma_i, D_i, \theta_i \} \) denote the ΣΛ parameters, with \(n_{p_i} = \mathcal{U}(-n_i, n_i)\) being the noise applied to each primitive, according to a uniform distribution (i.e., a rectangular distribution with constant probability) centered around the expected human variability ranges [15], [22]: \(n_{\mu} = n_{\sigma} = 0.1\), \(n_D = 0.15\), \(n_{\theta} = 0.06\). Figure 1 shows some examples of the synthetic gestures produced with the Kinematic Theory.

Previous works have demonstrated the connection between the distortion of the Sigma-Lognormal parameters and the intra-variability found in human handwriting [13]. Combining both types of variations reflects real-life situations like performing the same movement under different psychophysiological conditions. For example, perturbations in \(\mu\) and \(\sigma\) mimic peripheral noise, e.g., a user who articulates the same gesture slightly different each time; perturbations in \(D\) and \(\theta\) refer to central fluctuations that occur in the position of the virtual targets of the action plan from one articulation to the next [21], [22].

**IV. GESTURE PERFORMANCE ANALYSIS**

We compared the performance of synthetic gestures with that of human samples under user-independent tests in terms of articulation speed, input device, and gesture variability. The interested reader may consult user-dependent tests and a follow-up evaluation in our previous work [21].

We synthesized two popular datasets in HCI: GDS [44] and MMG [4]. On the one hand, the GDS dataset comprises 5,280 unistroke gestures (16 classes). Ten users provided 10 samples per class at 3 articulation speeds (slow, medium, fast) using an iPAQ Pocket PC (stylus as input device). On the other hand, the GDS dataset comprises 5,280 multistroke gestures (16 classes). Twenty users provided 10 samples per class at 3 articulation speeds (same as in GDS) using either finger (half of the users) or stylus as input device on a Tablet PC.

**A. Impact of Articulation Speed**

We sought to analyze whether gesture articulation speed leads to a difference in classification error rates between human and synthetic templates. The GDS dataset was analyzed with the S1 recognizer, whereas MMG was analyzed with the S1 recognizer. Both recognizers were fed with 10 templates. Table I summarizes this experiment. A two-tailed paired t-test (Bonferroni corrected) revealed no statistically significant differences for any of the articulation speeds, suggesting thus that synthetic gestures perform the same as their human counterparts.

<table>
<thead>
<tr>
<th>Type</th>
<th>GDS</th>
<th>MMG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human</td>
<td>0.09</td>
<td>0.06</td>
</tr>
<tr>
<td>Synthetic</td>
<td>0.20</td>
<td>0.27</td>
</tr>
</tbody>
</table>

Fig. 1: Examples of gestures synthetized with the Kinematic Theory, using a single human example as input.
B. Impact of Input Device

We also sought to analyze whether the input device leads to a difference in classification error rates between human and synthetic templates. This analysis was performed over the MMG dataset, which is the one that provides two data splits: finger and stylus. We used the SP recognizer with 10 templates. Table II summarizes this experiment. A two-tailed paired t-test (Bonferroni corrected) revealed no statistically significant differences, suggesting thus that gestures can be successfully synthesized with both a stylus and the finger.

<table>
<thead>
<tr>
<th>TABLE II: Effect of input device on error rates (in %)</th>
<th>Human</th>
<th>Synthetic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finger-Synthetic</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Finger-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.03</td>
<td>0.04</td>
<td>0.43</td>
</tr>
<tr>
<td>Stylus-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.29</td>
<td>0.29</td>
<td>0.49</td>
</tr>
</tbody>
</table>

C. Impact of Gesture Variability

Finally, we sought to analyze whether an increase in the amount of noise $\xi$ introduced to the Sigma-Lognormal model parameters leads to more variable synthetic gestures. We computed the mean squared error between human and synthetic gestures for different number of synthesized samples using $\xi$ from 0.0 (no variability) to 1.0 (maximum variability, in the allotted human ranges [21]). Table III summarizes this experiment. As expected, it was found that synthetic samples are more variable as $\xi$ increases. Interestingly, variability was found to increase as the number of requested synthetic samples increases.

<table>
<thead>
<tr>
<th>TABLE III: Gesture variability (mean squared error)</th>
<th>N</th>
<th>GDS</th>
<th>MMG</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\xi = 0.0$</td>
<td>$\xi = 1.0$</td>
<td>$\xi = 0.0$</td>
<td>$\xi = 1.0$</td>
</tr>
<tr>
<td>10</td>
<td>554.9</td>
<td>593.6</td>
<td>169.7</td>
</tr>
<tr>
<td>100</td>
<td>&quot;</td>
<td>622.1</td>
<td>&quot;</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;</td>
<td>621.4</td>
<td>&quot;</td>
</tr>
</tbody>
</table>

V. GESTURE SIMILARITY ANALYSIS

To provide further evidence on the value of the Kinematic Theory as a means to generate stroke gestures, we conducted an online survey that measured the user perception toward gestures’ human-likeness. We used the same datasets depicted in the previous section, both in their original and synthesized form. The survey is still available online at https://g3.prlht.upv.es/guessit/. Eventually, 236 participants took part in this study.

We defined the guessing accuracy as the user’s ability to distinguish between human and synthetic samples; i.e., the proportion of gestures that were successfully classified by the user. Then, two types of errors can be committed [14], [15]: (i) a synthetic gesture is mistaken with a real sample, measured by the False Real Rate (Type I error); and (ii) a real gesture is marked as synthetic, measured by the False Synthetic Rate (Type II error). The results are presented in Table IV.

<table>
<thead>
<tr>
<th>TABLE IV: Accuracy and error rates (in %)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>49.65</td>
</tr>
</tbody>
</table>

A paired two-sample t-test (two-tailed alternative hypothesis) revealed that there is no difference between classified and misclassified gestures, proving thus the human-like appearance of synthetic samples. In sum, participants could not tell human and synthetic gestures apart. Follow-up analyses [22], [19] provided further evidence that synthesized gestures are actually reflective of how users produce stroke gestures. We concluded therefore that the visual appearance of the synthetic samples is very similar and close to that of human gestures.

VI. DISCUSSION

Users tend to be reluctant to invest time and effort upfront to train or adjust software before using it [5]. Further, users are unwilling to provide more than a small set of samples for training [24]. Consequently, synthesizing techniques like are of high value, as they help to lower time and costs associated to recruiting users and subsequent data labeling.

Until now the “human likeness” of synthesized gestures was measured indirectly, intermediated by classification/recognition accuracy performance. Our studies are important because recent research has shown how different users produce different gesture articulations in various conditions.
Yet, finally researchers and practitioners can be confident that synthesized gestures using the Kinematic Theory are actually reflective of how users produce stroke gestures.

VII. CONCLUSION

We have shown that the Kinematic Theory generates stroke gestures that can be useful to researchers and practitioners in many ways. The synthesized gestures not only perform equally similar to their human counterparts but also they look and feel the same. In sum, the Kinematic Theory provides the HCI community with a reliable way to synthesize gesture sets without having to expressly collect them from a large pool human subjects. However, we do not to encourage the substitution of human gestures by synthetic ones, but rather to provide an automated way to lower the need of recruiting a large number of users and subsequent data labeling. Our online application and accompanying web service (JSON RESTful API) is available at https://g3.prhlt.upv.es/.

REFERENCES

Kinematic Reconstruction of Calligraphic Traces from Shape Features

1st Daniel Berio
Department of Computing
Goldsmiths, University of London
London, United Kingdom
d.berio@gold.ac.uk

2nd Frédéric Fol Leymarie
Department of Computing
Goldsmiths, University of London
London, United Kingdom
ffl@gold.ac.uk

3rd Réjean Plamondon
Department of Electrical Engineering
Polytechnique Montréal
Montréal, Québec, Canada
rejean.plamondon@polymtl.ca

Abstract—Our goal is to be able to reproduce computationally calligraphic traces, e.g. as found in the art practices of graffiti and various forms of more traditional calligraphy, while mimicking the production process of such art forms. We design our user interfaces in a procedural generation and computer aided design (CAD) setting. As a result, we seek to seamlessly work between data used in design packages (without kinematics) and data easily digitised by users (e.g. online, with kinematics). To achieve these goals, we propose a method that allows to reconstruct kinematics from solely the geometric trace of handwritten trace in the form of parameters of the Sigma-Lognormal model. We purposely ignore the kinematics possibly embedded in the data in order to treat online data and vector patterns with the same procedure.

Index Terms—graphonomics, graffiti, calligraphy, curvature, symmetry, kinematic theory, sigma-lognormal

I. INTRODUCTION

Many handwriting analysis methods rely on a prior segmentation of the handwriting trace into constituent primitives or strokes. Some methods exploit the kinematics of the movement and segment the trajectory in correspondence with minima of velocity. Accordingly with the stereotypical inverse relation between speed and curvature of handwriting traces, other methods rely on the identification of curvature extrema along the pen-trace. This fits with a modelization of handwriting in the form of ballistic stroke primitives, in which curvature extrema will typically correspond with velocity minima and are indicative of the initiation of a new stroke.

A robust identification of curvature extrema can be difficult, since curvature is a second order differential quantity which tends to amplify the effects of noise in the input as an outcome of the digitisation process. One popular method to overcome this problem is to first smooth the digitised signal using a filter (e.g. convolving with a Gaussian) or interpolating with some analytic function (e.g. smoothing splines). However, smoothing risks to remove (perceptually) important features of an outline and choosing reasonable parameters remains a difficult task. To overcome this fundamental issue, one possible avenue is to generate an intermediate scale-space in which features are identified and tracked at different scales [1]. Such a scale-space is very often produced by iterative Gaussian filtering in the spatial domain, or via the frequency domain using wavelets [2]. An alternative to such traditional filtering (which tends to blur away details) is to use a structural notion of scale, e.g. by associating a support metric along a contour with each curvature feature being tracked, such as when performing morphological operations on the curvature function [3]. Nevertheless, such methods operating directly on the curvature function (along a contour) suffer from poor localisation, and do not capture well singularities which can be perceptually significant, such as curvature breaks (typical of corner features). Also, such methods are usually developed and implemented for closed contours, which is not typical of handwritten traces, and further do not behave well with loops or other singular behaviours which are common in calligraphy.

An alternative to working directly with the curvature function is to exploit the correspondence existing between symmetry axes and the curvature behavior of a contour [4]. Originally pioneered by Harry Blum in the 1960’s for the study of biological shape [5], the Symmetry Axis Transform (SAT) — also known as Medial Axis or simply skeleton for closed contours — is a shape representation that provides a bridge between geometry and topology. The SAT is commonly viewed as the set of centers of “maximally inscribed” disks, or with the “prairie grassfire” or wave-front analogy, in which the symmetry axes are given by the “quench” points at which fire fronts or waves propagating from the object boundary meet and stop expanding. Contrary to a common misinterpretation, the SAT is not only defined for closed shapes, but is valid also for open contours or even point samplings [6] — in the latter case becoming similar to the Voronoi graph very popular in computational geometry and computer aided design (CAD).

In this paper, as a starting point, we make use of the duality between curvature and symmetry axes [4] in order to extract more robustly curvature features, such as extrema along a handwriting or drawing trace. The method is also directly adaptable to open contours, to contours with breaks in curvature, and can further be used to identify loops. Each feature is also explicitly paired with corresponding contact circles and a support region — where curvature is approximately monotonic. Given such a robust and rich feature description of a handwritten trace, we show how to exploit this spatial and structural geometric representation to infer the kinematics...
of a likely generative movement. To do so, we rely on the Kinematic Theory of Rapid Human Movements [7], [8], a family of models of reaching and handwriting motions, in which a movement is described as the result of the parallel and hierarchical interaction of a large number of coupled neuromuscular components.

The resulting method allows the reconstruction of physiologically plausible velocity profiles for the geometric trace of an input movement given as an ordered sequence of points. While state of the art methods exist for the parameter reconstruction of Kinematic Theory based models from digitised traces of handwriting [8]–[10], we design our method with the goal of targeting applications in both graphonemics and CAD/computer graphics. As a result, we purposely ignore the kinematics of the input in order to seamlessly handle online handwriting with arbitrary sampling quality as well as vector art in which only the sequential ordering of points may be available. We also choose this approach with the future aim of combining our method with one that recovers temporal traces of handwriting [8]–[10], we design our method with the construction of Kinematic Theory based models from digitised neuromuscular components.

On the basis of the Kinematic Theory [7], we describe the kinematics of a handwriting/drawing movement via the vectorial superimposition of $N$ time shifted stroke primitives. The speed of each stroke is given by a lognormal

$$\Lambda_i(t) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{(\log(t-t_{0i}) - \mu_i)^2}{2\sigma_i^2}\right)$$

which describes impulse response of each stroke to a centrally generated command occurring at time $t_{0i}$. The parameters $\mu_i$ and $\sigma_i$ respectively describe the stroke delay and response time in a logarithmic time scale, and determine the shape and asymmetry of the lognormal. With the assumption that handwriting movements are made with rotations of the elbow or wrist, the curvilinear evolution of a stroke can be described by a circular arc. The angular evolution of a stroke is described by using the time integral of eq. 1:

$$\phi_i(t) = \theta_i - \frac{\delta_i}{2} + \delta_i \int_0^t \Lambda_i(u)du$$

$$= \theta_i - \frac{\delta_i}{2} + \frac{\delta_i}{2} \left[ 1 + \text{erf} \left( \frac{\log(t-t_{0i}) - \mu_i}{\sigma_i \sqrt{2}} \right) \right],$$

where $\theta_i$ is the direction of the stroke and $\delta_i$ is the *stroke curvature* parameter which determines the internal angle of the circular arc. The planar pen-tip velocity is then calculated with:

$$\dot{x} = \sum_{i=1}^{N} \hat{D}_i \Lambda_i(t) \cos(\phi_i(t)) \quad \text{and} \quad \dot{y} = \sum_{i=1}^{N} \hat{D}_i \Lambda_i(t) \sin(\phi_i(t))$$

where $\hat{D}_i = D_i h(\theta_i)$ is the length $D_i$ of the stroke scaled by

$$h(\theta_i) = \left\{ \begin{array}{ll} 2\phi_i &\text{if } |\sin\theta_i| > 0, \\ 1 &\text{otherwise,} \end{array} \right.$$ 

which compensates for the stroke curvature based on the ratio between the perimeter and the chord length of a circular arc. The acceleration components of the lognormal trajectory are then given by [12]:

$$\ddot{x} = \sum_{i=1}^{N} \hat{D}_i \dot{\Lambda}_i(t) \cos(\phi_i(t)) - \hat{D}_i \delta_i \Lambda_i^2(t) \sin(\phi_i(t))$$

$$\ddot{y} = \sum_{i=1}^{N} \hat{D}_i \dot{\Lambda}_i(t) \sin(\phi_i(t)) + \hat{D}_i \delta_i \Lambda_i^2(t) \cos(\phi_i(t)).$$

with

$$\dot{\Lambda}_i(t) = \Lambda_i(t) \frac{\mu_i - \sigma_i^2 - \log(t-t_{0i})}{\sigma_i^2(t-t_{0i})}$$

which allows us to compute the curvature at time $t$ with the well known formula:

$$\kappa(t) = \frac{(\dot{x}\ddot{y} - \dot{y}\ddot{x})}{(\dot{x}^2 + \dot{y}^2)^{3/2}}.$$

The sequence of curvilinear strokes describes an *action plan* consisting of an initial position $p_0$ followed by a sequence of $N$ virtual targets $p_1, \ldots, p_N$ each corresponding to an imaginary aiming locus per stroke (Fig. 5). The degree of time overlap between lognormal components then defines the degree of smoothness of the trajectory in correspondence with each virtual target, where a greater time overlap results in a smoother trajectory. In order to facilitate interactive applications [13] and to simplify the subsequently described parameter reconstruction method, we compute the directions $\theta_i$ and length $D_i$ for each pair of consecutive virtual targets $(p_{i-1}, p_i)$. Furthermore, we explicitly define the time overlap of each lognormal through an intermediate parameter $\Delta t_i \in [0,1]$ where $t_{0i} = t_{0i-1} + \Delta t_i \sinh(3\sigma_i)$ if $i > 1$ and $t_{01} = 0$.

III. Trajectory Segmentation and Analysis

The proposed trajectory reconstruction method exploits a prior feature analysis of the input which is based on the SAT together with the Symmetry Curvature Duality (SCD) theorem [4], a result presented by Leyton which links the symmetry axes of a shape to its curvature extrema and states [4]:

*Any segment of a smooth planar curve, bounded by two consecutive curvature extrema of the same type, has a unique symmetry axis, and the axis terminates at the curvature extremum of the opposite type.*

Following the SCD theorem, given a symmetry axis, it is then possible to identify a curvature extrema at one end (tip).
In a traditional setting, the SAT is computed at once for a given final contour or written trace (in our case). This however does not allow to identify all perceptually significant curvature extrema, as part of a contour may forbid the existence of a symmetry axis that would otherwise end near a curvature extremum.\textsuperscript{1} Belyaev and Yoshizawa \cite{belyaev2015} have recently developed an alternative way to compute symmetry axes which avoids this masking effect.\textsuperscript{2} In summary, a symmetry axis is evaluated as one travels along a written trace. Once a significant axis is found, its existence ends once another significant axis emerges, and the previous written trace already traversed is “forgotten” (so as to not mask other potential extrema of curvature associated to later parts of the trace).

\section{Feature analysis}

The input to our method is a curve $z(s)$ parameterised by arc length $s$ and with total arc length $l$. For each curvature extrema, the feature segmentation outputs a sequence of disks $\{\Omega_i\}$ centered at $c(\Omega_i)$, with radius $r(\Omega_i)$ and corresponding with the circle of curvature osculating the extrema. Each disk has contact with the input curve along a region $P(\Omega_i)$ at which curvature is approximately constant, which we refer to as the disk’s projection and is delimited between two anchor points defined by their respective arc lengths $s_0(\Omega_i)$ and $s_1(\Omega_i)$. The location of the curvature curvature extrema $s(\Omega_i)$ is then given by the midpoint of the projection $s_0(\Omega_i) + s_1(\Omega_i))/2$. The curvature of the extrema is then simply $\kappa(s(\Omega_i)) = \pm 1/r(\Omega_i)$, where the sign is computed using the signed area of the triangle $[z(s_0(\Omega_i)), c(\Omega_i), z(s_1(\Omega_i))]$.

\section{Trajectory segmentation}

The symmetry axis extraction process results in the identification of a set of curvature extrema, where each extrema is paired with the corresponding osculating circle of curvature. The projection at which the disk has contact with the curve describes a trajectory segment where curvature is approximately constant. Given the estimate of curvature extrema we proceed with the evaluation of a curvilinear profile for the trajectory by fitting Euler spirals to contour segments bounded by consecutive curvature extrema (Section III-B1). Each Euler spiral segment is then decomposed into one or two circular arc segments, depending on the presence or not of an inflection (Section III-B2). This results in a sequence of circular arc segments that are then used in the subsequent estimation of $\Sigma \lambda$ curvature parameters $\{\lambda_i\}$ (Section IV).

1) Euler spiral fitting: Euler spirals (also known as Cornu spirals, or clothoids) \cite{whitney2005} are an interesting type of curve in which curvature varies linearly with arc length, permitting the description of variably curved segments which may contain one inflection (Fig. 1). An Euler spiral is commonly parameterized by arc length $s$ using the cosine $(C(s))$ and sine $(S(s))$ Fresnel integrals:

$$C(s) + iS(s) = \int_{0}^{s} e^{it^2/2} du.$$  \hfill (10)

The curve is then defined between an initial $(s_{0})$ and final $(s_{1})$ parameter and can be conveniently computed in an efficient manner using an approximation method developed by Heald in \cite{heald1970}.

In order to fit an Euler spiral, we first compute approximate tangent directions along the trajectory in correspondence with the initial and final points of the segment under examination. This allows to compute a first estimate of the spiral’s initial and final parameters rapidly using a secant method described in \cite{whitney2005}. On the other hand, the tangent estimate is likely not to be precise due to noise in the input so we then proceed to refine the fit with a least squares optimisation using the Gauss-Newton method. In order to do so we parameterise a spiral segment scaled by a factor $\alpha$ and rotated by an amount $\omega$ with:

$$p_s(s) = \begin{bmatrix} \alpha \cos(\omega) C_0(s) - \alpha \sin(\omega) S_0(s) \\ \alpha \sin(\omega) C_0(s) + \alpha \cos(\omega) S_0(s) \end{bmatrix},$$ \hfill (11)

$$C_0(s) = C(s) - C(s_{0}) \quad \text{and} \quad S_0(s) = S(s) - S(s_{0}).$$ \hfill (12)

We then transform the input curve segment so that its end points match the end points of the spiral in its canonical form with $\alpha = 1$ and $\omega = 0$, and then proceed with the minimisation

$$\min_{s_{0},s_{1},\alpha,\omega} \frac{1}{2} \sum_{i=1}^{n} \| p_s(s_0[i]) - z(s[i]) \|^2,$$ \hfill (13)

where $s_0[i]$ and $s[i]$ respectively give $n$ equally spaced points (i) sampled along the spiral $p_s(s)$ between $s_{0}$ and $s_{1}$ and (ii) along the input curve $z(s)$.

2) Inflections and Circular Arc Decomposition: Inflections are directly found by checking the signs of the two spiral parameters $s_0$ and $s_1$. If the parameters have different sign the position along the spiral at which the arc length parameter $s = 0$ determines the location of the inflection. For each inflection, we check if the ratio $\min(\|s_0[i],s_1[i]\|)/|s_1 - s_0|$ is less than a user defined threshold $\epsilon_{inf}$ (which we empirically set to 0.2 in the accompanying examples), in which case we discard it as a degenerate case (being too close to one spiral end point).

Depending on the presence of an inflection we fit either one or two circular arcs to each Euler spiral segment. The internal angle of the circular arcs is easily estimated by integrating the curvature of the spiral and distinguishing between 3 cases.
(a) For the case of two arcs the internal angles are given by $s_i|s|N$ for each parameter $s$ and $s_1$ (Fig. 2a). (b) In the case of a degenerate inflection, we use the same method to fit a single arc and choose only the parameter with the greatest absolute value and consequently higher curvature. (c) When no inflection is present the internal angle is given by $|s_i(1)|s|N|sgn(s_0)|$. For each arc, we then check if the absolute internal angle is greater than $(3/4)\pi$, in which case we subdivide the arc in two (Fig. 2b).

These final steps produce an ordered sequence of $N$ circular arcs with internal angles $\hat{s}_i|s|N$, center $c(\hat{\theta})$ and radius $r(\hat{\theta})$. The circular arcs are delimited by $N+1$ feature points $\{\hat{s}_i\}_{i=0}^N$, each defining a distance along the contour and with $\{\hat{s}_0,\hat{s}_N\}$ indicating the initial and final points. Each feature point corresponding with a curvature extrema is also paired with its corresponding osculating circle with radius $r(\hat{s}_i)$ and curvature $\kappa(\hat{s}_i)$. Thus the projection $p(\hat{s}_i)$ for each extremum also defines a circular arc, at which the curvature $\kappa(\hat{s}_i)$ is constant (Fig. 3). Extrema at which the trajectory is smoother will result in a larger radius of curvature and a larger projection. As a result, it is then possible to produce an approximate reconstruction of the original trajectory in the form of circular arc segments, similarly to the method originally proposed in [17], but with the difference that each curvature extremum is also paired with a supporting arc, which results in a more accurate reconstruction of the original trace.

Fig. 2. Decomposing Euler spirals (stippled grey) into arcs. (a) two arcs delimiting an inflection. (b) one segment with angle $>(3/4)\pi$ divided into two arcs.

Fig. 3. Feature extraction (left) followed by Euler spiral fitting (middle) and arc decomposition (right) of a sample from the UJI handwritten character dataset. The arcs in red indicate regions with approximately constant curvature corresponding with curvature extrema.

IV. ITERATIVE RECONSTRUCTION OF $\Sigma\Lambda$ PARAMETERS

Given the previous trajectory segmentation step and a number of simplifying assumptions, we have the information necessary to reconstruct the input trajectory given its geometry only, in terms of the shape features previously indentified (i.e. curvature extrema, inflections, interpolating arcs of spirals). The method is a development and improvement over our prior efforts [18], [19] in a similar direction. While a number of state of the art methods exist [8]–[10] for the accurate reconstruction of $\Sigma\Lambda$ parameters from digitised traces, these methods require the kinematics of the original trajectory. In our method, we sacrifice to some extent the accuracy of the reconstruction in order to seamlessly deal with online handwriting data as well as vector art in which only the time/sequential ordering of points may be available.

An initial estimation of the trajectory parameters is given by a virtual target sequence $p_1 = \pi(\hat{s}_i)$, stroke curvature parameters $\delta_i = \theta(\hat{s}_i)$ and time overlap parameters $\Delta t_i = 0.5$. For the sake of simplicity, we consider the remaining parameters $\sigma_i$ and $\mu_i$ as typical properties of the neuromuscular system of a writer and keep these set to a user configurable value. The initial trajectory estimate is likely to differ from the original and to possess a reduced scale due to the smoothing effect of the lognormal stroke overlap (Fig. 5a).

To improve the reconstruction, we adopt an iterative refinement scheme (Fig. 5b) in which we adjust the curvature and time overlap parameters together with the virtual target positions in order to minimise the difference between the reconstructed and original trajectories. At each iteration, we rely on the estimation of a series of feature points along the generated lognormal trajectory. We compute $N + 1$ key-points $\{\tau_i\}_{i=0}^N$ along the trajectory (Fig. 4) where $\tau_1, \ldots, \tau_{N-1}$ indicates the time occurrence at which the influence of one stroke exceeds the previous one and curvature is maximal, while $\tau_0$ and $\tau_N$ are respectively the starting and ending time of the trajectory. In addition we compute $N$ peak-points $\{\gamma_i\}_{i=1}^N$, which indicate the approximate time occurrence of the maximum speed for each stroke (Fig. 4), which for each stroke is given by the mode of the corresponding lognormal $t_0 + \exp(\mu_i - \sigma_i^2)$.

The iterative refinement scheme is based on 3 observations: Observation 1. The time parameter $\Delta t_i$ is proportional to the curvature $\kappa(\tau)$ at the time of the corresponding key-point. Thus, a higher value of $\Delta t_i$ will decrease the amount of overlap of successive lognormals. This will result in a lower speed and higher curvature $\kappa(\tau)$ at the time occurrence of the key-point. Since we have a good approximation of the curvature $\kappa(\hat{s}_i)$ in the original trajectory, the relation between the two can be exploited in order to adjust $\Delta t_i$ proportionally at each iteration. We observe that changes in $\Delta t_i$ are not linearly related to changes in the curvature $\kappa(\tau)$ at the corresponding key-point. In order to compensate for this, we assume a 1/3 power relation [20] which has been
often observed in human movement and particularly holds for elliptical portions of the trajectory [12], which is often the case near key-points. The reasoning is that given the relations
\[ \Delta t \propto \kappa \quad \text{and} \quad \Delta t \propto 1/v \]
where \( v \) denotes speed, we have the proportions relating desired and generated curvature/velocity:
\[ \rho_\kappa = \frac{\hat{\kappa}}{\kappa} \quad \text{and} \quad \rho_v = \frac{\hat{v}}{v}. \]
As a result, given the power law \( v = \kappa^{1/3} \) [20] and because velocity and \( \Delta t \) are inversely proportional, we finally get the relation:
\[ \rho_{\kappa v} = \frac{v}{\hat{v}} = (\frac{\kappa}{\hat{\kappa}})^{-1/3} = (\frac{\kappa}{\hat{\kappa}})^{1/3}. \]

Observation 2. Shifting a virtual target \( p_i \) in a given direction, will cause the generated trajectory point \( p(\tau_i) \) to move in a similar direction. As a result, shifting the virtual target \( p_i \) along the vector \( z(\hat{s}_i) - p(\tau_i) \) will decrease the distance between curvature extrema in the generated and original trajectories.

Observation 3. The distance \( D_i \) between successive virtual targets \( p_i \) and \( p_{i-1} \) will influence the curvature of the stroke. In fact, augmenting this distance will increase the radius of curvature of the circular arc defined by the parameter \( \delta_i \) and will result in a decrease of curvature for the stroke. While the trajectory tends to depart from the circular arc near the key-points at \( t = \tau_i \), due to the smoothing effect of the lognormal time overlap, it tends to pass closer to the circular arc at \( t = \gamma_i \) where the amplitude of the lognormal is maximal. As a result, we utilise this locus to evaluate the deviation from the desired arc \( \hat{\theta}_i \) and correct the parameter \( \delta_i \) accordingly.

As a result of these observations, we define each iteration of our method that consists of the following steps:
\[
\Delta t_i \left\{ \begin{array}{ll}
\Delta t_i (\kappa(\hat{s}_i)/\kappa(\tau_i))^{1/3} & \text{if } r(\hat{s}_i) \neq \infty \\
\Delta t_{\text{min}} & \text{otherwise}
\end{array} \right., \quad (14)
\]
\[
\delta_i \left\{ \begin{array}{ll}
\delta_i + \lambda_\delta (\hat{\delta}_i - \delta_i) & \text{if } r(\hat{s}_i) \neq \infty \\
\delta_i & \text{otherwise}
\end{array} \right., \quad (15)
\]
\[
p_i \left\{ \begin{array}{ll}
p_i + \lambda_p (c(\hat{s}_i) - p(\tau_i)) & \text{if } r(\hat{s}_i) \neq \infty \\
p_{i-1} & \text{otherwise}
\end{array} \right., \quad (16)
\]
where \( \lambda_\delta \) and \( \lambda_p \) are damping parameters that we experimentally tune to 0.1 and 0.5 to avoid excessive adjustments at each iteration, and
\[
\hat{\delta}_i = 4 \tan^{-1} \left( \frac{h}{a} \tan \left( \frac{\delta_i}{4} \right) \right) \quad \text{with} \quad (17)
\]
\[
a = \parallel p_i - p_{i-1} \parallel \quad \text{and} \quad (18)
\]
\[
h = \left( r(\hat{s}_i) - \parallel p(\gamma_i) - c(\hat{s}_i) \parallel \right) \text{sgn}(\hat{\delta}_i). \quad (19)
\]
The term \( h \) determines the amount to shift the curvature parameter \( \delta_i \) by comparing the radius of the circular arc \( \hat{\theta}_i \) initially fitted to the input to the distance between its center and the lognormal peak point \( p(\gamma_i) \). Note that for the case of inflection points \( r(\hat{s}_i) = \infty \), where the curvature is 0, we force \( \Delta t_i \) to a user-defined minimum value. This results in a maximal overlap between lognormal components and gives a smooth transition between strokes in the generated trajectory.

V. DISCUSSION AND APPLICATIONS

We tested the iterative refinement on different inputs ranging from online data (e.g., Graffiti Analysis database [21] and UJI handwritten character dataset [22]) to vector traces with no kinematic information, and it consistently produces visually accurate reconstructions of the input. We observe that, while fluctuations may appear during the iteration, the refinement scheme consistently and rapidly converges towards a reduction of the error between the input and the generated trajectories.
The reconstructed $\Sigma \Lambda$ parameters provide a concise and easily manipulable representation of a geometric trace. This can be exploited in a number of applications that are relevant to our desired use cases in CAD [13] and procedural content generation [19]. For example, new instances of a given trace can be generated by randomly perturbing the virtual target positions and scaling the curvature and time overlap parameters, which results in variations that evoke multiple instances of writing by the same or multiple authors (Fig. 6). Furthermore, the $\Sigma \Lambda$ parameterisation is well suited for interactive editing applications [13]. As a result a user can easily adjust the output of the reconstruction in real time using a point and click procedure that is similar to the ones traditionally used in CAD software packages.

VI. CONCLUSION

We demonstrated a simple method to reconstruct $\Sigma \Lambda$ parameters from solely the geometric trace (left by handwriting), which relies on a prior segmentation at perceptually salient points. The proposed method produces an accurate kinematic and geometric reconstruction of an input given just an ordered sequence of points. On the other hand, we consider this still as a stepping stone towards the development of more accurate and physiologically plausible methods, which exploit the same feature extraction and preprocessing steps. In future work, we intend to develop more sophisticated methods of choosing the $\Sigma \Lambda$ parameters $\mu_i$ and $\sigma_i$, which are currently experimentally set, and plan to explore in depth how the inferred kinematics relate to human data.

REFERENCES

Abstract—One of the biggest challenges in on-line signature verification is the detection of counterfeited signatures. Recently, novel schemes based on the kinematic theory of rapid human movements and its associated Sigma-Lognormal model has been proposed to improve the detection of on-line skilled forgeries. But for a more realistic and reliable estimation of the forgery detection rate, we would need more challenging on-line forgeries than those included in current databases. To get better on-line skilled forgeries, this paper aimed at leveraging the Sigma-Lognormal model to improve the skill of any online forged signature. Specifically, we propose to replace the original velocity profile of any on-line signature by a synthetic Sigma-Lognormal profile. The new profile emulates a genuine-like velocity profiles without modifying the original ballistic trajectory. Experimental results were performed with the 132 on-line users of publicly BiosecureID database. It is shown that the detection rate of forgeries is significantly worsened when the velocity profile is replaced by the synthetic one. A countermeasure to detect this kind of improved fake signatures is also proposed.

Keywords—Automatic Signature Verification, Sigma-Lognormal model, forged signatures.

I. INTRODUCTION
Biometrics have emerged as a reliable, fast and automatic identification technology. Among the different biometric traits (i.e., fingerprint, face, voice, iris, etc.), one of the most widely accepted is the signature. Even though the verification performance rates of Automatic Signature Verifiers (ASV) have reached significant ratios, skilled forgeries still remains a major challenge for those systems.

According to forensic handwriting examiner nomenclature, the spectrum of signature forgeries spans from random to simple or zero effort up to skilled or freehand specimens. Although efforts have been put in recent years to incorporate better forgeries in test database, most of these cannot be considered as fully skilled forgeries, at least in a forensic document perspective, where it refers to the action of a forger who tries to imitate, after time and practice, as closely as possible the static and dynamic information hidden in a specimen.

Recently, several papers have considered the use of the kinematic theory of rapid human movements and its associated Sigma-Lognormal model [4] to improve the forgery detection. Briefly, this theory models the velocity profile of a rapid movement, like a signature, as a weighted sum of delayed lognormals. Each of these lognormals represent a stroke, the complete movement being a composition of overlapped strokes. One of the advantage of this model is that it considers physical body features such as the neuromuscular system response for the production of a signature difficult to imitate.

In the literature we can find some articles which use the kinematic theory of rapid human movements to detect skilled forgery. For example, the lognormal parameters are combined with classical parameters to improve an ASV in [5]. In the same context, in [6] a dissimilarity measure between lognormal features are proposed. These parameters have been also used to train ASV with only one signature [7].

However, it could be said that the results reported in such papers are biased as most of the skilled forgeries available in the current databases usually reproduce the trajectory of a genuine signature accurately but the dynamic of the signature is poorly imitated [2]. It arguably explains the poor performance of off-line ASV against skilled forgeries and the better accuracy of on-line ASV as they analyze not only the trajectory on the paper (pen-down), but also consider the trajectory in the air (pen-up) and its dynamic properties.

Obviously, impersonating pen-ups and dynamic properties is really challenging for a forger. On the one hand, a genuine signer signs quickly and swiftly which corresponds to a well-learnt movement. On the other hand, the forgeries sign carefully producing a larger and slower velocity profile than the genuine counterpart. This fact is easily detectable by on-line ASV such as those based on Dynamic Time Warping (DTW) [3].

To get a more realistic estimate of an ASV performance, it would be required to build a new database with more realistic on-line skilled forgeries. But this is an almost unrealistic goal to reach from a time, monetary and legal perspective.

In this paper we address this problem by providing, on the one hand, a methodology to generate better synthetic skilled forgeries at the testing phase of an ASV design and, on the other hand, a countermeasure technique to avoid this methodology to be used against such an ASV system in real life application.

For this purposes, to improve the skillfulness of any on-line
forged signatures, we propose the following algorithm, outlined in Figure 1, which is mainly based on replacing the real velocity profile by a synthetic one as follows: 1) The sampling points of an on-line signature are interpolated and the perceptual important points of the 8-connected trajectory are estimated. 2) A time is given to each stroke between perceptual important points. 3) A lognormal is assigned to each stroke between perceptual important points. Each lognormal is overlapped with the adjacent lognormals. 4) The 8-connected trajectory is resampled with the new velocity profile.

In this way, it is expected that the performance of skilled forgeries in a database fully resampled will be worse than the original one. At the same time, it is also expected that the performance in random forgeries will be similar to the original. In other words, the resampling will move the probability density function (pdf) of skilled forgeries scores toward the pdf of genuine signature scores. Besides, the pdf of genuine signature scores will not change.

The outline of the paper is as follows: Section 2 describes the procedure to estimate the perceptual important points of the trajectory. The velocity synthesis based on lognormal and resampling is detailed in the third section. The results are given in Section 4 while in Section 5 the conclusions are discussed which are drawn upon the results in our study.

II. PERCEPTUAL IMPORTANT POINTS ESTIMATION

Even though many proposals have been issued, the accurate estimation of the perceptual important points in handwriting is still a challenging problem. According to [8], some approaches worked out the curvature at point \( p \) as the tangent of the angle between the lines \( < p, p + d > \) and \( < p, p - d > \), \( d \) being a predefined distance. Different values of \( d \) were used in multiscale estimators. Other approaches estimated the curvature as the radius of the osculating circle [9] or spline reconstruction [8].

Focusing on handwriting segmentation, two relevant approaches have been found in the literature. The first ones [10] proposed a method on the basis that handwriting is composed of curvilinear and angular strokes. The second one suggested a multiresolution algorithm [11].

These methods calculated the curvature of the trajectory in order to estimate the perceptual important points by thresholding. The thresholding works reasonably well when it is applied to handwriting text. But some problems arise with Western signatures. This results mainly from the wide variety of curvatures because they combine text with flourishes. As a consequence, it produces corners of different sharpness in the same specimen.

To address these effects, a Two-Steps Perceptual Important Points Estimator (TS-PIPE) for handwriting signatures has been proposed in [12]. In the first step, the more salient corners are worked out by means of a multiscale estimation of the curvature [11]. In the second step, a novel approach is applied to work out the missed salient curvature points based on the fact that each single stroke can be approximated by a circumference arc [4]. Consequently, it could be thought that the trajectory between two salient curvature points is fairly circular and will change from one circle to the next one around the corners. In other words, if the trajectory between two detected corners in the first step is circular enough, no more corner will be added in the middle. Otherwise, new corner points will be added in the middle. For further details, please go through [12].

III. VELOCITY PROFILE SYNTHESIS AND RESAMPLING

This section is devoted to generate genuine-like synthetic lognormal velocity profile. For this purposes, we use the location of the perceptual important points to set up the minima in the velocity profile. The new velocity is obtained from the kinematic theory of rapid movements. It claims that the human being performs their movements with a velocity profile \( \bar{v}(t) \) which can be modeled as a linear combination of lognormals [4] as follows:

\[
\bar{v}(t) = \sum_{j=1}^{M} \bar{v}_j(t;D_j,\tau_j,\mu_j,\sigma_j^2)
\]  

being the velocity profile of each stroke \( v_j(t) \) defined as:

\[
v_j(t;\tau_j,\mu_j,\sigma_j^2) = \frac{D_j}{\sigma_j \sqrt{2\pi}(t-\tau_j)} \exp \left( -\frac{\ln(t-\tau_j) - \mu_j}{2\sigma_j^2} \right)
\]  

where \( t \) is the time basis, \( \tau_j \) the time of stroke occurrence, \( D_j \) the amplitude of each stroke, \( \tau_j \) the stroke time delay and \( \sigma_j \) the stroke response time, both on a logarithmic time scale. The distance \( s(t) \) traveled at time \( t \) is obtained as:

\[
s(t) = \int_{-\infty}^{\infty} v_j(t) dt = \frac{D_j}{2} \left( 1 + \text{erf} \left( \frac{\ln(t-\tau_j) - \mu_j}{\sqrt{2}\sigma_j} \right) \right)
\]
which is the lognormal cumulative function. Solving for \( t \) this equation, we get the time in terms of the distance as:

\[
    t(s) = \exp\left(\sqrt{2}\sigma_{j}\text{erf}^{-1}\left(\frac{2s(t)}{D_j} - 1\right) + \mu_j\right) + \tau_j
\]  

(4)

Consequently, to generate synthetic specimens, two tasks have to be performed, the first one to synthesize the velocity profile (working out the parameters of Eq. (1)) and the second one to sample the signature trajectory with Eq. (4).

A. Velocity profile synthesis

In this section, we adapt the procedure proposed in [13] to the present problem. Basically, to synthesize the velocity profile, the signature is segmented in strokes. Then a velocity profile is generated for each stroke which are accumulated to obtain the global synthetic velocity profile of the signature. Each stroke is located between two consecutive important points obtained by the TS-PIPE algorithm [12]. Let be \( Ns \) the number of strokes and \( ls_j, \forall j = 1, ..., Ns \) the length of the \( j^{th} \) stroke.

To each perceptual important point, it is assigned a time \( ts_j, \forall j = 0, ..., Ns \), where \( ts_0 = 0 \) is the beginning of the signature. Therefore, the \( j^{th} \) stroke spans from \( ts_{j-1} \) up to \( ts_j \). As has been said in section II, each perceptual important point corresponds to a minimum in the synthetic velocity profile.

The time of each perceptual important point is obtained as follows. The time between perceptual important points or velocity minima \( ts_j - ts_{j-1} \) is fixed to a fairly constant time following the hypothetical existence of the so-called Central Pattern Generators (CPG). The CPG produces rhythmic patterned outputs, without sensory feedback, to activate different motor pools [14]. This can be observed in the clearly periodic pattern of the handwriting velocity. Therefore, if the velocity generation is assimilated to the CPG step cycle, the duration of each stroke should be similar. Specifically, in the BiosecureID-132 database [16], the time between velocity minima has been calculated and modeled by a Normal distribution of average 0.1 and variance 0.005. Consequently, the time \( \Delta ts_j = ts_j - ts_{j-1}, \forall j = 1, ..., Ns \) is worked out following such distribution. This time corresponds with the assigned duration for each stroke.

Once defined the time scale of the signature, we generate a synthetic velocity profile for each stroke. Let \( v_j(t) \) be the velocity profile of the \( j^{th} \) stroke. Then it must be overlapped with the previous and next stroke. For this reason, the starting time of the stroke is set to \( \tau_j = ts_{j-1} - \Delta ts_j, \forall j = 1, ..., Ns \).

The values of \( D_j, \mu_j \) and \( \sigma_j^2 \) are set from the following two hypotheses: firstly, the margins for natural human handwriting given in [4] and secondly, it was heuristically observed that most of the lognormals were centered, i.e. the lognormal peak approaches the center of the stroke. Therefore, our skewness is close to zero, but positive and the kurtosis is around three.

The calculation of these values is suggested as follows. From Eq. (3) we deduce:

\[
    ts_j = \frac{D_j}{2}\left(1 + \text{erf}\left(\frac{\ln(ts_j - \tau_j) - \mu_j}{\sqrt{2}\sigma_j}\right)\right)
\]  

(5)

Note that \( ts_j - \tau_j = 2\Delta ts_j \). As \( \text{erf}(3) = 1 \), a possible solution for Eq. (5) is:

\[
    D_j = ls_j
\]  

(6)

\[
    \mu_j = \ln(2\Delta ts_j) - 3\sqrt{2}\sigma_j
\]  

(7)

Furthermore, if the lognormals were centered in the middle of the stroke with a low positive skew, their maximum or mode, defined by \( e^{\mu_j - \sigma_j^2} \), is around \( ts_{j-1} + \Delta ts_j/2 \) with a slight left skew. Therefore, it holds that:

\[
    ts_{j-1} + k_j \cdot \Delta ts_j - \tau_j = \Delta ts \cdot (1 + k_j) = e^{\mu_j - \sigma_j^2}
\]  

(8)

where the value \( k_j \), which provide a slight left skew, is worked out randomly for each stroke, following a uniform distribution in the margin [0.4 0.5]. This procedure is useful only for isolated strokes.

Finally, combining Eq. (7) and Eq. (8) we obtain:

\[
    \sigma_j^2 + 3\sqrt{2}\sigma - \ln\left(1 + k_j/2\right) = 0
\]  

(9)

Thus, this approach leads to assign to the parameters \( D_j \), the value of \( ls_j \) (see Eq. (6)), \( \sigma_j \) as the positive solution of a simple second order equation (see Eq. (9)), and \( \mu_j \) by substituting \( \sigma_j \) in Eq. (7).

Once the velocity profile of individual strokes are obtained, the velocity profile of the signature is worked out following Eq. (1). As a check of the obtained velocity profile of the signature, we must be sure that the integral of the signature velocity profile is equal to the total length of the signature.

B. Lognormal sampling of the trajectory

The time at every pixel in the signature trajectory is calculated...
as the integral of the velocity from zero up to each multiple of \( \frac{1}{f_m} \), \( f_m \) being the sample frequency. Figure 2 shows a synthetic example of a reconstructed velocity profile for a genuine and a forgery specimen, highlighting that the synthetic profile in the forged signature contains less lognormals than the original one.

IV. EXPERIMENTS

The experiments aimed to verify whether the resampled on-line profile is genuine-like enough. As such, the genuine signature should be similarly detected by an ASV whereas skilled forgeries should be more difficult to detect than the original ones. Therefore, the experimental methodology is designed as follows: 1. the performance of the original database is worked out for both random and skilled forgeries by using two on-line ASV, 2. the database is completely resampled with the proposed method, 3. the performance of the resampled database is obtained and compared to the original one.

The experiments have been run with the BiopsecureID-132 database [16] which contains 132 users, 16 genuine and 12 forgeries per user.

A. Comparing False Acceptance Rate and False Rejection Rate curves

The experiments in this subsection aimed at assessing whether the proposed method makes on-line skilled forgeries more skillful. For this purposes, we compare the False Acceptance Rate (FAR) and False Rejection Rate (FRR) curves of both original and resampled signatures. Such comparison is performed for both random and skilled forgeries (RF and SF, respectively) experiments. The random forgery experiment use as forgeries the genuine signatures of other signers. The so-called skilled forgery experiment uses as forgeries the 12 ones provided by the database. For the evaluation, two state-of-the-art on-line ASV have been taken into account: a Dynamic Time Warping (DTW) [3] and a Manhattan distance ASV [17].

Both ASV have been trained with the first 5 genuine signatures of the BiosecureID database [16]. For FRR, we have used the remainder 11 genuine signatures of the same signer. In the random forgery experiment we have used the remainder 11 signatures of all the other signers for calculating the FAR. In the skilled forgery experiment, we have used the 12 available skilled forgeries of each signer to calculate the corresponding FAR. These experiments were performed for the original and resampled database under the same conditions.

The results in terms of EER can be seen in Table I and II for DTW and Manhattan distance ASV respectively. As expected, the EER in the random forgery (RF) experiment, which only involves genuine signatures, are similar between original and resampled signatures. However, the EER increases significantly in the skilled forgery (SF) experiment.

This experiment was repeated removing the pen-ups from the trajectory, that is to say discarding the samples that corresponds
Two ways of detecting the resampling of a signature are devised in this paper based on the speed profile:

**Lognormality-wise:** Following the Sigma-Lognormal model, a handwritten signature can be decomposed as a sum of weighted and overlapped lognormals. However, there are many factors that modify the free performance of the motor system, which leads to deviations in the lognormal speed profile. For instance, some joint pain, uncomfortable wear or posture, emotional state of the signer, and so on. As consequence, the Sigma-Lognormal model is only able to approach the velocity profile of an original handwritten signature up to a reasonable Signal-to-Noise ratio (SNR), which is defined as:

$$SNR = 10 \log \left( \frac{\int_{t=0}^{T} v_o(t)^2 \, dt}{\int_{t=0}^{T} (v_o(t) - v_r(t))^2 \, dt} \right)$$

Where $v_o(t)$ is the original velocity profile and $v_r(t)$ is the resampled velocity profile.

On the contrary, the velocity profile of the resampled signature is purely a sum of lognormals. Therefore, it is expected that its Sigma-Lognormal decomposition reached higher Signal-to Noise ratios. As example of the performance of the SNR score as countermeasure, Figure 5 shows the SNR distribution of original and resampled signatures for the genuine and forgery cases along the on-line real dataset in BiosecureID database.

**Regularity-wise:** Following the above rationale, the velocity profile of resampled signatures is expected to be more regular than original ones. This regularity can be measured as the variance of the time between minima of the velocity profile. It is expected that the regularity is more stable for resampled than for original profiles. As example, Figure 6 shows the distribution of the regularity for original and resampled signatures for the genuine and forgery cases through the on-line real dataset in BiosecureID database.

In both cases, to detect a counterfeited signature, the countermeasure algorithm compares the above mentioned SNR or regularity score with a threshold. In this case, we use the Bayes threshold. If the SNR score of a given signature is greater than its threshold or the Regularity score is lower than its threshold, then the signature is supposed to be a resampled signature and discarded as original.

To evaluate the performance of both methods, the next measures have been worked out for genuine, forgeries and all together: precision, recall or sensitivity, specificity and accuracy which are defined as:

$$precision = \frac{tp}{tp + fp} \quad recall = \frac{tp}{tp + fn}$$

$$specificity = \frac{tn}{tn + fp} \quad Accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

Being $tp$: true positive, $tn$: true negative, $fp$: false positive and $fn$: false negative. The positive hypothesis is that the signature has been resampled.
The precision is the ratio of signatures classified as resampled that are truly resampled. Recall refers to the ratio of resampled signatures detected. Specificity appertains to the ratio of signatures classified as original that are really original. Finally, accuracy is the ratio of signatures rightly classified.

The results are displayed in Table III for SNR and Table IV for Regularity. The measure based on regularity is more effective detecting resampled signatures than SNR one as was expected as original signatures are also lognormals. Therefore, we recommend the countermeasure based on regularity score.

VI. CONCLUSION

This paper proposes a procedure based on the Sigma-Lognormal model to make on-line forgeries more skilful. While a well-trained forger can imitate accurately the genuine signature trajectory, they usually fail to emulate feasibly the velocity profile. Therefore, this paper proposes to improve the skill of a forged signature by modifying the speed profile of the signature by resampling. It is expected that the new synthetic velocity profile will be more genuine-like.

On the other hand, it could be said that the minima in the velocity profile of a genuine signature coincide with the perceptual important points in the trajectory. So that, the skillfulness of the skilled forgeries is improved by resampling the on-line signature producing minima in the estimated perceptual important points. The perceptual important points are estimated with a two steps algorithm (TS-PIPE) [12]. Then, the synthetic velocity profile is build up based on the kinematic theory of rapid movements: A set of lognormals are fitted to the trajectory regarding the location of the estimated perceptual important points. This strategy is applied to original signatures, genuine or forgeries, to produce a synthetic (resampled) version of the signature with a more genuine-like velocity profile.

The conducted experiments show that the resampled skilled forgeries contain a similar number of speed minima than their respective genuine. Additionally, the EER is significantly increased for skilled forgeries while it is barely modified in random forgeries. These experiments have been conducted in two different on-line ASV. The robustness of these observations have been confirmed in on-line signatures with and without pen-up trajectories.

Some further work is still required to reduce the differences between the False Rejection Curves of the original and resampled databases. Additionally, the False Acceptance Curves should be more similar in the random forgery experiments and move more toward left in the skilled forgery experiments. The paper ends by proposing a countermeasure to detect this kind of fake signatures in terms of measuring the regularity of the speed profile of the given signature.

ACKNOWLEDGMENT

This study was funded by the Spanish government’s MIMECO TEC2016-77791-C4-1-R research project and European Union FEDER program/funds and by grant from NSERC Canada grant RGPIN-2015-06409 to Réjean Plamondon.

REFERENCES


### Table III: Precision, Recall, Specificity and Accuracy for SNR

<table>
<thead>
<tr>
<th></th>
<th>Genuine</th>
<th>Forgeries</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>0.79</td>
<td>0.87</td>
<td>0.83</td>
</tr>
<tr>
<td>Recall</td>
<td>0.80</td>
<td>0.87</td>
<td>0.83</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.79</td>
<td>0.87</td>
<td>0.83</td>
</tr>
</tbody>
</table>

### Table IV: Precision, Recall, Specificity and Accuracy for Regularity

<table>
<thead>
<tr>
<th></th>
<th>Genuine</th>
<th>Forgeries</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>0.89</td>
<td>0.98</td>
<td>0.92</td>
</tr>
<tr>
<td>Recall</td>
<td>0.89</td>
<td>0.97</td>
<td>0.91</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.89</td>
<td>0.97</td>
<td>0.91</td>
</tr>
</tbody>
</table>
Modeling the Complexity of Biomechanical Tasks using the Lognormality Principle: Applications to Signature Recognition and Touch-Screen Children Detection

Ruben Vera-Rodriguez  
*Biometrics and Data Pattern Analytics (BiDA) Lab - ATVS*  
*Universidad Autonoma de Madrid*  
Madrid, Spain  
ruben.vera@uam.es

Ruben Tolosana  
*Biometrics and Data Pattern Analytics (BiDA) Lab - ATVS*  
*Universidad Autonoma de Madrid*  
Madrid, Spain  
ruben.tolosana@uam.es

Javier Hernandez-Ortega  
*Biometrics and Data Pattern Analytics (BiDA) Lab - ATVS*  
*Universidad Autonoma de Madrid*  
Madrid, Spain  
javier.hernandezo@uam.es

Aythami Morales  
*Biometrics and Data Pattern Analytics (BiDA) Lab - ATVS*  
*Universidad Autonoma de Madrid*  
Madrid, Spain  
aythami.morales@uam.es

Julian Fierrez  
*Biometrics and Data Pattern Analytics (BiDA) Lab - ATVS*  
*Universidad Autonoma de Madrid*  
Madrid, Spain  
julian.fierrez@uam.es

Javier Ortega-Garcia  
*Biometrics and Data Pattern Analytics (BiDA) Lab - ATVS*  
*Universidad Autonoma de Madrid*  
Madrid, Spain  
javier.ortega@uam.es

Abstract—This paper focuses on modeling the complexity of biomechanical tasks through the usage of the Sigma LogNormal model of the Kinematic Theory of rapid human movements. The Sigma LogNormal model has been used for several applications, in particular related to modeling and generating synthetic handwritten signatures in order to improve the performance of automatic verification systems. In this paper we report experimental work for the usage of the Sigma LogNormal model to predict the complexity of biomechanical tasks on two case studies: 1) on-line signature recognition in order to generate user-based complexity groups and develop specific verification systems for each of them, and 2) detection of age groups (children from adults) using touch screen patterns. The results achieved show the benefits of using the Sigma LogNormal model for modeling the complexity of biomechanical tasks in the two case studies considered.

Index Terms—On-line signature verification, user profiling, neuromotor model, signature complexity, age prediction, touch dynamics, biometrics

I. INTRODUCTION

On-line signature verification and other handwritten tasks (drawings, touch patterns, etc.) are experiencing a high development recently due to the technological evolution of digitizing devices, including smartphones and tablets. Such handwritten data can be applied to many applications in different sectors such as security, e-government, healthcare, education, user profiling, advertising or banking [1], [2].

This work has been supported by project TEC2015-70627-R MINECO/FEDER and by UAM-CecaBank Project. Ruben Tolosana is supported by a FPU Fellowship from Spanish MECD, and Javier Hernandez by a FPI Fellowship from UAM.

This paper focuses on modeling the complexity of handwritten information, which can be a very important factor in different applications related to handwriting. We propose to model the complexity of handwritten tasks through the usage of the Sigma LogNormal model of the Kinematic Theory of rapid human movements [3]. The Sigma LogNormal model has been used in the past for several applications. One of the most successful ones has been the synthetic generation of handwriting, in particular signatures (two examples in [4] and [5]). This model has recently been used in [6] and [7] not to generate synthetic signature samples, but to improve the performance of traditional signature verification systems. In [6] the authors proposed a skilled forgery detector using some features extracted from the Sigma LogNormal model whereas in [7], a new set of features based on the Sigma LogNormal model was proposed achieving very good performance.

In this paper we report experimental work for the usage of the Sigma LogNormal model to predict the complexity of biomechanical tasks on two case studies: 1) The first one describes its application to on-line signatures in order to generate user-based complexity groups (as there are users with very complex signatures and other with very simple ones). Then, a specific signature verification system is developed for each complexity group achieving very significant improvements of verification performance [8]. 2) On the other hand, the second one describes its application to detect age groups (children from adults) in touch dynamic tasks performed on smartphones or tablets [9], as the difference between adults and children is mainly caused by the different maturity of their anatomy and...
neuromotor system. These are less mature in children, so they have worse manual dexterity causing rougher movements [10] [3].

The remainder of the paper is organized as follows. Sec. II describes the Sigma LogNormal model, used in this work to model the complexity of handwritten tasks. Sect. III describes the first case study focused on modeling the complexity of on-line signatures and its experimental results. Sect. IV describes the second case study focused on modeling the complexity of touch dynamic information in order to detect age groups and its experimental results. Finally, Sec. V draws the final conclusions and points out some lines for future work.

II. THE SIGMA LOGNORMAL MODEL

Many models have been proposed to analyze human movement patterns in general and handwriting in particular. These models allow the analysis of features related to motor control processes and the neuromuscular response, providing complementary features to the traditional X and Y coordinates related to handwriting tasks. One of the most well known writing generation models is the Sigma LogNormal model [3] [11].

The Sigma LogNormal model states that the velocity profile of human hand movements can be decomposed into strokes. Moreover, the velocity of each of these strokes, $i$, can be described with a speed signal $v_i(t)$ that has a lognormal shape:

$$|v_i(t)| = \frac{D_i}{\sqrt{2\pi}\sigma_i(t-t_0)} \exp\left( -\frac{(\ln(t-t_0) - \mu_i)^2}{2\sigma_i^2(t-t_0)} \right) \tag{1}$$

where each of the parameters are described in Table I. The complete velocity profile is modelled as a sum of the different individual stroke velocity profiles as:

$$v_{ic}(t) = \sum_{i=1}^{N} v_i(t) \tag{2}$$

where $N$ is the number of lognormals of the entire movement. A complex action, like a handwritten signature or touch task, is a summation of these lognormals, each one characterized by different values for the six parameters in Table I. Fig. 1 shows an example of the lognormal velocity profiles extracted for each stroke of one signature.

III. CASE STUDY 1: ON-LINE SIGNATURE COMPLEXITY

Signature verification systems have been shown to be highly sensitive to signature complexity [12]. In [13], Alonso-Fernandez et al. evaluated the effect of the complexity and legibility of the signatures for off-line signature verification (i.e. signatures with no available dynamic information) pointing out the differences in performance for several matchers. Signature complexity has also been associated to the concept of entropy, defining entropy as the inherent information content of biometric samples [14], [15]. In [16] a “personal entropy” measure based on Hidden Markov Models (HMM) was proposed in order to analyse the complexity and variability of on-line signatures regarding three different levels of entropy. In addition, the same authors have recently proposed in [17] a new metric known as “relative entropy” for classifying users into animal groups where skilled forgeries are also considered. Despite all the studies performed in the on-line signature trait, none of them have exploited, as far as we are aware, the concept of complexity in order to develop more robust and accurate on-line signature verification systems.

A. Proposed System

Based on the parameters of the Sigma Lognormal model, we propose to use the number of lognormals ($N$) that models each signature as a measure of the complexity level of the signature. Once this parameter is extracted for all available genuine signatures of the enrolment phase, the user is classified into a complexity level using the majority voting algorithm (low, medium and high complexity levels). Only genuine signatures are considered in our proposed approach for measuring the complexity level. The advantage of this approach is that the signature complexity detector can be performed off-line thereby avoiding time consuming delays and making it feasible to apply in real time scenarios.

Then, after having classified a given user into a complexity group, a specific on-line signature verification module based on time functions (a.k.a. local system) [18] has been adapted to each signature complexity level. For each signature acquired, signals related to $X$ and $Y$ pen coordinates are used to extract a set of 23 time functions, similar to [19]. The most discriminative and robust time functions of each complexity level are selected using the Sequential Forward Feature Selection algorithm (SFFS) enhancing the signature verification system in terms of EER. A DTW algorithm [20] is used to compute
the similarity between the time functions from the input and training signatures.

B. Database and Experimental Protocol

In this case, BiosecurID database [21] is considered. Signatures were acquired from a total of 400 users using a Wacom Intuos 3 pen tablet with a resolution of 5080 dpi and 1024 pressure levels. The database comprises 16 genuine signatures and 12 skilled forgeries per user, captured in 4 separate acquisition sessions. Each session was captured leaving a two month interval between them, in a controlled and supervised office-like scenario. Signatures were acquired using a pen stylus. The available information within each signature is: X and Y pen coordinates and pressure. In addition, pen-up trajectories are available.

The experimental protocol has been designed to allow the study of different signature complexity levels in the system performance. Two main experiments are carried out: 1) evaluation of the signature complexity detector proposed in this work in order to classify users into different complexity levels, and 2) evaluation of the proposed approach based on a separate on-line signature verification system adapted to each signature complexity level.

For the first experiment, our proposed signature complexity detector is analyzed using all available users from BiosecurID. For the second experiment, the BiosecurID database is split into development dataset (40% of the users) and evaluation dataset (the remaining 60% of the users). The development dataset is considered in order to select the most discriminative and robust time functions for each signature complexity level using the SFFS algorithm whereas the evaluation dataset is considered for the evaluation of the proposed system. Both skilled and random forgeries are considered using the 4 signatures from the enrolment session as reference signatures and the remaining 12 genuine signature and 12 skilled forgeries signature as the test. The final score is obtained after performing the average score of the four one-to-one comparisons.

C. Results

The first experiment was designed to evaluate the proposed approach for signature complexity detection. For this, the signature complexity detector was performed in two different steps. First, each user of the BiosecurID database was manually labelled in a signature complexity level (low, medium, high). This process was carried out seeing the image of just one genuine signature per user and it was performed by two annotators and two times each in order to keep consistency on the results. Three different complexity levels were considered based on previous works [17]. Users with signatures longer in writing time and with an appearance more similar to handwriting were labelled as high-complexity users whereas those users with signatures shorter in time and with generally simple flourish with no legible information were labelled as low-complexity users. This first stage served as a ground truth. Following this stage, the Sigma LogNormal parameter $N$ was extracted for each available genuine signature of the BiosecurID database (i.e. a total of $400 \times 16 = 6400$ genuine signatures). Then, we represented for each complexity level using all genuine signatures of the BiosecurID database. The three proposed complexity-dependent decision thresholds are highlighted by black dashed lines and were selected in order to minimize the number of misclassifications between different signature complexity levels. Signatures with lognormal values equal or less than 17 are classified as low-complexity signatures whereas those signatures with more than 27 lognormals are classified into the high-complexity group. Otherwise, signatures are categorized into medium-complexity level. Fig. 3 shows some of the signatures classified into each complexity level.

We now analyse each resulting complexity level following the same procedure proposed in [17]: analysing the system performance for different complexity groups considering only $X$ and $Y$ pen coordinates. It is important to remark that each user is classified into a complexity level applying the majority voting algorithm to all available enrolment signatures of the user. Table II shows the system performance for each complexity level in terms of EER(%). The results show different system performance regarding the signature complexity level. Users with a high complexity level have an absolute improvement.
of 4.3% compared to users categorized into a low complexity level for skilled forgeries.

Then, the second part of the experimental work was focused on developing a specific verification system for each group of signature complexity. For this, the SFFS algorithm was applied to the development dataset in order to find the most discriminative time functions for each complexity group. Then, the evaluation of the proposed system was compared to a baseline system based on DTW and the same system (same time functions) for all complexity groups, similar to the baseline system presented in [6].

Table III shows the evaluation results achieved considering our proposed approach based on personal entropy on-line signature verification systems. Analysing the results obtained, our Proposed Systems achieve an average absolute improvement of 2.5% EER compared to the Baseline System for the case of skilled forgeries. It is important to note that for the most challenging users (users with high personal entropy level), our proposed approach achieves an absolute improvement of 3.7% EER compared to the Baseline System. Analysing the results obtained for the random forgery cases, our Proposed Systems also achieves improvements for all personal entropy levels. For this case, the improvement has been lower than for skilled forgery cases due to its low values and the way that the SFFS algorithm was applied during the training of the systems (focused on skilled forgery cases). Results obtained after applying our proposed approach based on personal entropy on-line signature verification systems outperform the results of the state-of-the-art for the BiosecurID database. In [6], the authors achieved an absolute improvement of 1.0% EER for skilled forgery cases whereas our proposed approach achieves an average absolute improvement of 2.5% EER compared to the same Baseline System.

IV. CASE STUDY 2: PREDICTING AGE GROUPS FROM TOUCH PATTERNS

Age groups prediction based on handwritten touch patterns acquired from touchscreen devices such as smartphones or tables is a recent and important challenge. Touchscreen panels have changed the way users interact with new devices. The touchscreen enables an intuitive experience of use that allows a direct interaction with what is being displayed. In the last years there has been a huge spread of the use of this kind of devices by young children. The study in [22] reveals that 97% of US children under the age of four use mobile devices, regardless of family income. The age is a key attribute in user profiling with direct application on several automatic systems (e.g. parental control, recommender systems, advertising, etc.).

In this case study we propose the use of the Sigma LogNormal model to detect age groups as simple application of the model to drag and drop touch tasks showed large differences between adults and children velocity profiles. In Figure 4, an example of these types of profiles is presented, consisting in performing a drag and drop task in both cases. A visual comparison between children and adults velocity profiles shows that children’s signals are usually composed by a higher number of strokes than the adults’ ones, and therefore have a higher degree of complexity.

Moreover, there are previous studies like [23], which have proved that the Sigma LogNormal model can be used to characterize children handwriting. They conclude that there are two main groups of children separable by looking at their learning stage. Children’s neuromotor skills become more similar to the adults’ skills when they grow up, namely, when they finish their preoperational stage. At age 10 children know how to activate each little muscle properly to produce determinate fine movements [24]. As they are based on the same neuromotor skills, the principles applied to handwriting models can be also used to model touchscreen patterns.

A. Proposed System

In this case, a more complex system was developed compared to Case Study 1 in order to predict age groups from drag and drop touch tasks, as the main focus here was to optimize the final classification result.

The parameters of the Sigma LogNormal model (as described in Sect. II) were used to calculate 18 different features per lognormal as described in [25]. These features can be classified into two groups: space-based and time-based. Space-based features are those that give information about the spatial distribution of the strokes, such as $D_i$, $\mu_i$, $\sigma_i$, and other features based in $\theta_{si}$ and $\theta_{ei}$ (see Table I). Time based features are composed by the values of speed at some relevant points of the strokes like their maximum or inflexion points; and the time-offsets between those points. The task time and the number of lognormals in each task have been added as additional features.

It is worth noting that the lognormals with amplitude value lower than a threshold were discarded. Then, the 18 features from [25] are computed for each stroke, and each parameter is averaged across strokes. The 18 averaged parameters are augmented with the task time and the number of strokes to generate the final feature vector of size 20.

As a classifier we use a SVM (Support Vector Machine) with a RBF (Radial Basis Function) kernel because of its good general performance in binary classification tasks and the few number of parameters to configure.

B. Database and Experimental Protocol

The database used is publicly available and was presented in [24]. It is comprised with data from touchscreen activity of both children and adults performing predesigned tasks in an ad-hoc app. In the present work, we have used the data from singletouch and multitouch drag and drop activities. Drag and
TABLE III

EXPERIMENT 2: SYSTEM PERFORMANCE RESULTS (EER IN %) ON THE EVALUATION DATASET FOR EACH SIGNATURE COMPLEXITY LEVEL.

<table>
<thead>
<tr>
<th></th>
<th>Low C. Baseline</th>
<th>Low C. Proposed</th>
<th>Medium C. Baseline</th>
<th>Medium C. Proposed</th>
<th>High C. Baseline</th>
<th>High C. Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skilled forgeries</td>
<td>13.8</td>
<td>10.1</td>
<td>7.5</td>
<td>5.2</td>
<td>6.2</td>
<td>4.6</td>
</tr>
<tr>
<td>Random forgeries</td>
<td>1.5</td>
<td>1.3</td>
<td>0.7</td>
<td>0.5</td>
<td>0.9</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Fig. 4. Comparison between Sigma LogNormal speed profiles for (a) an adult and (b) a child following the same task.

drop activities consist of picking one object on the device screen and moving it to a target area. Multidevice information is available as the users have completed the tasks both in a smartphone and in a tablet. Both single-sensor and cross-sensor tasks are analyzed.

The dataset is composed by 89 children between 3 and 6 years old and 30 young adults under 25 years old. The mean age of the children is 4.6 years. The total number of drag and drop tasks is 2912 for children and 1157 for adults (see [24] for more details).

As the experimental protocol, the database was divided randomly into training (60%) and testing (40%). The random selection was repeated 50 times and the final performance is presented in terms of averaged correct classification accuracy.

C. Results

Table IV shows the accuracies obtained according to the different scenarios. They are presented in terms of correct classification accuracy (percentage of samples from both classes correctly classified).

The mean value of accuracy having into account all the evaluated scenarios is 92.8%. The classification rates are over 96% in a single-sensor setting and over 95% in a cross-sensor scenario. The best results are obtained with tablets as sensors, while using smartphone’s data slightly degrades the results.

Compared with [26] where they get an accuracy rate of 86.5% using one tap task for classification and with a single-sensor approximation (using smartphone’s data), our system performs better, getting a 93.6% of accuracy using only data from smartphones, and over 96% using data from tablets. Another conclusion that can be extracted of Table IV is that the data obtained from multitouch tasks get worse results than the singletouch cases. The best multitouch scenario is obtained using tablet’s data for both training and testing, with a 94.6% of accuracy, compared with its singletouch counterpart that gets a 96.3%. This may be caused by the less developed control of the left hand by right-handed people and vice versa. The main reason for using the Sigma LogNormal model is that adults have a better control of fine movements than children, what is translated to different values for the model parameters [24].

The cross-sensor scenarios get results not too far from the single-sensor scenarios. The results obtained using smartphone singletouch data for training, and tablet singletouch data for testing (95.9% of accuracy) are quite similar to those obtained using only tablet singletouch data (96.3% of accuracy). This fact makes this type of systems very suitable for real applications due to its high independence of the device used.

Due to the higher number of children in the database compared to adults, selecting a percentage of the total users make the two scenarios unbalanced. Experiments balancing the number of both classes in training and testing have been made. The results show small variations around 1% of accuracy with respect to the presented results.

V. CONCLUSIONS

This work has reported experimental results on modeling the complexity of biomechanical tasks through the usage of the Sigma LogNormal model of the Kinematic Theory of rapid human movements. Two different case studies have been analyzed.

The first case study has focused on applying the Sigma LogNormal model to develop an on-line signature complexity detector. Just by using the number of strokes of the signatures was enough to obtain very good results differentiating between three different signature complexity groups (low, medium and high). As a second stage, a specific signature verification system was developed for each signature complexity group by carrying out a time functions selection process. Very significant improvements of recognition performance have been shown when comparing the proposed system with a baseline, being both based on DTW and time functions as features. For future work, the approach considered in this work will be further analysed using the e-BioSign public database [27] in order to consider new scenarios such as the case of using the finger as the writing tool. Also, novel systems based on the usage of Recurrent Neural Networks (RNNs) [28] and the fusion of different systems [29] will be considered.
On the other hand, the second case study has focused on age group prediction (children from adults) from handwritten touch patterns acquired from touchscreen devices such as smartphones or tablets. Applying the Sigma LogNormal model to some examples of drag and drop tasks from children and adults showed that children had a more complex velocity profiles with a larger number of sigma LogNormals. The proposed approach is based on 20 features extracted from the model, and results achieved were very promising with classification rates over 96% in a single-sensor setting and over 95% in a cross-sensor scenario.

REFERENCES


Can the Sigma-lognormal Modelling Help to Monitor Child Graphomotor Skill Progress?

Ist Rémi Céline  
Department of Mathematics, Computer sciences  
LAMIA Laboratory  
Pointe-à-Pitre, Guadeloupe  
celine.remi@univ-antilles.fr

2nd Vaillant Jean  
Department of Mathematics, Computer sciences  
LAMIA Laboratory  
Pointe-à-Pitre, Guadeloupe  
jean.vaillant@univ-antilles.fr

3rd Nagau Jimmy  
Department of Mathematics, Computer sciences  
LAMIA Laboratory  
Pointe-à-Pitre, Guadeloupe  
jimmy.nagau@univ-antilles.fr

4th Plamondon Réjean  
Département de Génie Électrique  
SCRIBENS Laboratory  
Montréal, Canada  
rejean.plamondon@polymtl.ca

Abstract—This work deals with the study of the relevance of the use of the Kinematic Theory as a conceptual basis to identify some reliable variables and to define an approach for the monitoring of children’s graphomotor progress all over their scholarship from kindergarten up to the stage when they manage cursive handwriting efficiently. It is focused on cursive handwriting of usual isolated words. Its first results underline the capacity of the sigma-lognormal modelling to reproduce movements produced by children at primary school and the interest of the studied variables to contribute to the discrimination between the pupils’ scholar grades.

Keywords—Handwriting learning, primary school, children, Kinematic theory, Sigma-lognormal analysis, fine motor control modelling, graphomotor movement control, monitoring

I. INTRODUCTION

Before handwriting becomes an additional and complementary way of a scholar expression for children, they have to be familiarized with an efficient use of a writing tool. This apprenticeship usually begins at the kindergarten mainly by scribbling and producing graphics and it continues at primary school mainly through copying tasks. The interest of automating children’s handwriting is to make them able to focus their attention on the linguistic dimension of their production. Studies have shown that a certain number of children labelled as presenting “disorders” saw their access to the written language and literacy strongly restrained by graphomotor difficulties as analyzed by [1] and [2]. Most of these graphomotor difficulties are temporary. Thanks to an effective monitoring of child graphomotor skill progress they could be avoided or at least should be detected earlier for their remediation. However, there is no objective and simple tool which could help the education community for making a continuous and objective monitoring of each pupil graphomotor skill progress from the beginning stage of graphomotor discovery at kindergarten up to the automation and mastery of handwriting. We have chosen to deal with this problem. In this perspective, we are beginning a longitudinal 3-years study with the Guadeloupean Direction of the academic problem. In this context, we are starting this study because, a priori, there is no interference with the basic knowledge such as the spelling, etc., during writing. Moreover, the writing of his name is the first type of cursive handwriting task that is usually and often taught to children in French school, starting at since kindergarten. In this paper; we present the results of a preliminary analysis of the interest of the sigma-lognormal processing of children’s productions, as made on a database acquired in a previous transversal study [3]. The following questions are explored among a set of sigma-lognormal and classical variables that we had already studied on sets of handwritten productions like movement patterns of pseudo-letters [4] and scribbles [5], [6]: 1-Can we satisfyingly reproduce children’s cursive handwriting movements with the sigma-lognormal model? 2-Can we identify variables that significantly reflect the learning grades in case of cursive words productions? In the sequel, a synthetic review of the principal hypotheses of the lognormal paradigm is first proposed. Next, we detail some aspects of the exploratory study like the experimental data used before the presentation and discussion of the main results. Lastly, we describe the current step of this ongoing work and its challenges.

II. MAIN HYPOTHESIS OF THE LOGNORMAL PARADIGM

The sigma-lognormal model was conceived as a way to approximate fast complex handwriting movements [7]. According to the lognormal paradigm, a complex handwriting movement takes its origin in the execution by the neuromuscular system of an action plan made up of a sequence of NbLog virtual targets. These targets are linked pairwise with circle arcs forming virtual trajectories. The speed profile for each virtual trajectory follows a scaled time-shifted lognormal function:

\[ v_j(t) = \frac{D_j}{\sigma_j(t-t_0)/\sqrt{2\pi}} \exp \left( -\frac{1}{2\sigma_j^2} [\ln(t-t_0)-\mu_j]^2 \right) \]  

(1)
The tangential velocity profile, \( v(t) \), for the complex movement that the set of neuromuscular commands contribute to generating is satisfyingly approximated, that is to say with a controllable value of the signal to noise rate (SNR), by the vector summation of all the set of lognormals \( v(t) \) defined by the equation:

\[
\varphi(t) = \sum_{i=1}^{N_{\text{log}}} \frac{\sum_{j=1}^{N_{\text{log}}} \frac{N_{\text{log}}}{\sigma_{j}(t-t_{j})^{2}} \exp \left( -\frac{1}{2\sigma_{j}^{2}} \left[ \log(t-t_{j}) - \mu_{j} \right]^{2} \right) \cdot \text{SNR} \geq 2 \quad (2)
\]

Each of the lognormal \( v_{i}(t) \) is defined by a set of six sigma-lognormal parameters \( \{ t_{i}, D_{i}, \theta, \theta_{i}, \sigma_{i}, \mu_{i} \} \) which are assumed to ultimately reveal the motor control process involved in the neuromuscular response [8].

### III. Participants and Protocols of Collection of the Handwritten Samples

For this preliminary study of the interest of the sigma-lognormal processing of children’s productions, we have chosen to take into account some previous data available from 111 pupils of 5 grades of French primary school who had participated to a transversal experimentation [3]. 18 of these pupils were from CP grades, 17 from CE1, 24 from CE2, 24 from CM1 too and 28 were from CM2 grade. Each participant had handwritten his first name under two conditions : at first with visual control (P1) and next without visual control of his production (P2). All the pupils had also handwritten the pseudo-word "tintin" 2 times: at first with visual control (T1) and next without visual control of his production (T2). Children’s graphomotor productions were recorded online with Calcomp DrawingBoardIII digitizers at 100 Hz with a spatial resolution of 0.001 inches. Table 1 indicates the number of children from each grade that have participated in the data collection. 111 valid samples were acquired for the first name tasks P1 and P2 and 110 ones for the T1 and the T2 tasks.

### IV. Method Used for the Processing of the Handwritten Samples and Types of Variables Extracted

All handwritten samples collected have been processed thanks to the Script Studio software [9] and the Dekattras software [4]. Twelve variables were extracted on each of the samples collected for each of the 4 tasks: P1, P2, T1 and T2. Their types are summarized in Table 1. These variables are related to 4 qualitative criteria with which graphomotor performances are usually analyzed and described [1], [4], [9].

### V. Can We Satisfyingly Reproduce Children’s Cursive Handwriting Movements with Sigma-Lognormal Modelling?

To answer this first question, we performed a descriptive analysis on the trajectory reconstruction quality. Like in [4], we categorized the SNR into four levels (SNR ≥ 25 = excellent quality, 20 ≤ SNR < 25 = good quality, 15 < SNR < 20= correct quality, SNR ≤ 15 = unsatisfactory quality). As illustrated by Fig. 1, according to the SNR, 2.3%, 40.5% and 50.7% of the recorded samples of cursive handwriting movements were reconstructed respectively with excellent, good, and correct quality considering movements for the four tasks: P1, P2, T1 and T2 produced by all participating children. 6.6% of the reconstructions were evaluated as unsatisfactory. This global distribution confirms that the sigma-lognormal model can accurately reproduce most of the children’s trajectories when they write familiar and simple cursive isolated words. In this context, it makes sense to use the sigmalognormal model to analyze the children productions of cursive isolated words. It must also be noted that 40.5% of the required trajectories is reproduced with a SNR higher than 20 dB. This means that these trajectories have been made by children who are almost in good neuromotor control. These rates provided by this preliminary study, in conjunction with those already obtained on kindergarten children’s production [5] reveal that the sigma-lognormal modelling can be used to satisfyingly reproduce children’s more or less complex cursive handwriting movements from scribbles up to cursive isolated words. Furthermore, these results signify that in scholar usual context of handwriting, the SNR estimated by the Xzero robust algorithm implied by the Scriptstudio Software could be used as a first filter for the graphomotor evaluation of the control of productions involved by pupils. If we consider each task Ti independently, as shown in Fig. 2, it appears that in the case of the cursive handwriting of the pseudo-first name “tintin”, whatever the condition of execution : with (T1) or without (T2) vision, the percentage of successfully reconstructed trajectories increased from younger to older pupils. Indeed, in the case of T1 as in the case of T2, from 33.3% for the CP, it rose to 60.7% in CM2. In the case of the tasks P1 and P2, although the good quality rates of reproduction are also lower in the first year of primary school, this trend to the progressive strengthening of the rates of good quality reproduction with the rise of the rank is not observed in a similar way than in the case of the tasks T1 and T2. This can be explained by the variability of the complexity of the first name defined by the number of letters and the level of their complexity which can introduce a bias. Another interesting observation is related to the rates of unsatisfactory quality of reconstruction that increases with the condition without vision for grades CE2 and CM1 while they decrease significantly for grades CP and CE1. This trend can

### Table 1. Table Type Styles

<table>
<thead>
<tr>
<th>Type of variable</th>
<th>Graphomotor Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rapidity</td>
</tr>
<tr>
<td>Spatial</td>
<td></td>
</tr>
<tr>
<td>Temporal</td>
<td></td>
</tr>
<tr>
<td>Kinematic</td>
<td></td>
</tr>
<tr>
<td>Lognormal</td>
<td>SNR</td>
</tr>
<tr>
<td>Postural</td>
<td>Pressure</td>
</tr>
<tr>
<td>Procedural</td>
<td>Pen Lift Duration / Writing Duration</td>
</tr>
</tbody>
</table>
be an expression of the changes of control strategy which is currently assumed to happen around 8-9 years old [1], [9].

Fig. 1. Percentages of the reconstructed trajectories within the four SNR intervals for each grade for all tasks. The global percentages for all grades within the four SNR intervals are also provided for task T1 and task T2.

Fig. 2. Percentages of the reconstructed trajectories within the SNR intervals for tasks T1 and T2 for each grade.

VI. CAN WE IDENTIFY VARIABLES THAT SIGNIFICANTLY REFLECT THE LEARNING GRADES IN CASE OF CURSIVE WORDS PRODUCTIONS?

For each pupil, each of the 12 types of variable was computed for each of the 4 tasks. Moreover, ratios were estimated between the two conditions with and without vision. Finally, the behaviour of 72 variables coded as Xi was analyzed thanks to various native functions from the R programming environment (Release 3.4.0). Before we focused on identifying discriminatory variables reflecting the learning grades from the set of variables extracted from the tests : P1; P2; T1 and T2, we performed Kruskal-Wallis tests to investigate if there were differences among the 5 grades with respect to each measured variable. The results are presented in Table 2. The results presented in Table 2 show that each of the 12 types of variable which were extracted can significantly contribute to discriminating between at least 2 of the 5 grades that are represented. It is interesting to note that all types of features chosen are a priori good candidate to contribute to discriminating graphomotor behaviours in the case of at least one chosen task. Moreover, 8 of the 12 types of variable can significantly contribute to discriminating between at least 2 of the 5 grades whatever the task of handwriting of this preliminary study. By means of the R package FactoMineR, we performed a principal component Analysis (PCA) with respect to the grades on the set of variables composed of the 48 ones extracted on the tasks enhanced by their ratios. We have introduced these ratios on the basis that they were supposed to be informative on the consistency of the graphomotor performance criteria with respect to perturbations like the lack of visual feedback. Sixteen variables were highlighted as particularly highly correlated with the first two principal components that explain 31% of the variance. A second PCA was performed with these variables. Its results are illustrated by Fig. 3.

Fig. 3. Left correlation circle for the 16 variables retained earlier. Right individual profiles according to the grade.

Fig. 3.a shows that the first principal component is highly positively correlated with Writing Duration and Pen Lift Duration for P2 and T2 (X27, X28, X63 and X64) and with Total Duration and mean Pen Lift Duration (X26, X33) for P2 and highly negatively correlated with writing speed for P1, P2, T1 and T2 (X19, X31, X55, X67). Therefore, the horizontal axis in Fig. 3 can be considered as a scale of rapidity and as an inverse scale of regularity and fluidity explains 38% of the variability. The second principal component is highly positively correlated with Writing Duration for P1 and T1 (X15 and X51) and highly negatively correlated with the ratio SNR/nbLog for T1 and T2 (X49 and X61). This axis is opposing rapidity and regularity for T1 and T2. Children from CE1 and CE2 write slowly and have longer pause duration compared to those from CM1 and CM2 whatever the type of isolated word and the nature of feedback, visuo-spatial or kinesthetic, involved by the task. Moreover, children from CE1, CE2 and CM1 take less time to write their first name or “tintin” than those of CP and CM2. However, a majority of these children from CE1, CE2 and CM1 have a greater regularity of their movements, they write with more application, than a majority of children from CP and CM2 when they are writing “tintin” whatever the conditions of execution : with or without vision. These results are consistent with the behaviour compiled in previous graphonomics researches [1], [9]. These first descriptive results and tests seem to confirm that, in the case of productions by children of isolated cursive words like their first names or imposed pseudo-first name like “tintin”, it could be possible to identify variables that significantly reflect their learning grades. Yet, the learning grade variable is correlated with the exposure time to handwriting apprenticeship and handwriting training and with the level of graphomotor control progress.
VII. CONCLUSION

The present work takes place as the preliminary feasibility step of an experimental longitudinal project and as a next step of the Script project [10] in which the formal basis was provided by the delta-lognormal model. It contributes to the completion of a series of studies showing that the sigma-lognormal model is sufficiently robust to be used to analyze various types of young children’s productions since kindergarten. Moreover, in the transversal experimental contexts explored, such type of work seems to indicate that the lognormal and classical variables which can be extracted from the pupils handwritten productions could be useful for the graphomotor evaluation of children productions. Lognormal variables like SNR and SNR/nbLog seem to be compact interesting candidates for such purpose because they can indifferently be used in the same algorithmic processing way to characterize various types of productions from simple strokes [11], [4], through constraint curvilinear or multilinear scribbles [5], [6] up to cursive words. In the real longitudinal context of the Guadeloupean experimentation that is beginning, our next challenges will be to study the potential power of expression of graphomotor progress of these variables on various types of graphomotor activities like simple strokes, scribbling, writing or drawing that will be realized by pupils from kindergarten, primary and middle school, when these productions are online captured thanks to classical digitizers or with new pen-devices like tablets.

REFERENCES

Kinematic Analysis of Fast Pen Strokes in Children with ADHD using the Sigma-lognormal Model

Patricia Laniel  
*Department of psychology*  
University of Montreal  
Montreal, Canada  
patricia.laniel@umontreal.ca

Nadir Faci  
*Department of electrical engineering*  
Polytechnique Montreal  
Montreal, Canada  
nadir.fac@polymtl.ca

Réjean Plamondon  
*Department of electrical engineering*  
Polytechnique Montreal  
Montreal, Canada  
rejean.plamondon@polymtl.ca

Miriam H. Beauchamp  
*Department of psychology and Ste-Justine Hospital Research Center*  
University of Montreal  
Montreal, Canada  
miriam.beauchamp@umontreal.ca

Bruno Gauthier  
*Department of psychology*  
University of Montreal  
Montreal, Canada  
bruno.gauthier@umontreal.ca

**Abstract**—This study aimed to examine whether the sigma-lognormal model derived from the Kinematic Theory of rapid human movements discriminates between the handwriting strokes produced by children with and without Attention Deficit Hyperactivity Disorder (ADHD). Twelve children with ADHD and 12 controls aged 8-11 years were asked to produce handwriting strokes on a digitizing tablet. The sigma-lognormal model was used to analyze the strokes. Children with ADHD demonstrated poorer motor control (SNR/nbLog), planning (t₀, D) and execution (nbLog, Sigma, Response time, Time delay) than controls did. The study provides preliminary evidence that the lognormal model may be useful as a tool for detecting motor skill problems in the context of ADHD.

**Keywords**— sigma-lognormal model, Kinematic Theory of rapid human movement, kinematic analysis, fine motor control, ADHD- associated problems, children, handwriting

## I. INTRODUCTION

Attention deficit/hyperactivity disorder (ADHD) is a neurodevelopmental disorder marked by patterns of inattention, hyperactivity and/or impulsivity that is inconsistent with developmental expectations and that interferes with daily functioning in a variety of settings (e.g., school, family) [1]. Many studies suggest that the motor skills of children with ADHD are poorer than those of their peers (e.g.,[2]). A delay in maturation of the prefrontal brain regions involved in executive functioning, attention, and motor planning has been observed in children with ADHD [3] and may underlie motor deficits. These deficits could also result from altered fronto-striato-cerebellar connectivity patterns, including dopamine pathways, which are responsible for the planning and modulation of movements [4,5]. Moreover, abnormal connectivity between the primary motor cortex and several other regions involved in motor and sensorimotor functioning (e.g., lower frontal gyrus, insular cortex, amygdala, putamen, pallidum) has been observed in children with ADHD and/or developmental coordination disorder (DCD) and may lead to both motor and attention deficits [6].

A recent meta-analysis indicates that children with ADHD have less legible and slower handwriting than their peers without ADHD [7]. Poor handwriting may impact on academic achievement, self-esteem, and quality of life [8,9]. It is thus essential to assess the motor and graphomotor skills in children with ADHD. Early detection of ADHD and of the problems associated with it may reduce their negative impact on development and help guide intervention strategies.

The study of graphomotor behavior through the kinematic analysis of handwriting movements yields online, precise, and objective quantitative measures of handwriting [10]. The results of the few studies that have used kinematic analysis with children with ADHD show that these children have greater difficulty planning and programming movements effectively than do typically developing children [11,12,13], as manifested by slower [13,14] and less consistent handwriting [11,15,16,17] and less precision and fluidity [11,13,15,18]. However, previous studies have been limited to examining global kinematic variables such as maximum speed, reaction time, total duration of movement, or degree of fluidity, which gives only partial information on the state of the neuromotor system.

Plamondon and colleagues [19] recently developed a novel approach based on the Kinematic Theory of rapid human movements [20,21] to study fine motor skills. They used the lognormal speed profile obtained from pen strokes executed on a digitizer as the model of ideal motor control. This technique can be used to assess graphomotor impairment caused by neurological problems [22] as well as the general state of the neuromotor system in children using global parameters (SNR, nbLog, SNR/nbLog). It can also provide new parameters describing the internal functioning of the neuromotor system, that is, a representation of its action plan (t₀, D) and execution (μ, σ, θ, θ₀, Mode, Median, Time delay, Response time, Asymmetry).

To produce a pen stroke, the representation of the desired movement and its motor program are first activated and selected and then converted into motor commands before the movement is finally produced adequately by the neuromuscular...
Effective programming of a fast pen stroke is reflected by the generation of motor commands to contract the agonist and antagonistic muscles involved in the movement [19]. This results in an asymmetric bell-shaped velocity profile, representing acceleration at the beginning of movement and deceleration at the end of movement [11,19]. According to the Kinematic Theory, lognormality occurs when rapid and spontaneous movements are sufficiently controlled and automated to require little or no visual feedback [20,21]. The generation of more complex movements thus amounts to concatenating and superimposing lognormal functions over time, exploiting the sigma-lognormal model [19,24].

In a typical experiment, the position of the pen is first recorded by a digitizer. Software programs then calculate the speed, then an optimization algorithm breaks down the movement into its lognormal components and extracts the parameters that best represent the pen's trajectory. Among the parameters obtained, the number of lognormal functions required to reconstruct the signal (nbLog) and the signal-to-noise ratio between the original profile and the reconstructed profile (SNR) are used to calculate the SNR/nbLog ratio. These variables are an index of the lognormality of the movements produced, which can be interpreted as an index of the fluidity of movement [25]. Using the lognormal approach, recent studies have shown that the signal-to-noise ratio (SNR) declines among older individuals [19], that nbLog is significantly higher for individuals with Parkinson's disease than for controls [22], and that young children (3 to 5 years old) tend toward lognormal patterns of pen strokes over time (SNR increased and nbLog diminished with the increase of fine motor control) [25]. Together, these results indicate that lognormal parameters (SNR and nbLog) are useful to describe the state of motor control in a variety of context.

In the present study we applied the sigma-lognormal model to examine the motor control of children with ADHD. We sought to determine whether the parameters obtained from sigma-lognormal modeling of fast pen stroke speed profiles would discriminate between strokes produced by children with and without ADHD. We predicted that children with ADHD would show poorer handwriting movement quality than controls and that this would be reflected in the lognormal parameters.

II. METHODOLOGY

A. Participants

Twenty-four children 8 to 11 years old (Mean age, 9.77 years) participated in the study. Children (n=12, M=9.57, SD=1.18 years, 8 males) in the ADHD group all had a medical diagnosis of ADHD and all took psychostimulant medication during school year. The majority (N=8) were taking methylphenidate (one in combination with clonidine and one in combination with guanfacine), and the others (N=4) were taking lisdexamfetamine (one in combination with guanfacine). They were asked not to take their medication on the day of the evaluation. They were recruited through ads posted on bulletin boards at the University of Montreal and on groups for parents of children with ADHD on social networks (e.g., Facebook). Children taking Strattera (atomoxetine) were excluded because of its long-lasting effect. Twelve children without ADHD recruited via day camps and through ads posted on social networks (e.g., Facebook) formed the control group (n=12, M=9.97, SD=1.33 years, 6 males). Exclusion criteria for the control group were: a diagnosed neurological condition, intellectual disability, developmental coordination disorder (DCD), psychotropic medication. No children were excluded.

B. Procedures

Parents agreed to have their child participate in the study and signed a written consent before the evaluation. Evaluation took place in a quiet room at University of Montreal. Children were asked to rapidly draw 30 straight single strokes on a digitizer with a stylus, one at a time, from a starting point to a target area indicated on a guide sheet. Strokes had to be more than 13 cm long. The go signal for the child to draw a stroke was a sound cue (at 1 kHz for 500 ms) emitted by an audio simulator. Movements were digitized at 200 Hz and recorded on a digitizer (Wacom Intuos2). The digitizer and simulator were synchronized with an in-house program called Sign@medic (for more details on the test, see [26, p.3]).

C. Extraction of the sigma-lognormal parameters

Invalid strokes were removed (automatically by the software if the stroke was started before the cue or manually by the examiner if the stroke was executed improperly). These behavioral rejects (versus software rejects, see below) represented 17.5% of the total number of strokes (157/897) (21% in the control group and 79% in the ADHD group). The ADHD group produced a significantly higher number of strokes leading to rejection. M = 10.33, SD = 8.08, than the control group, M = 2.75, SD = 5.01, t(22) = -2.76, p = .013. Children with ADHD often began their movement before the start cue.

The data were then filtered. The pressure exerted on the tablet indicated when participants lifted the pen, marking the end of data collection for a particular stroke. Disturbances in the signal due to lifting of the pen were eliminated. The speed profiles were then reconstructed with the sigma-lognormal model. This method allowed the extraction of eight lognormal parameters, namely, the number of lognormals per stroke (nbLog), SNR, t₀, D, μ, σ, θ, θ₀. Six other parameters were calculated based on these indicators (SNR/nbLog, Mode, Median, Time delay, Response time and Asymmetry) (see below for a description of each parameter).

After parameter extraction, some strokes were rejected due to software processing errors (software rejects). A total of 12 strokes (control group = 3; ADHD group = 9) were rejected because the SNR was less than 15 dB (meaning that the software was unable to reconstruct the trajectory adequately [24]) and 24 (control group = 17; ADHD group = 7) were rejected when t₀ was negative and μ was positive. Finally, 43 (control group = 24; ADHD group = 19) were rejected because D (movement amplitude) exceeded the maximum distance a child could reach. Strokes could be rejected for more than one reason. Thus, a total of 669 strokes (on 897) were kept for analysis. On average, for all the rejects (behavioral and software rejects), 3.17 strokes, SD = 2.25, were rejected for each child in the control group and 2.75 strokes, SD = 2.01, for
each child with ADHD. The difference was not significant, $t(22) = 0.46, p = .637$. Table I presents the number of strokes kept for analysis per participant.

<table>
<thead>
<tr>
<th>Participant</th>
<th>ADHD group</th>
<th>Control group</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30</td>
<td>29</td>
</tr>
<tr>
<td>2</td>
<td>22</td>
<td>36</td>
</tr>
<tr>
<td>3</td>
<td>26</td>
<td>41</td>
</tr>
<tr>
<td>4</td>
<td>23</td>
<td>29</td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>27</td>
</tr>
<tr>
<td>6</td>
<td>29</td>
<td>30</td>
</tr>
<tr>
<td>7</td>
<td>25</td>
<td>28</td>
</tr>
<tr>
<td>8</td>
<td>29</td>
<td>26</td>
</tr>
<tr>
<td>9</td>
<td>27</td>
<td>28</td>
</tr>
<tr>
<td>10</td>
<td>28</td>
<td>22</td>
</tr>
<tr>
<td>11</td>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>12</td>
<td>31</td>
<td>17</td>
</tr>
<tr>
<td>Total</td>
<td>328</td>
<td>341</td>
</tr>
</tbody>
</table>

**TABLE I. Number of strokes kept for analysis per participant**

D. Sigma-lognormal parameters

For each group, the median value of the following parameters was used in the analyses [20,21,27,28]. The first three parameters are global parameters and describe the general state of the neuromotor system:

- **SNR**: The signal-to-noise ratio between the original speed profile and the reconstructed speed profile, computed in decibels (dB). This is a measure of the quality of the sigma-lognormal reconstruction. Higher SNR indicates better reconstruction.

- **nbLog**: The number of lognormal functions required to reconstruct the signal. This parameter represents fluidity of movement by the participant. Higher nbLog indicates less fluid movement.

- **SNR/nbLog**: SNR/nbLog is used as a performance criterion. The capacity to reconstruct a movement speed profile with lognormals can be interpreted as an indicator of the quality of motor control since the lognormal speed profile corresponds to perfect motor control [19]. Higher SNR/nbLog indicates a movement closer to ideal lognormal behavior.

Other parameters describe the internal functioning of the neuromotor system. Two parameters represent the neuromotor action plan:

- **$t_0$**: Time required (in seconds) for the brain to produce a motor command. $t_0$ should not to be confused with the moment a sound cue is emitted ($t = 0$), nor with the response time (RT) measured by moment of movement onset. $t_0$ is the moment when a population of neurons sends a motor command. In other words, $t_0$ occurs after the start cue, given that the central nervous system must perceive the sound cue before sending the command, but before RT, given that a delay is necessary for the motor command to be transmitted and for enough muscle cells to be recruited to begin the movement effectively [28]. Longer $t_0$ are indicative of longer delays before a motor command is emitted and, therefore, longer time to perceive the stimulus and prepare the command.

The other parameters are related to motor program execution:

- **$\mu$**: Logtime delay, the time required to reach the middle of movement distance on a logarithmic scale. It can be associated with the general response speed of the neuromuscular system to a command.

- **$\sigma$**: Logresponse time, the response time on a logarithmic scale of the neuromuscular system’s response to a command. Sigma ($\sigma$) is used to compute an estimate of movement duration.

The following parameters are also related to motor program execution and are intrinsic characteristics of each lognormal [23,27].

- **$\theta_s$**: The starting angular direction of each stroke in radian.

- **$\theta_e$**: The ending angular direction of each stroke in radian.

- **Mode**: The Mode ($M$) is the time at which the amplitude of the impulse response reaches its maximum value.

\[
M = t_0 + e^{(\mu-\sigma^2)}
\]  

- **Median**: The Median ($m$) is the time when the integral under curve reaches its half-value. This indicates when half the action is executed. In the current study, the Median means half the distance. A smaller Median means that half of the movement is reached faster.

\[
m = t_0 + e^{\mu}
\]  

- **Time delay**: The Time Delay ($\bar{\tau}$) is a global evaluation of the rapidity of a given system to react to a command. A smaller Time delay means that the reaction to the command is faster.

\[
\bar{\tau} = t_0 + e^{(\mu+\frac{\sigma^2}{2})}
\]
• Response time: The response time (s) represents the impulse response of a neuromuscular system. \( s^2 \) is a measure of the spread of the impulse response over time. A smaller response time means that the movement time is smaller.

\[
s = (t - t_0) + \sqrt{(e^{s^2} - 1)}
\]  

(4)

• Asymmetry: Asymmetry \((A_e)\) is related to the shape of the curve. Higher Asymmetry means that the Mode (M) moves to the left with a smaller velocity peak, but without changing the Median position. It means that the peak will be reached faster, but it will be smaller and movement time will be longer.

\[
A_e = 1 - e^{-\sigma^2}
\]  

(5)

E. Statistical analysis

To compare the quality of motor control of single strokes between children with and without ADHD, we conducted t-tests for independent measures on the parameters obtained from the sigma-lognormal analysis. The analysis is conducted on all strokes by group (and not on the average for each child). Bonferroni corrections for multiple comparisons were applied. Since six parameters (SNR/nbLog, Mode, Median, Time delay, Response time and Asymmetry) are calculated from the other eight \((t_0, D, \mu, \sigma, \text{Theta Start, Theta End, SNR, nbLog})\), we applied Bonferroni corrections for eight parameters (significance threshold is p<0.00625).

III. RESULTS

Two-sided chi-square tests revealed no significant difference between the clinical and control groups on sex, \(X^2(2, N=24) =0.686, p>.05\), and handedness, \(X^2(2, N=24) =3.429, p>.05\). Independent measures t-tests showed no significant difference in age, (ADHD group M= 9.57, SD = 1.18; control group M= 9.97, SD=1.33), t (22) = 0.77, p>.05.

The results revealed a significant between-group difference on SNR/nbLog, indicating significantly poorer quality of motor control in the ADHD group. Results are presented in Table II. Mean nbLog was significantly higher for the ADHD group, which indicates that more lognormals were required to reconstruct the pen stroke signal and that the movements were less fluid. A significant between-group difference was found in terms of \(t_0\), which was greater for the ADHD group, indicating a longer delay before motor command transmission (see Table II).

In addition, the D parameter was significantly smaller for the ADHD group, which reflects a smaller movement amplitude in their action plans. Sigma (\(\sigma\)) and Response time parameters were also significantly greater in the ADHD group, suggesting that strokes made by children with ADHD were slower than those of the children in the control group. Time delay was significantly greater in the ADHD group, indicating that the neuromotor system of children with ADHD took more time to react to the command than children in the control group. No significant difference (with Bonferroni correction) was observed on the other parameters (see Table II).

TABLE II. RESULTS OF T-TEST FOR INDEPENDENT MEASURES BETWEEN ADHD GROUP AND CONTROL GROUP ON SIGMA-LOGNORMAL PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>ADHD (M,S.D)</th>
<th>Control (M,S.D)</th>
<th>Degrees of freedom</th>
<th>t</th>
<th>Sig.</th>
<th>Cohen’s d(^f)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNR</td>
<td>26.22 (2.67)</td>
<td>26.53 (2.77)</td>
<td>653</td>
<td>1.45</td>
<td>0.148</td>
<td>-</td>
</tr>
<tr>
<td>nbLog</td>
<td>4.94 (2.40)</td>
<td>3.79 (1.75)</td>
<td>653</td>
<td>-7.05</td>
<td>&lt;0.0001*</td>
<td>0.55</td>
</tr>
<tr>
<td>SNR/nbLog</td>
<td>6.55 (3.15)</td>
<td>8.09 (2.90)</td>
<td>653</td>
<td>6.46</td>
<td>&lt;0.0001*</td>
<td>0.51</td>
</tr>
<tr>
<td>(t_0)</td>
<td>0.69 (0.77)</td>
<td>0.41 (0.69)</td>
<td>2854</td>
<td>-10.07</td>
<td>&lt;0.0001*</td>
<td>0.38</td>
</tr>
<tr>
<td>D</td>
<td>62.83 (74.70)</td>
<td>73.29 (84.36)</td>
<td>2854</td>
<td>3.51</td>
<td>&lt;0.0001*</td>
<td>0.13</td>
</tr>
<tr>
<td>(\mu)</td>
<td>-1.55 (0.76)</td>
<td>-1.64 (0.74)</td>
<td>2854</td>
<td>-2.27</td>
<td>0.023(^b)</td>
<td>-</td>
</tr>
<tr>
<td>(\sigma)</td>
<td>0.34 (0.28)</td>
<td>0.31 (0.25)</td>
<td>2854</td>
<td>-3.17</td>
<td>0.002*</td>
<td>0.11</td>
</tr>
<tr>
<td>Teta start</td>
<td>-0.23 (6.51)</td>
<td>-0.86 (5.99)</td>
<td>2854</td>
<td>-2.65</td>
<td>0.008(^b)</td>
<td>-</td>
</tr>
<tr>
<td>Teta end</td>
<td>-0.51 (13.04)</td>
<td>-0.88 (9.23)</td>
<td>2854</td>
<td>-0.84</td>
<td>0.399</td>
<td>-</td>
</tr>
<tr>
<td>Mode</td>
<td>0.25 (0.22)</td>
<td>0.24 (0.20)</td>
<td>2854</td>
<td>-1.72</td>
<td>0.085</td>
<td>-</td>
</tr>
<tr>
<td>Median</td>
<td>0.27 (0.21)</td>
<td>0.25 (0.19)</td>
<td>2854</td>
<td>-2.40</td>
<td>0.016(^b)</td>
<td>-</td>
</tr>
<tr>
<td>Time delay</td>
<td>0.29 (0.21)</td>
<td>0.26 (0.19)</td>
<td>2854</td>
<td>-3.05</td>
<td>0.002*</td>
<td>0.15</td>
</tr>
<tr>
<td>Response time</td>
<td>0.38 (0.43)</td>
<td>0.33 (0.30)</td>
<td>2854</td>
<td>-3.46</td>
<td>0.001*</td>
<td>0.13</td>
</tr>
<tr>
<td>Asymmetry</td>
<td>0.33 (1.82)</td>
<td>0.20 (3.07)</td>
<td>2854</td>
<td>-2.36</td>
<td>0.018(^b)</td>
<td>-</td>
</tr>
</tbody>
</table>

\(^a\) Significant with Bonferroni corrections for multiple comparisons (p < 0.00625)  
\(^b\) Significant at p < .05  
\(^c\) Effect size is considered large when Cohen’s \(d\) is at about .80, medium at about .50, and small at .20 [29].
SNR parameter could not discriminate between groups as the required value to be reached by the software was fixed (>15). The number of lognormals required to reconstruct the signal thus seem to best discriminate between groups.

The amplitude of movement D was significantly smaller in the ADHD group; that is, more motor commands were needed to produce a stroke, but of a smaller amplitude, to cover the same distance as typically developing children. This might reflect a problem with programming movements in children with ADHD [5]. However, more research is needed to better understand the mechanisms underlying this between-group difference in the D parameter. This could be done by studying more complex movements.

The time required for the neuromuscular system to emit a motor command, $t_0$, was significantly higher in the ADHD group. This may reflect a lag in the perception of the start cue or diminished motor planning [34] among children with ADHD. The latter hypothesis is supported by previous studies, where children with ADHD needed more time to plan their actions [35,36].

It appears that in addition to prolonged movement planning in children with ADHD compared to control children, the condition is also associated with increased movement time. As such, Sigma ($\sigma$) and Response time were significantly greater for children with ADHD than for controls. This shows that children with ADHD tend to take more time to execute their movements. The Time delay was also greater in the ADHD group than in the control group, indicating that the neuromuscular system of children with ADHD may be slower to react to a command than that of control children. These results are consistent with other studies that show slower handwriting in children with ADHD compared to controls [13,14].

This study is limited in that it only included 24 children (12 with ADHD). Since the group was small, comorbidities associated with ADHD (e.g., developmental coordination disorder) were not controlled for. Futures studies should include larger sample of ADHD children and control for comorbidities to insure a better generalization of the results. Furthermore, the type of medication taken by the children with ADHD was not controlled (except for Strattera, which was excluded). Different molecules have different modes of action and duration effects. Though the children did not take their medication the day they were evaluated, the long-term intake of psychostimulants to treat ADHD could have a protective effect on brain development (particularly the cerebellum) [37]. Hence, despite medication cessation, the brains of children having taken medication might differ from those of drug-naïve children.

V. CONCLUSION

In this study, we aimed to examine whether the sigma-lognormal model derived from the Kinematic Theory of rapid human movements could discriminate between the handwriting strokes produced by children with and without ADHD. As we predicted, a significant difference was found between children with and without ADHD on many lognormal parameters: SNR/nbLog, nbLog, $t_0$, D, $\sigma$, Time delay and Response time. These results suggest that ADHD may be associated with deficits in graphomotor skill. The sigma-lognormal analysis shows promise as it offers the possibility to study the internal functioning of the neuromotor system of children with ADHD.

REFERENCES


Copyright © 2018 by CENPARMI, Concordia University
Lognormality In Children With Mild Traumatic Brain Injury: A Preliminary Pilot Study

Nadir Faci
Dept. of Electrical Engineering
École Polytechnique Montréal
Montréal, Canada
nadir.faci@polymtl.ca

Naddley Désiré, Ph.D.
Dept. of Neurosciences
Alberta Children’s Hospital
Alberta, Canada
naddley.desire@ahs.ca

Miriam H. Beauchamp, Ph.D.
Dept. of Psychology & Ste-Justine Research Center
Université de Montréal
Montréal, Canada
miriam.beauchamp@umontreal.ca

Isabelle Gagnon, Ph.D.
School of Physical and Occupational Therapy/Pediatrics
McGill University
Montréal, Canada
isabelle.gagnon8@mcgill.ca

Réjean Plamondon, Ph.D.
Dept. of Electrical Engineering
École Polytechnique Montréal
Montréal, Canada
rejean.plamondon@polymtl.ca

Abstract—This preliminary pilot study examines whether the sigma-lognormal model proposed by the Kinematic Theory of rapid human movements is able to detect a difference between the handwriting strokes produced by individuals with mild traumatic brain injuries (mTBI) 1 and 3 months post-injury. 90 participants, aged between 6.00 and 17.99 years, who sustained mTBI were asked to produce handwriting strokes on a digitizing tablet. Strokes were analyzed using the Sigma-Lognormal model. This model showed significant differences between the two sets of strokes produced 1- and 3-months post-injury by the participants, with the quality of strokes improving over time. Only four participants did not show any significant changes. Findings from this preliminary pilot study warrant further research into the potential benefits of using the Sigma-Lognormal model to assess the integrity of neuromotor systems following pediatric mTBI.

Keywords—mild traumatic brain injuries (mTBI), children, fine motor control, sigma-lognormal model, Kinematic Theory of rapid human movement, kinematic analysis, handwriting

I. INTRODUCTION

A. Background

Pediatric traumatic brain injury (TBI) is a public health burden and the leading cause of disability worldwide [1]. Each year, millions of children sustain TBI, with mild traumatic brain injuries (mTBI) and concussions accounting for more than 90% of all TBI cases. Mild TBI can result in detrimental consequences for children and potentially interfere with development, impacting a child’s academic, cognitive, psychological, social and behavioural functioning. Children with mTBI often report post-concussive symptoms (PCS) which include somatic (e.g., headache, dizziness), cognitive (e.g., inattention, forgetfulness), and affective (e.g., irritability, dysphoria) complaints. Previous studies have shown that 15-30% of children with mTBI continue to experience PCS for several months following injury, which in turn can result in functional deficits and declines in quality of life [2] [3] [4]. These challenges have led multiple governmental organizations, such as the World Health Organization and the Centers for Disease Control and Prevention, to call for additional research to inform clinical care and improve outcomes in children with mTBI [5] [6].

In light of the current need for advancing knowledge in mTBI, extensive research is also required to study brain-behaviour disturbances following mTBI and to examine how these changes may be associated with functional impairments. Consequences of pediatric mTBI may differ from those observed in the adult population, as there is evidence that childhood brain injuries can result in poorer outcome in some children due to the vulnerability of the developing brain [7]. The cognitive, motor and social functions that develop during childhood and adolescence strongly depend on the integrity of particular brain structures and networks at critical developmental stages. Research is therefore imperative to identify children who may be at risk of developing persistent problems.

Although the majority of children and adolescent with mTBI are expected to have favorable long term outcome, a subset may also experience cognitive and motor difficulties in addition to presenting with PCS. Specifically, problems with attention, processing speed and executive functioning have been reported within the first few months [8] [9]. The investigation of motor functioning in children following mTBI has been more scarce. Only a limited group of studies have reported difficulties with complex and dynamic gross motor skills that would go undetected with standard neurological testing. Studies to date do show that although some motor deficits (e.g., balance) tend to diminish over time after injury, many difficulties still remain present several months after mTBI, even after PCS have resolved [10] [11]. This is an important finding to consider as children are often recommended to gradually return to daily activities (i.e., school, physical activities) after being asymptomatic without further assessment or intervention [12]. The investigation of fine motor skills following mTBI has also been limited. On one hand, fine motor difficulties may impact a child’s visuo-motor processing speed which can result in problems with producing written work quickly [13]. On the
other hand, children displaying fine motor difficulties may also require additional time to complete school assignments as they continue to recover from mTBI. However, there is currently a lack of accurate and developmentally appropriate tools to sensitively assess fine motor skills after mTBI.

In the current study, we aimed at investigating fine motor control in children and adolescents with mTBI using a handwriting stroke task that can extract specific objective parameters to obtain information about the integrity of the neuromuscular system after pediatric mTBI. We hypothesized that children and adolescents with mTBI would display atypical fine motor control as reflected by poorer performance at an earlier stage (1 month post-injury) of recovery than at a later stage (3 months post-injury).

B. The Kinematic Theory and Lognormality

The Kinematic Theory stipulates that to make a simple rapid movement referred to as a stroke, a neuromuscular network has to plan and execute the movement. To execute this movement, numerous sub-systems are recruited. In the case of a simple stroke, the sub-systems consist of the agonist and antagonist systems. The first is responsible of starting and accelerating the movement. The second sub-system is in charge of the deceleration and the ending of the movement. According to this theory, when a person is in control of his movement, the velocity of his movement will tend to be closer to lognormality. It means that each subsystem involved in the movement will generate a lognormal velocity curve. In the case of an ideal rapid pointing movement, it should be made up of two lognormals, and it can be analyzed by the Delta-Lognormal model [14] [15] [16] [17] [19]. As it can be seen in Plamondon and all [18], children tend to get closer to the Delta-Lognormal with aging, but during the adolescence, they tend to use more than two lognormals in their movement. To address this issue, the Kinematic Theory proposes the Sigma-Lognomal model that can fit the complex movement.

The Sigma-Lognormal model provides an approach to extract parameters that can describe the state of the neuromuscular system. The three following parameters describe the general state of the neuromotor system: SNR, nbLog and SNR/nbLog. It also extracts parameters to describe the planning of the movement: θ and D. Those two parameters describe the input command of the system. The other parameters describe the execution of the movement: μ, σ, θ0, θc. Some of those parameters are more likely to be significant, as for example, the SNR, the nbLog and the SNR/nbLog have more impact because they reflect the Lognormality of the participant. As it has been shown in previous research [18] [21] [22], those parameters are most likely to discriminate between a clinical group and a control group.

In this research, the Sigma-Lognormal model was used to fit the velocity profile of participants, because the participants are adolescents, and even if they tend towards lognormality, they have not reached it yet. In addition, the participants in the current study sustained concussion, therefore they are hypothetically moving away from lognormality, so they may tend to make more than two lognormals per stroke. Using this model, the 1 month post-injury results for each participant will be compared to their 3-month post injury results. It was expected that the quality of handwriting movement would change over time with recovery.

II. METHODS AND ANALYSES

A. Design

This is a sub-study of a larger prospective, longitudinal, multicenter cohort study (CDE Study, Gagnon et al., 2014) designed to collect a standard set of common data element measures for children and adolescents in pediatric mTBI follow-up programs across Canada and explore relations between initial presentation, personal factors, and 6-month outcome post-injury in terms of PCS, motor, cognitive and social functioning.

B. Participants

Participants included children and adolescent who presented to the ED of two tertiary care pediatric hospital (i.e., Montreal Children’s Hospital and CHU Sainte-Justine). 90 children and adolescent were initially recruited to the sub-study, but complete data was only available in 32 participants. Inclusion criteria were 6.00-17.99 years of age, having sustained a mTBI, as defined by the WHO task force [1], suffered the initial injury in the previous 28 days, are proficient in either French or English. Exclusion criteria included Glasgow Coma Scale score below 13, multi-system injury requiring hospitalization, operating room or procedural sedation, neurosurgical intervention, intubation, or intensive care unit admission required, intoxication at time of ED presentation as per clinical judgment, significant developmental delays resulting in communication difficulties that could impact a child’s ability to complete testing.

C. Procedure

Recruitment was done on a voluntary basis. Children were enrolled a) from the ED or b) from the mTBI follow-up program in the respective institution. Families were introduced to the study by the mTBI clinic coordinator after confirmation of their diagnosis. Families who agreed to participate were contacted by a study coordinator who explained procedures, screened for eligibility and obtained written consent. Information and details regarding participant’s demographics, past medical and developmental history, injury characteristics, and acute symptoms were then collected at enrolment by a research assistant or nurse. Participants who were recruited as part of the larger study underwent motor and cognitive assessments at 2-weeks/1/month, 3-months and 6-months post-injury. However, the stroke task was administered only at the 1-month and 3-month timepoints.

D. Data extraction and Analyses

1) Data extraction

In this research, each participant had to draw fast single strokes, one at a time. After the test, every participant should have produced 30 valid strokes. The trials were recorded at 100 Hz using a tablet digitizer (Wacom Intuos2). Every stroke had to begin from a starting point located at the middle of a guide sheet, and to end at one of the sides of the sheet. The direction side was set depending of the laterality of the participant. The guide sheet is shown in figure 1. The participants start drawing the stroke when they see a green light signal displayed by a
stimulator. The stimulator displayed the cue stimulus after a random time set by homemade software called Sign@medic. This software manages and synchronizes the digitizer and the stimulator. From the 90 children who took the test, 56 of them did not execute the test properly. Therefore, their data was rejected for the purposes of the pen stroke analyses. That is, every stroke that began before the cue was displayed was automatically removed by the software. Data rejected in this way was counted as Behavioral reject. For the 34 children included in the analyses, 2351 strokes were collected (1150 for participant at 1 month and 1201 for those at 3 months post-injury), and approximately 34 strokes per participant were performed to obtain 30 valid strokes. (For more details on the test, see [23].)

Furthermore, the original data were filtered to segment each stroke. The time when the cue was displayed indicates the beginning of the trial, and the moment the pressure returns to 0 fixes the end of the trial (the pen could then be lifted from the digitizer). If the velocity of the movement is not null at the end of the trial, this means that the pen was lifted before the stroke was finished. In this case, the stroke was not executed properly and it was rejected. Those rejected strokes were counted as Behavioral rejects. There was a high number of Behavioral rejects at both 1 and 3 months post-injury. 645 strokes (27.44%) were rejected for both timepoints (307 (26.70%) at 1 month and 338 (28.14%) at 3 months). The average at 1 month was M = 9.03, SD = 5.24, and M = 9.94, SD = 5.34 at 3 months post-injury. There was no difference between the two timepoints (p = 0.48).

Then, the velocity profiles were extracted using Script Studio [24] [25] [26] [27]. Script Studio is a software that extracts the parameters of the Sigma-Lognormal model. Eight parameters can be extracted. The first two are the number of Lognormal per stroke (nbLog) and SNR. Those parameters are global parameters for each stroke. The following six parameters are related to each Lognormal composing the movement: t0, D, μ, σ, θ, and ϑ0.

After the extraction, 67 strokes (2.85%) were rejected due to software processing errors. Those rejected strokes are named Software rejects. Two criteria were used to reject strokes, first, an SNR lower than 15 dB means that the stroke was not correctly reconstructed by the software, 37 trials (1.57%) were rejected following this criterion (23 (2.00%) at 1 month and 14 (1.17%) at 3 months). Second, if a t0 is negative and a μ is positive, the stroke is rejected. 31 trials (1.32%) were rejected following this criterion (20 (1.74%) at 1 month and 11 (0.92%) at 3 months). One stroke was rejected for both criteria. There was no significant difference between the two timepoints (p = 0.17). Two participants had fewer than 15 valid strokes for their 1 month trial block, therefore both participants were excluded from the analysis. 1567 strokes were finally used for the statistical analysis.

2) Sigma-Lognormal parameters

For each child, the mean value of each parameter was calculated. The following three parameters describe the global state of the neuromotor system: [14] [15] [17] [24].

- SNR: The signal-to-noise ratio between the original speed profile and the reconstructed speed profile, computed in decibels (dB). This is a measure of the quality of the sigma-lognormal reconstruction. Higher SNR indicates better reconstruction.

- nbLog: The number of lognormal functions required to reconstruct the signal. This parameter represents fluidity of movement by the participant. Higher nbLog, indicates less fluid movement.

- SNR/nbLog: SNR/nbLog is used as a performance criterion. The capacity to reconstruct a movement’s speed profile with lognormals can be interpreted as an indicator of the quality of motor control since the lognormal speed profile corresponds to perfect motor control [18]. Higher SNR/nbLog indicates that the movement is closer to ideal lognormal behavior.

The other parameters are an index of the internal functioning of the neuromotor system. The two following parameters represent the neuromotor action plan:

- t0: Time required (in seconds) for the brain to produce a motor command. t0 should not to be confused with the moment a sound cue is emitted (t = 0) nor with the response time (RT) measured by moment of movement onset. t0 is the moment when a population of neurons sends a motor command. In other words, t0 occurs after the start cue, given that the central nervous system must perceive the sound cue before sending the command, but before RT, given that a delay is necessary for the motor command to be transmitted and for enough muscle cells to be recruited to begin the movement effectively [24]. Longer t0 are indicative of longer delays before a motor command is emitted and, therefore, longer time to perceive the stimulus and prepare the command.

- D: The amplitude of the movement associated with each motor command, in millimeters. Greater D indicates greater amplitude of the movement associated with a command. Dtot is the total amplitude of the movement.

The other parameters are related to motor program execution:

- μ: Logtime delay, the time required to reach the middle of movement distance on a logarithmic scale. It can be associated with the general response speed of the neuromuscular system to a command.

Fig. 1. The guide sheet used in this study.
• $\sigma$: Logresponse time, the response time on a logarithmic scale of the neuromuscular system’s response to a command. Sigma ($\sigma$) is used to compute an estimate of movement duration.

The following parameters are intrinsic characteristics of each lognormal and related to motor program execution. [20] [17].

• $\theta_s$: The starting angular direction of each stroke in radian.
• $\theta_e$: The ending angular direction of each stroke in radian.

E. Statistical analysis

The analysis was performed for every participant separately. The results at 1 month post-injury were compared to those at 3 months for each child. In order to compare the evolution of each child over time, a t-test for independent groups was applied. Bonferroni correction was calculated using the 8 originals parameters. To have a significant difference, the $p$-value has to be less than 0.00625 ($\alpha = 0.05/8 = 0.00625$).

III. RESULTS

Of the 32 valid participants, 28 of them have significant differences for at least 1 parameter when comparing their result at 1 and 3 months post-injury. The other 4 children do not have any significant difference between their parameters. The following table presents the most significant difference by parameters. There are 17 participants who had a significant difference, with the Bonferroni correction, for the SNR/nbLog. From the 32 participants, 53% of them had a significant difference over time for the SNR/nbLog parameter. As it can be seen in the table below, 11 participants (34% of all participants) have significant difference between their 1 and 3 months results for the total amplitude of movement (Ddot). Concerning the number of lognormals (nbLog), 12 participants (38% of all participants) have a significant difference between their 1 and 3 months results post-injury. Those results show that the quality of the performance on the stroke task evolves over time. Further analysis is needed to determine if they tend to get closer or farther to Lognormality.

### TABLE I: MOST SIGNIFICANT PARAMETERS THAT DISCRIMINATE BETWEEN THE 1 MONTH AND 3 MONTHS RESULTS POST-INJURY

<table>
<thead>
<tr>
<th>Parameters</th>
<th>D_{1m}</th>
<th>nbLog</th>
<th>SNR/NbLog</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 month group higher</td>
<td>4</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>3 months group higher</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Total of difference</td>
<td>11</td>
<td>12</td>
<td>17</td>
</tr>
<tr>
<td>ratio (all participants)</td>
<td>34%</td>
<td>38%</td>
<td>53%</td>
</tr>
<tr>
<td>ratio (1 month)</td>
<td>36%</td>
<td>42%</td>
<td>59%</td>
</tr>
<tr>
<td>ratio (3 months)</td>
<td>64%</td>
<td>58%</td>
<td>41%</td>
</tr>
</tbody>
</table>

IV. DISCUSSION

The primary purpose of this preliminary pilot study was to examine fine motor control parameters in children and adolescents who sustained a mTBI. Consistent with our initial expectations, children with mTBI are likely to have reduced fine motor control, as reflected by changes in the sigma-lognormal parameters between 1-month and 3-months post-injury. These findings are consistent with previous studies that have shown that pediatric mTBI may be associated with fine motor deficits, such as reduced fine motor speed and dexterity and graphomotor processing speed [13].

The major limitation of this preliminary study is the rejection of the 56 children due to the lack of compliance to instructions. Specifically, as many rejected participants did not respect the instructions (e.g., stroke should end and be held on the tablet for 1 second), the final velocity was not null and the reconstruction with the Sigma-Lognormal model was not possible. However, from a qualitative perspective, participants appeared to display a tendency to disregard instructions and rules, which may be suggestive of lack of impulse control and disinhibition. Problems with executive control are not uncommon in mTBI [8], and should be taken into account when interpreting the results in future studies, as they may be indicative of subtle neurocognitive and neuromuscular deficits.

Future studies should add a detector to the homemade software Sign@medic, to check if the pen is lifted at the end of the stroke and the velocity is null. The second solution is to ensure that research assistants are extensively trained to notice administration problems to attain a more standardized administration. Moreover, to better understand and examine the clinical value of the sigma-lognormality parameters, it will be essential to include a control group (with similar demographics) look at group differences between children with mild traumatic brain injury (mTBI) and children without mTBI. This control group could consist of typically developing children or children with orthopedic injuries. The age will be consider as a co-variable in our statistical analyses. The pilot data are promising enough to justify future research using stronger methodologies (e.g., control group, additional cognitive and motor measures) to more clearly characterize the process by which fine motor changes occur. Lastly, the preliminary results provide interesting clinical information as it may be useful to add measures of motor stroke to neurobehavioural assessments as a screening for outcome after mTBI.

REFERENCES


Part II: Movement time and control, Biological Cybernetics, 72(4), 309-320. doi:10.1007/BF00202786

Part III: Kinetic outcomes. Biol Cybern 78: 133–145


A common framework to evaluate Parkinson’s disease in voice and handwriting

Cristina Carmona-Duarte
Instituto Universitario para el Desarrollo Tecnológico y la Innovación en Comunicaciones
University of Las Palmas de G.C.
Las Palmas de G.C., Spain
carmona@idetic.eu

Miguel A. Ferrer
Instituto Universitario para el Desarrollo Tecnológico y la Innovación en Comunicaciones
University of Las Palmas de G.C.
Las Palmas de G.C., Spain
miguelangel.ferrer@ulpgc.es

Pedro Gomez-Vilda
Neuromorphic Speech Processing Lab, Center for Biomedical Engineering,
Universidad Politécnica de Madrid
Madrid, Spain
pedro@fi.upm.es

Arend. W. A. Van Gemmert
Fine Motor Control & Learning Laboratory
School of Kinesiology,
Louisiana State University, Baton Rouge, LA 70803, USA
gemmert@lsu.edu

Réjean Plamondon
Department of Electrical Engineering
École Polytechnique, Université de Montréal
Montreal, Canada
rejean.plamondon@polymtl.ca

Abstract— Parkinson’s disease is manifested as well in handwriting as in voice. Previous researches have carried out different procedures to estimate the dysfunctions of the illness in voice and handwriting separately. This paper proposes one parameter to evaluate the influence of the illness on both voice and handwriting as the symptoms affecting both has a common origin. Specifically, the parameter proposed is based on the Kinematic Theory of rapid human movements. It allows to quantify the deficits caused by Parkinson’s disease in both handwriting and voice. The velocity profile obtained to characterize voice between the first and second formant is computed by a spatio-temporal approximation. In handwriting, the velocity profile is obtained from the sampled positions of the pen on a digital tablet. Once the velocity profile is derived, it is transformed to fit into the lognormal model in which similarities between voice and handwriting has been found for performance of these tasks by Parkinson’s patients. The experiments with different databases of voice and handwriting recorded from different patients in different labs display encouraging results.

Keywords—Sigma-lognormal model; kinematic theory of rapid movements; articulation; Parkinson; Voice; handwriting.

I. INTRODUCTION

Parkinson’s disease (PD) is a neurodegenerative disease that has symptoms which manifest in deficiencies affecting both handwriting and voice. PD symptoms are the result of a dopaminergic deficiency characterized by the presence of two or more cardinal motor symptoms (i.e., bradykinesia, rest tremor, rigidity, and postural disturbances) [1]. PD is a slow progressive disease with a long duration where clinical treatment and rehabilitation can help to improve the quality of life. Therefore, an early diagnosis and continuous monitoring of the effects of treatments are important. PD has been monitored by recording handwriting and processing its signals and recording the voice and processing its signals.

In handwriting changes of the kinematic aspects of movements and the analysis of in-air movements have been proposed as useful methodologies to monitor and diagnose early the disease. [2]–[5]

On the other hand, in voice processing there have been recent studies about the evaluation of the voice of individuals with Parkinson’s disease using the variability of the pitch, the voice rate and pausing [6]–[9].

In the current paper, we propose using a common methodology to analyse handwriting and voice. As both voice and handwriting are complex tasks involving the neurological and muscular system, in which muscular system is synchronized to communicate an idea through sentences, words, and letters. When a person writes on a Wacom or other tablet device that allows capturing the temporal position of the pen during handwriting, the velocity information can be obtained and analysed. To this end, the Kinematic Theory of rapid human movements [10] is applied to divide a complex movement in simple movements (strokes), each one is modelled by a sigma-lognormal function and the complex movement is then the summation of all the parametrized sigma-lognormals. This theory has been applied in different fields to assess movements as it pertains to handwriting variations across time [11], [12]. Thus this model has allowed us to specify diagnostic systems for neuromuscular disorders [13], [14] and the assessment of risk factors for stroke risk [15].

In voice production, the resonating cavities modifiable by the articulatory organs allow the energy of the voice signal to be concentrated at certain frequencies (formants), due to oropharyngeal tract resonators. It is well known that the formants are related with the tongue-yaw reference centre (JTRC) [16]. Also, the JTRC is related with the first and second formant [17]. In recently studies [18]–[20], a
relationship between the formants and the lognormal model have been shown.

The present work is intended to compare voice and handwriting production and the derived velocity signals as both tasks can be captured by a common parameter which can be validated to detect Parkinson disease. This validation of a common parameter in both domains reinforces earlier findings that could assist to develop more reliable diagnoses.

II. FROM VOICE OR HANDWRITING TO VELOCITY

In order to get the kinematic signal from the voice, formants are calculated. The formant estimation is obtained by adaptive inverse filtering [21]. This computation has been carried out with a resolution of 15 Hz using an 8-order prediction-error lattice-ladder filter [22].

The first formant F1 is related to the longitudinal movement and the second formant F2 is related to the vertical movement. These movements can be correlated with the formants positions in the plane F1 vs F2 [16] as:

$$\Delta x = [c_{11} \Delta F1 + c_{12} \Delta F2]$$

where \( \Delta x \) and \( \Delta y \) are the relative displacement from the previous position of the JTTC. \( c_{ij} \) are the weights of the combination matrix.

Once the displacement is calculated, the velocity signal \( \bar{v}(t) \) is estimated as:

$$\bar{v}(t) = \frac{\sqrt{\Delta x(t)^2 + \Delta y(t)^2}}{\Delta t}$$

In the case of handwriting, the patient is asked for write down on a tablet, in our case a WACOM tablet which record the position \( x(t) \) and \( y(t) \) of the pen on the tablet 200 times per second. As a result, the pen velocity can be worked out as:

$$\bar{v}_{handw}(t) = \sqrt{(x(t) - x(t-1))^2 + (y(t) - y(t-1))^2}$$

Both voice signals are normalized to mm/s.

III. FROM VELOCITY TO SIGMA-LOGNORMAL MODEL

This section is devoted to parametrize the velocity profile of both voice and handwriting signal using the Kinematic Theory of rapid movements framework. In this way, the velocity profile \( \bar{v}(t) \) can be modeled as a linear combination of monomodal lognormals [23] as follows:

$$\bar{v}(t) = \sum_{j=1}^{M} \bar{v}_j(t; \tau_j, \mu_j, \sigma_j^2)$$

where the velocity profile of each stroke \( \bar{v}_j(t) \) is defined as:

$$\bar{v}_j(t; \tau_j, \mu_j, \sigma_j^2) = \frac{D_j}{\sigma_j \sqrt{2\pi}} \exp \left\{ \frac{-(t - \tau_j)^2}{2\sigma_j^2} \right\}$$

with \( t \) the basis of time, \( \tau_j \) the time of stroke occurrence, \( D_j \) the amplitude of each stroke, \( \mu_j \) the stroke time delay and \( \sigma_j \)

the stroke response time, both on a logarithmic time scale.

Based on the facts that Parkinson patients perform shorter movements and have some difficulties to plan next movements, among all of the expected effect of the disease on the lognormal decomposition is a lower stroke logresponse time \( \sigma_j \) [4], [19]. Therefore, we propose as a common parameter to detect Parkinson disease in both voice and handwriting, the averaged the stroke logresponse time \( \bar{\Delta \sigma} \) as defined in (6):

$$\bar{\Delta \sigma} = \frac{\sum_{j=1}^{M} |\sigma_j - \sigma(\tau_j)|}{M}$$

It is expected that people with some degree of Parkinson disease will show a lower \( \bar{\Delta \sigma} \) than healthy people.

IV. METHOD

A. Voice Database

A database comprising phonations from five PD patients selected by neurologists and five control subjects was used in this study. The subjects, with aged from 52 to 78 years old, were diagnosed with PD as grade 2 or 3. Each subject was asked to utter the vowel /a/ at the same normal loudness and their most natural way (modal phonation). Each sample of the database comprises the recording of the vowel /a/ from the Spanish vowel set ([a] from the International Phonetic Alphabet) uttered in a sustained way, each utterance lasting approximately two seconds, separated by silences from repeated utterances. The voice records were taken by a hand recorder at 16 KHz and 16 bits, in the neurologists’ office. They were segmented automatically by an energy-based method and the central part of the phonation selected, avoiding the initial and final transients.

B. Handwriting database

A database comprising of seven individuals with Parkinson’s disease and seven age-matched controls were used in this study. All patients were tested on medication. All participants had written six loops which progressed to the right (i.e., cursive connected ‘lilill’) with an electronic pen. The participants were instructed to match the size of the cursive ‘l’ which were displayed before each trial. After each trial the performance of the participant was displayed between two lines which were 25 mm (i.e., 2.5 cm) apart, so the participant could see if s/he matched the required size. This condition was one of several size and speed conditions which were part of a much larger study. Each participant did repeat the writing task 8 times in which they tried to match the 25 mm size requirement.

The data were automatically segmented with a custom made segmentation procedure which searched for the first zero crossing in vertical velocity after the first full loop and thereafter searched for the first zero crossing in vertical velocity after the fourth full loop, i.e., each segment consisted of three loops per repetition (i.e., the connected second, third, and fourth loops of each trial).
C. Signal processing.

To process the voice production samples, first the central part of the signal was isolated to avoid the transients, where the speed is higher in Parkinson’s patients [20]. Second, the first and second formants were identified, after which the velocity was calculated as explained in section II.

Once the velocity signal of the voice and the handwriting samples was obtained, the procedure to estimate the parameter was the same, i.e., $v_{\text{voice}}(t)$ and $v_{\text{handw}}(t)$. The position signal data were fed into the sigma-lognormal estimator, where the signals were filtered with a cutoff frequency of 7 Hz. The lognormal parameters were calculated using the low-pass filtered signal as explained in section III. A new analytical signal was reconstructed from the estimated parameters as it is shown in Fig. 1 and Fig. 2.

V. RESULTS

A. Qualitative comparison of Parkinson and normal voice and handwriting profiles

As a first step, we compared the low-pass filtered velocity profile (original) with the analytical one. As can be observed in Fig. 1, the analytical and the original seem similar. The SNR is 21.7 dB in PD and 25.38 dB in control, being SNR the error between the original and its reconstructed signal gives the reconstruction quality in the sigma-lognormal domain [23].

If we compare the handwriting signal (Fig.1) with the voice signal (Fig.2), we can observe that they are similar but the time between peaks is longer in handwriting, i.e., 0.35 s in handwriting and 0.08 s in voice production. This could be due to try keeping the tongue position at a constant target to utter the vowel /a/. Instead, in handwriting, the movement is fast and longer, where simple movements are joined. The overlapping of two consecutive lognormals depends on the initial time of each lognormal and on the stroke logresponse time, therefore, the longer is the movement also the longer will be the stroke logresponse time, and there will be more overlapping between lognormals, as it is explained in[24].

Comparing a control (Fig. 1 and Fig. 2 upper) with a PD (Fig. 1 and Fig. 2 down), one can observe how the velocity signal of the PD has more and shorter peaks and the variability of speed is also noticeable. When inspecting the voice signal it can be seen also that the speed is higher for the PD. This higher speed could be due to the disability of PDs to keep the tongue in the same position.

B. Quantitative comparison of voice and handwriting averaged stroke response time $\Delta \sigma$.

The average of the values obtained for $\Delta \sigma$ for the voice data and handwriting data for healthy controls and PD patients across the two databases are given in Table I, Table II, Figure 3 and figure 4. It can be clearly observed that the mean of the parameter $\Delta \sigma$ is lower for handwriting and voice production of PDs.

To evaluate whether a statistical difference exists between the $\Delta \sigma$ value of controls and PDs an ANOVA (Analysis of Variance) was performed (using the statistical toolbox of Matlab). The two groups were considered different when the residual p-value is close to 0 and statistically similar if the p-value is greater than 0.05 [25]. It was shown that the groups differed on their $\Delta \sigma$ values in both handwriting and voice (p-values were lower than 0.05). However, voice showed a larger difference than handwriting between the two groups, suggesting that voice production is more discriminative than handwriting. However, to verify this latter suggestion, this study should be repeated using voice and handwriting samples of the same participants.
The results seem to validate that it is possible to use a common parameter to assess voice production and handwriting. Furthermore, the current study shows that Parkinson’s disease affects the velocity profile of both handwriting and voice production. Comparing these handwriting and voice productions, it seems that voice production is better to distinguish PDs from controls. This pattern of findings, could be due smaller movements and less inertia when using the tongue as compared to hand movements made when writing. These characteristics could affect the width of the lognormal and the separation between them.

VI. CONCLUSION

The results seem to validate that it is possible to use a common parameter to assess voice production and handwriting. Furthermore, the current study shows that Parkinson’s disease affects the velocity profile of both handwriting and voice production.

Comparing these handwriting and voice productions, it seems that voice production is better to distinguish PDs from controls. This pattern of findings, could be due smaller movements and less inertia when using the tongue as compared to hand movements made when writing. These characteristics could affect the width of the lognormal and the separation between them.

ACKNOWLEDGMENT

This study was funded by the Spanish government’s MIMECO TEC2016-77791-C4-1-R and TEC2016-77791-C4-4-R research project and European Union FEDER program/funds and the NSERC-Canada Grant RGPIN-2015-06409 to R. Plamondon.

REFERENCES


Exploiting the Lognormality Principle: Three Ongoing Projects.

Alexandra Nadeau1, Karina Lebel2, Joan,Carbo3, Arnaud Boré1, Ovidiu Lungu, Julien Doyon1, Hung Nguyen4, Christian Duval4, Patrick Boissy2, Nicole Vincent5, Koi Zapula6 and Réjean Plamondon3

1-Centre de recherche de l’Institut universitaire de gériatrie de Montréal, Montréal, Canada
2-Faculty of Medicine and Health Sciences, Orthopedic Service, Dept of Surgery, U. de Sherbrooke, Sherbrooke, Canada.
3-Laboratoire Scribens, Département de Génie Électrique, École Polytechnique de Montréal, Montréal Canada
4-Département des Sciences de l’activité Physique, Université du Québec à Montréal, Montréal, Canada
5-Laboratoire LIPADE - Université Paris Descartes, Paris, France
6-Laboratoire IRIS, Physiopathologie de la Vision et Motricité Binoculaire, CNRS, Université Paris Descartes, France.

Email addresses: alexandra.nadeau.0501@gmail.com; Karina.Lebel@USherbrooke.ca; johan.carbo@gmail.com; arnaud.bore@gmail.com; o.lungu@umontreal.ca; julien.doyon@mcgill.ca; hpnguyen@utexas.edu; duval.christian@uqam.ca; Patrick.Boissy@USherbrooke.ca; Nicole.Vincent@mi.parisdescartes.fr; zoi.kapoula@gmail.com; rejean.plamondon@polyml.ca.

Abstract—This paper presents an overview of three ongoing projects exploring new potential applications of the Kinematic Theory of rapid human movements. The first one deals with the development of a metric to objectively evaluate the efficacy of aerobic exercise through stationary cycling in improving upper limb functions in patients suffering from Parkinson Disease (PD). The second investigates the possibility of defining objective cranio-caudal kinematic signature metrics for Healthy and Parkinson subjects. The third explores the possibility of modelling and characterizing sequences of eye saccades to evaluate Alzheimer’s disease. The first study relies on standard protocols based on digitizer data, while the second relies on orientation data provided by an inertial measurement system. The third one analyzes data collected using an eye movement tracker. Preliminary results are reported and upcoming follow up studies are briefly described.

Keywords— Kinematic Theory, Delta-Lognormal model, Sigma-Lognormal model, Parkinson disease, Alzheimer disease.

I. INTRODUCTION

Over the years, using simple behavioural tasks and analyzing the data within the framework of the Kinematic Theory has been proven reliable in describing the velocity profile of simple human movements exploiting lognormal functions to describe the status of the synergistic action of an agonist and antagonist neuromuscular networks in numerous pointing tasks [1,2,3,4,5]. Doing so, the parameters extracted in the signal reconstruction provided a global evaluation of a subject fine motricity in terms of a central representation of his action plans and the timing properties of the peripheral processes reacting to it. Different types of tasks have been proposed within the kinematic model framework; they included for example simple or choice reaction time tasks in response to visual or auditory stimuli, fast pen stroke sequence tasks, such as drawing triangular trajectories or executing one’s signature repeatedly [6]. Recently, such tasks, have proven to be successful in discriminating people at risk for strokes [7], in estimating the fine motor control of kindergarten children [8], in studying and modelling motor control variability [9] in studying handwriting [10] as well as speech [11] in Parkinson’s disease.

This paper presents an overview of three ongoing projects exploring new potential applications of the Kinematic Theory of rapid human movement. The first one deals with the development of a metric to objectively evaluate the efficiency of physical exercise using stationary cycling in improving upper limb functions in patients suffering from Parkinson Disease (PD). The second investigates the possibility of defining objective cranio-caudal kinematic signature metrics to study PD axial control when turning. The third explores the possibility of modelling and characterizing eye saccades to evaluate Alzheimer’s disease. The first study relies on standard protocols based on digitizer data while the second makes use of orientation data obtained with an inertial measurement system. The third one analyzes data collected using an eye tracker. For each project, a short description of the experimental protocol and preliminary results are reported. Upcoming follow up studies are briefly described.

II. CYCLING TRAINING AND PARKINSON DISEASE

A few studies have suggested that aerobic exercise, like stationary cycling, could be an effective mean by which to improve specific upper limb functions in Parkinson’s disease.
patients [12]. Yet this assumption has not been tested directly. In this project, we thus investigated whether the Kinematic Theory could help in assessing the efficiency of exercise in improving upper limb functioning in PD [13].

Eighteen PD patients and twenty healthy adults (HA), aged between 40 and 80 years old, took part in the study. They were right-handed, sedentary, neurologically intact. HA participants were matched with PD patients at the group level with respect to sex distribution, age, number of years of education as well as cognitive and fitness levels.

The duration and frequency of aerobic exercise training (AET) sessions were set at 12 weeks, 3 times per week. Time of each exercising began at 20 minutes and 60% of intensity, and was then increased by steps of 5 minutes and 5% of intensity every week, until participants reached 40 minutes of training at 80% intensity. The kinematic properties of the upper limb movement were assessed with a target directed fast simple reaction time task using the delta-lognormal model. This task employs a digitizer and an electronic display to present visual stimuli. The tablet has a dot in the centre (starting position) and target zones on either side, participant being asked to draw straight lines on the graphic tablet, using a non-inking pen. The task consist in executing simple arm movements in response to a visual stimulus, each participant using his preferred hand in his preferred direction. Figure 1 depicts the test sheet used. This guide sheet was placed under the transparent plastic fold on the tablet to reduce friction and indicate the starting point and the target zone to subjects.

![Figure 1](image1.png)

Participants’ trials were used to extract several kinematic parameters of the movement based on the Kinematic Theory and its delta-lognormal model. A repeated model ANOVA (with pre- and post-AET assessments as repeated measures) was used to test the effect of cycling on primary and secondary outcomes in PD participants. In addition, measures at baseline that showed significant inter-group differences (i.e. depression as measured with BDI) as well as age were used as covariates in a mixed-model ANCOVA; the latter allowing to test for group differences across assessments (Group*assessment interaction). We also tested for the effect of training within each group and for group differences, separately at the baseline and after AET, for all kinematic movement parameters. In order to account for the effect of multiple comparisons, statistical significance was adjusted using the Bonferroni method. Paired t-tests were used to evaluate AET-related changes in UPDRS sub-scores in PD participants. All results were expressed as means ± standard deviations for descriptive statistics. Analyses were conducted using SPSS 21.0 (IBM, Armonk, NY; IBM Corp.). The level of statistical significance for all tests was set at p<0.05.

One main advantage of the Delta-Lognormal approach is that the reconstruction of the velocity profile gives access to the agonist and antagonist component of the synergy involved in the stroke generation. Figure 2 illustrates typical mean changes observed on PD subjects after exercise.

![Figure 2](image2.png)
state of the global neurophysiological mechanisms involved in the motor control of the upper limb.

III TURN CRANIO-CAUDAL SIGNATURE IN PARKINSON DISEASE.

Falls are common events for older adults (OA) with one in three OA experiencing a fall every year [14]. Amongst patients living with Parkinson’s disease (PD), the rate of occurrence of falls almost doubles [15], with a majority of these falls occurring while turning [16]. Turning is recognized as a demanding functional task requiring planning, coordination, and postural stability to process and execute the central nervous system instructions in a safe and efficient manner [17, 18, 19]. Healthy individuals typically exhibit a specific cranio-caudal sequence during a turn [20, 21]: the head initiates the motion, followed by the trunk, and then the pelvis to complete the body’s reorientation process into the new desired direction. This sequence can therefore be interpreted as a pointing task in the orientation domain. In this project, we investigate if the Kinematic Theory could be of any use to model and define cranio-caudal kinematic signatures. [22]

To explore this possibility, 31 participants (16 asymptomatic older adults-OA with 50% male, 50% female; mean age = 69.1 ± 7.5 years old; 15 older adults diagnosed with early stages of Parkinson’s disease (PD), 60% male, 40% female; mean age = 65.8 ± 6.4 years old) were recruited from the community in collaboration with Quebec Parkinson Network. Participants performed repeated 10 meters Timed-Up and Go trials. Briefly, each trial was initiated with the participant sitting on a chair. Upon the signal, the participant stood up, walked away for 10 meters, turned around and came back to his initial seated position. While performing these trials, participants were equipped with the IGS-180 suit (Synertial Ltd, UK), an inertial measurement system enabling full-body kinematics assessment. Specifically, the IGS-180 contains 17 inertial modules which are placed strategically on the participants’ segments. Each module is comprised of a set of inertial sensors (3D accelerometer, 3D gyroscope, 3D magnetometer) as well as a fusion algorithm which estimates, from the raw data obtained with the inertial sensors, the 3D orientation of the module in a global reference frame defined by the system. Hence, the modules positioned on the head and the trunk provide measures of the orientation of the head and the trunk, expressed in that same reference frame. Mathematically, it is then possible to compute the relative orientation of the head to the trunk, and then to derive the associated relative angular velocity profile.

Specifically, each trial was manually reviewed and segmented using the avatar in IGS-Bio, the application available with the IGS-180. The turn portion of the trials was then further investigated. Using Matlab (v2015a, Mathworks, USA), relative orientation of the head to the trunk was mathematically computed and expressed in anatomical planes of reference for each trial. Once filtered out (4th order Butterworth, cut-off frequency = 1.5 Hz), the relative angular profile was transferred back into its quaternion form to enable the computation of the associated relative angular velocity profile.

We then used the sigma-lognormal model to characterize the turn cranio-caudal signature. Indeed, the two segments involved (head and trunk) can be seen as two neuromuscular systems (NMS), each one having its own lognormal impulse response. The head-trunk system, which output can be seen as the vectorial summation of both basic systems outputs, can thus be characterized by its cranio-caudal velocity profile. The latter can then be decomposed into two phases corresponding to the moment the head initiates the turn, moving away from the trunk (phase 1) and the moment the trunk engages into the turn, closing the gap with the head (phase 2). We can therefore mathematically describe this complex system as the subtraction of the two illustrated velocity profiles (See Figure 3).

A first estimate of the lognormal parameters was calculated using specific points of the velocity profile following equations reported in [23, 24] to define the cranio-caudal signature of a subject [25] and analyzed statistically. Figure 4 illustrates typical turn cranio-caudal signatures captured for both a healthy elderly and a patient living with early Parkinson’s disease.

Among other things, it was observed that both phase 1 and phase 2 NMS commands are significantly reduced in patients with early PD compared to asymptomatic elderly (D₁ Healthy: 23.7 [17.1, 28.3], D₁ PD: 16.2 [11.9, 21.5], p=0.041; D₁ Healthy: 31.0 [22.3, 36.4], D₁ PD: 13.1 [11.4, 15.5], p <0.001). Sensitivity to change of the proposed metrics was also studied, comparing the patients’ metrics assessed ON versus OFF medication. Again, both NMS commands have shown a statistically significant improvement with medication (D₁ PDOn: 18.5 [11.0, 28.7], p=0.009; D₂ PDOff: 16.2 [11.9, 21.5], D₂ PDOn: 20.2 [11.9, 26.3], p=0.033; D₁ PD: 6.2 [4.6, 7.8], D₂ PD: 7.3 [5.8, 8.8], p=0.001).
The turn cranio-caudal kinematic signature approach was then confronted with traditional metrics used to describe the quality of a turn when assessed using inertial systems [26, 27, 28]. For the same participants and trials, comparing healthy with patients in early stages of Parkinson’s disease in their OFF state (i.e. no medication) have demonstrated that the number of steps required to complete the turn significantly increased (Healthy: 4.0 [3.5, 4.0], PDOff: 4.5 [4.0 5.2]; p=0.014), while the turn mean and max velocities significantly decreased (Mean turn velocity—Healthy: 1.5 rad/s [1.5,1.7], PDOff: 1.1 rad/s [1.1, 1.2], p <0.001; Max turn velocity—Healthy: 3.9 rad/s [3.6, 4.1], PDOff: 2.9 rad/s [2.8, 3.3], p <0.001). However, traditional metrics did not capture a significant change in the number of steps required to complete the turn, the mean turn velocity nor the max turn velocity between medication states (Number of steps: p=0.462, Mean turn velocity: p=0.173, Max turn velocity: p=0.552).

IV. EYE SACCADES AND ALZHEIMER DISEASE.

Many studies have investigated and reported alterations to oculomotor function and viewing behaviour as a result of Alzheimer's disease (AD) [29]. In this project, we investigate if the Kinematic Theory could be of any use to model and study eye saccades. For many years, it has been known that the delta-lognormal model could be used to model single eye saccades. In this study, we want to first check if the model could be used to reconstruct log sequences of saccades as currently used in some Alzheimer disease tests, as illustrated in figure 5. These data were captured with an eye movement tracker (the 2008 version of the gaze-driven camera EyeSeeCam).

![Figure 5](image)

To do so the patients have to follow some protocol. The task is to follow with the eye, a moving target on a computer display. First the target appears in the center of the screen during a delay of D1 and disappears during a D2 laps of time and comes back on a peripheral point on the screen during a D3 period of time. The location can take four points, either on the left, on the right, on the top or the bottom of the screen as illustrated on figure 6.

![Figure 6](image)

The hypothesis is that the structure of the oculomotor trajectories is linked to the presence or not of some neurodegenerative dementia and even the type of dementia such as Alzheimer or dementia with Lewy bodies. Indeed, studies show the reaction time as well as the variability in the ocular saccades are varying with respect to pathology or non-pathology and with the type of pathology [30].

In this study, the data are captured from 12 patients that have been labeled by the experts in three groups according to clinical information (severe, mild, sane) as well as some quantitative information as age and MMSE (a neuro test). The file to be analyzed contains the sequences of geometrical coordinate of the eye trajectory. Some noise has to be removed to eliminate some aberrant values and this must be done in a way as coherent as possible with the expert interpretation.

In this study, the delta-lognormal model, did not fulfill our expectations in order to have a well reconstructed velocity profile. The principal problem which caused the non-success of the delta-lognormal model was due to the difficulty of isolating individual eye movements, as we tried at the beginning of this project. We thought that we would deal with straight and simple movements, which the delta-lognormal could be capable of fitting and representing. After looking the real complexity of the movement, we decided to go ahead with the analysis but now, using the sigma-lognormal model. Using the Script Studio executable software, provided by the Scribens lab, we proceed to do the analysis of the eye movements files.

The experimental task was evaluated using the axiom that the error between the reconstructed specimen and the original one should be smallest as possible, and also the extracted solution should be somehow consistent. The error is easily evaluated using the SNR. The larger the SNR, the better is the reconstruction. The conclusion we reached is that the sigma-lognormal model comply with our expectations and it is able to reconstruct the original signal with a SNR over 25 dB, as opposed to delta-lognormal. Another important thing to take into account is the number of lognormals extracted using sigma-lognormal. Once we have a good SNR, we have to focus on the number of lognormals, since a smaller number is generally better. This is a topic we have to work on, and see if we can get some information about the saccade analyzed knowing the number of lognormals extracted. Figure 7 shows a typical result.

![Figure 7](image)

Moreover, we tried our executable with greater and smaller signals than the one presented above. The aim of analyzing smaller signals is to study the behaviour of the extractor with
only a unique saccade being this the predominant one of the signals.

In this experiment, we found that the extractor needs more information to reproduce the saccade. For example, for a typical signal, which gives us 27.8823 dB of SNR with a 0.5 seconds duration, the same signal, now of 0.17 seconds of duration and with only the predominant lognormal lead to a 17 dB reconstruction with 2 lognormals. On the other hand, if we increase the window and we analyze the same signal but now with 1.5 seconds we get 34.54 dB of SNR and 39 lognormals. We introduce the maximum of signals that the executable able us (100 seconds from a total signal of 217 seconds). It took 4876 seconds to finalize the parameter extraction, and the results were so good. We reach a 28.64 dB of SNR with 1831 lognormals. This confirms that we can use the sigma-lognormal model to study the effect of Alzheimer’s disease on eye saccades, as illustrated in Figure 8.

Figure 8

V. DISCUSSION

The first study has shown so far that aerobic exercise training has a significant positive impact on the capacity to draw lines in a more efficiency way, in PD patients, indicating an improvement in the upper limb motor function. Specifically, after training there was a significant decrease of the antagonist response of the movement (D2), an amelioration that can be interpreted as an improvement in the control of the motor command in PD patients. The fact that we observed improvements in kinematic parameters, but not in the UPDRS scores, after training suggests that our task using the delta-lognormal model may be a more sensitive test to assess changes in motor function (and indirectly, motor symptoms) during treatment in PD.

In the second study, we have first shown the ability of the proposed methodology to capture the turn cranio-caudal signature in asymptomatic elderly. We then confirmed the capability of the same approach to be used in patients living with early Parkinson’s disease. This study allowed us to verify that the general turn cranio-caudal signature patterns are similar between both populations, but that the specific characteristics of the signature are different. The metrics derived from the signature using the sigma-lognormal model (i.e. amplitudes of the neuromuscular system commands for phases 1 and 2 of the turn) have shown a good potential to discriminate between healthy individuals and PD patients. The same metrics were also able to discriminate between medication states in early PD, a difference that could not be captured with traditional metrics. Although exploratory, this study has therefore shown that the use of the sigma-lognormal model to characterize the turn cranio-caudal signature has the potential to provide us with a set of discriminative metrics offering a good sensitivity to change.

In the third study, our preliminary results have shown that the delta-lognormal model is not appropriate to study long eye saccades sequences, since in the majority of the cases the movements are not straight saccades as we thought of in the beginning. It was thus decided to exploit the sigma-lognormal model and to continue this analysis.

VI. CONCLUSION

Based on the Kinematic Theory of Rapid Human Movements, the fundamental hypothesis of these studies assumes that the speed profile of a movement contains significative enough information to define the global neuromotricity of individuals. This profile is dependent on the equilibrium of several physiologic systems (homeostasy). Any disruption of the homeostatic state might change the characteristic of the profile. Detecting those disruptions in the movement profiles, would provide indices reflecting the homeostatic state of an individual. Similarly, as we can think of sismographic indices to allow ourself to get the intensity of an earthqwake, the kinematic parameters of this theory would be the mirror or the thermometer of the global health of an individual, as postulated by the lognormality principle.

In the forthcoming months, for the first study, we will correlate our findings on aerobic exercise training with various measurements and experimental conditions. For the second study, our goal is to reanalyze the data using the Scriptstudio software to get the full picture of the cranio-caudal signature. Regarding the eye movement study, we are now ready to proceed with the full analysis of our database. Updated results will be reported at the conference.

REFERENCES


Workshop: CALL in the Context of Teaching and Learning Chinese as a Foreign Language

Fei Song
Application of Virtual Reality Classroom with Panoramic Cameras into Chinese Language Teaching

Hongying Gao
Create Attractive Teaching by Using Domestic Animation

Hong Luo
Applying Internet Thinking in Teaching Chinese as a Foreign Language
Abstract—Chinese is in a rapid process of globalization, but in the same time, problems also arise. For example, teaching institutions and teachers are distributed unbalanced, and Chinese learners are numerous but dispersed. In this context, it is not realistic, only relying on the onsite teaching in classroom, to meet the needs of Chinese learning in the world. Thus, online teaching is becoming an inseparable part in Chinese learning. However, it also has problems, such as the lack of classroom atmosphere and language communication environment, weak classroom management and so on, which seriously restrict its development. Accordingly, applying virtual reality (VR) technology into online teaching might be a feasible method. But virtual scene was hard to make and VR hardware was expensive in the past, thus keeping it at a semi dormant state for a long time. Until recently panoramic camera is commercialized, it is possible for its large-scale use. Thus, in this study, panoramic camera was used to build a VR classroom for Chinese teaching, to create an immersive learning environment for online Chinese learners. Besides, the main application scenarios, software and hardware, building process, participants’ feedback, and development direction were discussed, in an attempt to make improvement for its application, as well as to offer a reference for the further research.

Keywords—Chinese Globalization; Panoramic Camera; Virtual Reality; Chinese Language Teaching

I. PRINCIPAL PROBLEMS DURING CHINESE GLOBALIZATION

With the rapid development of China's economy in recent years, Chinese is in the process of rapid globalization. The phenomenon of “Chinese fever” appears in many countries and regions and Chinese learners also rises rapidly. Meanwhile, problems about the International Chinese Language Teaching arise, too. For example, Chinese globalization is rapid but teaching institutions and teachers are distributed unbalanced; learners are numerous but scattered; online learning is in great demands but its effect is poor.

A. Contradiction between the rapid Chinese globalization and the unbalanced distribution of teaching institutions and teachers

With the rapid Chinese globalization, teaching institutions and teachers is in extremely urgent demand, but distributions of teaching institutions and teachers are unbalanced worldwide.

Taking Confucius Institute/Classroom as an example, the number of Confucius Institute/Classroom in Europe and the Americas far exceeds Africa and Asia. By October 23rd, 2017, on average, there were 7.32 Confucius Institutes and 25.23 Confucius Classrooms in the Americas, but with only 1.32 Confucius Institutes and 0.71 Confucius Classrooms in African countries, showing obvious difference.

I. AVERAGE WORKLOADS OF INTERNATIONAL CHINESE LANGUAGE TEACHERS IN CONFUCIUS INSTITUTE ACROSS THE WORLD

The data comes from the website of Confucius Institute Headquarters (Hanban): http://www.hanban.edu.cn/.
To some extent, whether the distribution of teachers is reasonable or not can be measured in terms of teachers’ per capita workload. If the distribution of teachers were balanced, the per capita work load of teachers in each continent should be the same or similar, otherwise unbalanced. The data above shows that the per capita workload of teachers working in Confucius Institute/Classroom has great difference. The average workload of 3 Chinese teachers at Confucius Institutes in Africa (27.84 persons per standard academic year) is five times more than that in Oceania (5.39 persons per standard academic year).

Thus, it is fully reflected the serious imbalance in the distribution of Teaching institutions and teachers represented by the Confucius Institute.

B. The contradiction between numerous learners and scattering distribution

It is believed that the number of Chinese test candidates presented in the Confucius Institute Headquarters (Hanban) Annual Report can approximately reflect the number of Chinese language learners, because those in a relatively formal Chinese language course must face a certain form of course examination.

II. CHINESE TEST CANDIDATES IN THE CONFUCIUS INSTITUTE HEADQUARTERS (HANBAN) ANNUAL REPORT OVER THE YEARS

As indicated in the above chart4, from 2005 to 2015, the number of Chinese test candidates has increased rapidly, with over 160 times. At present, there are more than 6 million candidates according to the statistics. Nevertheless, those learners are scattered around the world, especially in those countries with vast territory but small population density, for which such phenomenon is rather obvious.

The large number of learners reflects the strong demand for Chinese language learning, but the scattering geographical distribution makes it difficult to meet such strong learning needs only by traditional onsite teaching, especially in a vast territory with a sparse population, which is even harder. And besides, the cost will be higher.

C. The contradiction between large demand of online learning and low efficiency

Under such conditions mentioned above, it is impossible to meet the ever-increasing demand for learning only relying on onsite teaching. Other approaches are urgently needed and online learning, which is deemed as the most reasonable, is one solution. On the Internet, wherever they are, learners can share resources equally from same teachers; and no matter how many learners there are, they can access to online teaching by widening

---


3 Wu Yinghui (2011) proposed an indicator named “person per standard academic year” which is used for measuring Chinese language teaching workload. The calculation formula of this indicator is: person per standard academic year = (number of learners × days of study) ÷ 160 days, which counts based on 5 days a week, 20 days a month, and 4 months per semester.

network bandwidth and improving servers to better cope with concurrent visits.

However, the deadly problem online learning faced with is dull atmosphere and lack of verbal communication in the classroom. Classroom teaching can be regarded as a miniature society which requires teachers and students to form a stable team. Their roles and responsibilities of these two types are clear. Using language, facial expressions, eye contacts, gestures, and movements, teachers interact with students instantly, deliver them linguistic knowledge, guide them in language exercises, and convey backgrounds beyond the textbooks with their own knowledge and charisma. In the same way, students give their immediate feedback to teachers, allowing teachers to master the content and rhythm of classroom teaching. However, current online learning cannot truly reproduce the same atmosphere and scene as classroom teaching. On the one hand, because expressions, eye contacts, gestures and movements of teachers to convey information are limited on a digital screen, causing severe information loss; on the other hand, students are not in the onsite classroom; namely, they are separated from the real-time teaching scene and lack of immersive learning environment. More importantly, students need to force themselves to focus on the teaching scene presented by a small digital screen for a long time, which in itself goes against the cognitive characteristics of people.

Restricted by the problems above, online teaching is quite different from onsite teaching, which also limits the application of online learning to a large extent. The key is to overcome the existing problems in online learning; make it more effective; simulate the real classroom teaching scene to the maximum extent; and enable learners immersed in onsite teaching. Hence, a VR classroom based on “Panoramic Camera + VR Technology” could be a way.

II. THE KEY SOLUTION: BUILDING AN ONLINE VR CLASSROOM

A. Definition of VR technology

Virtual Reality (VR) technology integrates various scientific and technological techniques such as computer graphics technology, computer simulation technology, sensor technology and display technology in different application scenarios. It can create a virtual information environment in multi-information space, in which users immerse themselves, and in advanced application, it can be inspiring and helpful for environment improvement. At present, VR technology has been applied into larger areas including Television Conferencing, Internet technology and Distributed Computing Technology, instead of being restricted into computer imaging, and it exhibits a tendency toward distributed virtual reality. Collaborative VR is a hot topic for VR technology research and application. New technology issues such as human factors, network, and database technology also emerge. In terms of human factors, it is necessary to consider how participants interact with each other in a shared space, and how virtual object behave under the combined action of many participants in the virtual space.

B. Resolutions to the existing problems with VR technology

Building an online VR classroom is a major solution to the existing problem in the rapid Chinese globalization there three reasons: Online classroom can avoid the space-time distance restriction; Teaching resources and teachers can be distribute more balanced over the network space by speeding up network and improving servers to receive and send heuristically; And gaps between online and onsite teaching scene can be shortened by VR technology. The first two have been solved or are solving by online learning; while, the last may be solved by VR technology.

Effects of online learning are limited, because information on electronic screens is consumed, learners cannot immerse themselves in online teaching scenes and they can easily be distracted. However, VR technology can avoid all of those shortcomings, for it can truly represent teachers’ movements and poses and other visual media for information delivery, instead of being restricted by electronic screen. And it can also reduce the information consumption. Meanwhile, the online teaching scene is completely covered by reality scene, which solves the issue of learner’s distraction.

C. Problems of Virtual Reality in current Chinese teaching

Nowadays, VR technology has been researched and applied in Chinese teaching, and most studies are about the application prospect including VR, VRML and VR game
(represented by Second life in a broad sense) applications in Chinese teaching. In addition, VRML studies began to involve how to develop virtual scenes, and papers discussing applications of the game Second life into Chinese teaching can be regarded as one example.

The present study has a clear problem: the prospect about applying visual reality into Chinese teaching has been put forward over the past decades, but no substantial achievements have been made till now. Strictly speaking, the game Second life does not belong to VR technology. And no substantial progress have yet been made in respect of VR application in Chinese teaching for a long time because of its high requirements of skill and high price of hardware, which made the former researchers and teachers cannot afford to apply them. Furthermore, the previous researches limited its application in virtual scene on computer, thus raising standards of scene design. But they ignored the fact that online visual class could be applied into the classroom teaching, which might have a better effect.

D. Application of VR technology in Chinese teaching with remote students to a maximum degree.

1) Definition of panoramic camera

Panoramic camera, commonly with two or more lenses, is used to take different pictures in the same space at the same time; then, these pictures will be send back to camera or computer’s image process unit for further processing by deformation and splicing; and finally, a complete space panorama is formed. Instead of a plane by common camera, pictures taken by panoramic camera is like a sphere, and viewers are in the center of the sphere for appreciation.

2) Breakthrough of panoramic camera for the development of VR teachers

A kind of technology must first be featured with easy operation and application for teachers, and then it can be practical and popularized widely, so does VR technology. In spite of years’ development, it cannot be applied into Chinese teaching due to difficulties in scene building. However, it changed completely with commercialization of panoramic camera.

Teaching scenes in onsite classroom are filmed and recorded by panoramic camera, and then delivered to the VR terminal via internet. Next, those videos are played and transformed into virtual scenes by remote students wearing VR glasses so that they enjoy virtual teaching. In this case, this panoramic camera is like eyes of those remote students, without any change of eyes in space coordinate, they can see the whole classroom just by moving their heads, ensuring fixed hardware in the classroom, stable environment, as well as free and immersive feeling for remote students to a maximum degree.

More importantly, panoramic camera is installed and used in the same way as a common camera. In this condition, most teachers without commanding special techniques can easily build virtual scene, which thus breaks the limitation of virtual reality on teacher resource development, and become the tipping point for VR application in Chinese teaching. Thereafter, VR technology could be applied in Chinese teaching classroom genuinely.

III. BUILDING OF VR CLASSROOMS RELYING ON THE PANORAMIC CAMERA

In this study, a VR classroom relying on the panoramic camera is built to investigate the practical application effects of Chinese online VR classrooms, and then corresponding teaching scenarios, software and hardware, specific building process and participants’ feedback are concluded.

A. Applicable Online VR Classroom Teaching

At the initial stage of application, the following three requirements had better be met for VR teaching:

First, time difference, between the learning place and the onsite classroom, should be as small as possible; or otherwise, learning effects might be not good enough due to poor state.

Second, network of the learning place should be fast and stable, to ensure the transmission of audio/video (AV) signals for live broadcast; or otherwise, it might cause network delay, and hence not facilitate classroom teaching and teacher-student interaction.

Third, VR glass should be affordable and accessible to VR teaching learners.

Thus, South Korean students are now chosen as the subjects of study, for they satisfy all the above requirements. South Korean is adjacent to China, so their time difference is small. Its IT technology develops fast and well, which provides technical support for the broadcast. Besides, it is also convenient for the students there to afford VR glasses owing to booming economy and commercial trades.

B. Hardware and Software for the VR Classroom

VR classroom is divided into two parts: onsite teaching and VR terminal.

Devices for onsite teaching mainly include one panorama camera (about 2,000RMB), one tablet personal computer (tablet PC, about 1,500RMB), and wireless network, of which onsite teaching videos are recorded by the camera; then sent to students at remote terminals by way of wireless network; finally played on the tablet PC, and vice versa for students’ feedback.
Devices for VR terminal mainly include a pair of VR glasses (about 50 RMB), a smart phone, a laptop, and wireless network. VR glasses connect to the smart phone to play panoramic onsite teaching videos, which are transmitted by wireless network, thus creating a VR classroom and providing an immersive learning environment for students at remote terminals. Then, a laptop is used to record videos of students in the VR classroom and collect their voices. Finally, these feedback videos and audios are sent back to the onsite teaching class, which forms teacher-student interactions.

C. Building Process of VR classroom

The whole process is shown as below:
1) Fix a smart phone connecting a panoramic camera by the camera support, and then place it in the center on site, but not close to the stage too much;
2) Open the camera and APP Insta360 Nano in the phone and select a live-broadcasting platform (for example, Sina Weibo VR live broadcasting and the others), to record and broadcast real-time teaching scene panoramically;
3) Open WeChat in the laptop to connect students’ WeChat videos at remote terminals, and keep the microphone and speaker of the teacher at a mute state;
4) Send the link of live broadcasting to students. Then, after they open the link and select a “VR mode”, they can immerse themselves in the onsite teaching class by watching live broadcasting, with the phone placed before the VR glasses.

According to field tests, there is a delay of five or six minutes when the onsite teaching videos are sent to students at remote terminals through a live broadcasting platform. It cannot disturb the continuity of VR classroom, but to some degree, it can affect the real-time interactions between students and teachers. In the following tests, the teaching effects would become better if the problem of network delay could be solved.

D. Participants’ Feedback

After the class was over, the author interviewed parts of participants in VR classroom, with the conclusions as below:

From the perspective of teachers, they can create a VR classroom relying on a panoramic camera, with accessible hardware and low cost. After that, they nearly pay zero nearly zero marginal cost for additional students. On the other hand, related software such as WeChat is common and easy to use. The key difficulty is how to create VR classroom with the help of live-broadcasting platforms and then share it to students at remote terminals, so they need to spend some time on it. The more they use them, the more familiar they will be. Besides, in a VR classroom, they find nothing obviously different from the previous teaching in an ordinary classroom. Their teaching paces are not disturbed except interactions with these remote students due to internet delay.

Onsite students also say there is no remarkable difference in a VR classroom, excluding the center seat, which is left for a panoramic camera. However, it is in the similar height of students, so it cannot obscure visions of students in the back. In addition, they are not distracted by those devices if remote students do not speak.

As for those remote students, they enjoy more benefits. In the first, it cost them a little money to attend such a class, because it saves them time and fees for transportation, as well as the expenses of purchasing new hardware equipment. In the second, they have a strong feeling of involvement in the immersive teaching, as if sitting in the center of the classroom. They can concentrate on the class for a longer time than those ordinary online teaching videos. They can also interact with teachers and onsite students as if in a real classroom, for example, discussing with them in groups and feeling their eye contacts. But due to network delay, they could be drawn back to the reality occasionally. Moreover, after two classes, they may have eyestrain to a certain degree because of “a screen window” (like looking things through a screen window) when pixel of a phone is amplified by VR glasses. And they feel no obvious dizziness.
IV. PROSPECTS

It is believed that human’s cognitive world have been changed by internet, VR and Ai technologies. Just like internet in those years, VR is playing an increasingly significant role in people’s life today, and it might be a subversive revolution. With the rapid pace of Chinese globalization, VR Classroom relying a panoramic camera, which is deemed as an important breakthrough in applying VR into international Chinese language teaching, can solve such problems as unbalanced distribution of teachers, disperse Chinese language learners, insufficient scales for designing curriculums or establishing teaching schools. It also covers the shortage of traditional online teaching and breaks the limit of developing VR teaching resources.

At present, there are still some problems in building VR classroom, like the sharpness of panoramic cameras and VR glasses, stability of VR live-broadcasting platforms and network delay in particular. But most of them are caused by common low-cost devices and thus have a little influence on VR classroom with the aids of software. Similarly, it is an inevitable trend for such a teaching method to be applied in various online teaching in the next five or ten years.

With enough funds and mature technology at present, “a screen window” could be eliminated using professional panoramic cameras of commercial level, with the image quality high up to 8k, and all-in-one VR glasses of the same image quality. Meanwhile, Hi-Fi devices could be adopted for audios, so as to the output devices of students, and thus remote students could enjoy the same sound resources as onsite students do, or even judge where the speaker is in VR classroom according the sound direction. In addition, Holographic Imaging could replace laptops to display the onsite teaching. Specifically, holographic film + display or augmented reality devices like HoloLens, according to different budgets, could be utilized to create a virtual image of the remote student sitting in the center. Thus they can communicate with each other from an omni-bearing multiple angles. Luckily, all of these have been able to come true based on the current technologies.

At present stage, VR classroom relying on a panoramic camera opens the first door for VR technology to be applied in Chinese teaching, but it could not be the only available mode of application. In many stances, language teaching is featured with high stylization for classroom discourses, which means that AI application could also become mature one day as today’s VR technology does. Therefore, in the near future, a new mode of VR classroom based on “a panoramic camera + an AI language practicing model + 3D scanning and modeling + automatic speech recognition and synthesis + image recognition and synthesis” might also become popular. Then, a virtual teacher could practice these basic language skills automatically in a virtual classroom, by modeling with 3D scanning and relying on the intercom core as well as automatic speech recognition and synthesis technology.

REFERENCE

Using Chinese animation to create "attractive teaching"

Hongying GAO  
School of Humanities  
Tianjin University of Finance & Economics  
Tianjin, China  
Gaohongying2000@126.com  

Ran Li  
School of Humanities  
Tianjin University of Finance & Economics  
Tianjin, China  
qianchengzhuren@163.com

Abstract-- In the early days of Chinese animation making, the creators had a strong sense of national mission and paid great attention to elements of Chinese traditional cultural thoughts, which gradually took shape of the animation of Chinese school. Taking the animated film "Big Fish and the Sea" as an example, the paper points out that the film has done well in citing material from literary classics, borrowing allusions, customs and ethnic constructions to highlight ethnic characteristics. Chinese animations represent the national culture and have already been broadcasted abroad largely, and by using the effects of its audio-visual power, a kind of attractive teaching would be constructed, from which a unique national culture might be conveyed to the class and in which the aesthetic taste with a psychological healing effect will be produced.

Keywords-- Chinese animation, literary classics, tradition culture, environmental education, attractive teaching

I. Introduction

Starting the animation industry in the 1920s, China is one of the earliest countries in Asia to develop animation. We have focused on the traditional elements since its beginning. Many animated works demonstrate distinctive national characteristics and highest Chinese traditional culture in figures, themes and connotations, such as the tadpoles painted in ink in The Little Tadpoles Looking For Their Mother, Secrets of the Heavenly Book drawn from a famous Chinese classical fiction Subjugation of Demons, and the connotation of the importance of solidarity from Magic Deer and The Story of The Three Buddhist Monks derived from Dunhuang fresco King Deer and so on. These high-quality works once gained the acclaim of “China School”. Animated films are based on imagination. Attractive tales and rich audio-visual language have a key role to play in attracting audiences. The domestic animation called Big Fish & Begonia mixes many Chinese traditional elements together. The film has a fascinating origin grounded in Daoist classic, Zhuangzi, A Happy Excursion, “In the northern ocean there is a fish, called Kun, I do not know how many thousand li it is in size. This Kun changes into a bird, called Peng. Its back is so big that I do not know how many thousand li it is in breadth.” This film, based on the above-mentioned concepts, creates its own world view. We can understand the meaning of Chinese traditional culture deeply, obtain aesthetic standard and expand the cultural accumulation through this animation teaching.

II. The Chinese traditional elements initiated by Big Fish & Begonia

It has been 12 years before Big Fish & Begonia is on. With Chinese traditional culture, this film tells a story of a young girl called Chun pays a debt of gratitude to revive the human boy Kun’s soul and struggles against fate. This animated film shows a human world in the ocean. The people there, living in “God around the floor”, know the manner that governs the nature. As part of a coming-of-age ceremony, Chun is sent to human world. But on the day of return, she is trapped in a fishing net. After being saved by a human boy named Kun at the cost of his own life, Chun decides to repay her debt, and trades half of her own life for the boy’s soul. Under her childhood sweetheart, Qui’s help, and after tackling various difficulties and obstacles, she finally helps Kun return to Earth. However, this endeavor, against God’s rules, soon brings disasters to her own world, and Qui dies because of helping Chun go to human world.

On one hand, based on A Happy Excursion, Big Fish & Begonia reflects the views on freedom and life and death. Freedom is setting yourself free to achieve what you want to be. In the film, witch’s exchange and Chun’s grandpa’s words “death is the door to eternity” explain Zhuangzi’s view on life and death— “You are growing while you are dying.” On the other hand, Chinese people pay attention to “give” and “repay”, and advocate “the favor of a drop of water has been rewarded with the gratitude of a fountain of water.” Qu Li, as a part of The Book of Rites, writes, “courtesy requires a return of visits received.” [1] We can see Chinese traditional concept “karma” clearly from Chun. Therefore, the whole film seems to tell a story of Kun, but actually, it is about Chun’s growth of...
awakening self-awareness. Her self-awareness is becoming stronger and stronger from the moment that she goes to the witch to trade for the boy’s life to the opposite place she insists against her mother and the people in her world just to protect Kun. At last, in order to save the people, she makes herself and her grandpa’s body be a begonia to stop the flood.

III. The methods of audio-visualization, visualization and teaching in animation

First, as a vehicle of delivering ideas, domestic animated films do well in expressing meanings by different ways. Chun’s clothing is full of typical characteristics of the Republic of China. So, the Chinese-Red has been expressed visually through Chun’s red upper garment and her black middle-skirt and black shoes. At the same time, this kind of visual expression can ignite us to dig the deeper main ideas from surface. The domestic animated films greatly contribute to making connections between the animated films and connotations.

Moreover, audio-visual language can improve recognition efficiency and effect. Audio-visual language in animation is rich and informative that benefits concluding, refining and summarizing, and it can help audiences catch the main idea and implicit traditional culture. Thanks to its characteristics, animation teaching comes to not boring, and every picture can convey a different idea. In Big Fish & Begonia, Luo Zu makes Chun a coat with patterns of begonia, and pageboy style. These details indicate Chun is ingenious and gentle, while she wears silver pine-shaped earring on the right ear, which shows she is ready to free herself and challenge rules. This design fleshes out the portrayal of character and makes her changes in personality reasonable.

Third, Chinese names in the film have various meanings. Qiu is the leading actor, and his real identity is a God. This name also comes from Zhuangzi, A Happy Excursion, “in the south of Chu there is a tree whose spring and autumn are each of five hundred years’ duration. And in former days there was a large tree which had a spring and autumn each of eight thousand years.” Qiu boasts the ability to control west wind and make plants fruit and withered. And “Qui” (has the same pronunciation of “秋” in Chinese, which means Autumn) indicates a miserable and tragic end. Winter always comes after autumn, bringing tragedies. The explanations of names on right time increase the indicating information and make the class more entertaining. Qiu and Chun are childhood sweethearts. Qiu loves Chun, but he is different from Chun who lives in a happy family. Without parents, Qiu is adopted by a grandma who guards the door to Celestial Lake. In order to help Chun leave, he sacrifices himself. In the film, he is saved by witch at last, and becomes the next wizard to guard Rusheng Building. This matches the concept of Buddhist karma of doing good things and good things will come your way. Qiu is the embodiment of kindness, mercy and dedication. “Human beings differ from animals because human perform rites and benevolence which grow from human’s affection. Whatever the affection is, it comes down to the milk of human kindness.”[2] Big Fish & Begonia serves well as a typical teaching case, with certain background, to bring this film to students, tradition and national conditions, with which we can extend traditional culture further, and keep great insight to national culture. “In teaching activities, it is important to respect the dominate position of students, make the dominate position work, motivate students, and do everything for students.” [3] Following this principle, the learners can accept because of delights in class.

In the film, Qiu has white hair and kind of dark complexion. He wears a red short-sleeved and open front upper garment, a red pants, with white border both at waist and the bottom of pants legs. This style with bare feet shows he is free and he can do everything he want to do. Qiu is free and casual from the very beginning while Chun is on the way to achieve self-awareness. Actually, Qiu is the right role who matches Zhuangzhou's ideal person. After sending Chun to human world and leaving his loved, Qiu rides himself of all restrictions and he gets real freedom. As the essay above mentioned, Kun comes from A Happy Excursion, Zhuangzi, “In the northern ocean there is a fish, called Kun, I do not know how many thousand li it is in size. This Kun changes into a bird, called Peng. Its back is so big that I do not know how many thousand li it is in breadth. When it is moved, it flies, its wings obscuring the sky like clouds.” Kun owns two styles, one is a human with a red scar on forehead, wearing a white top, a black pants, a belt and a dragon of cloud carved silver collar. The other is the fish from Kun's soul—a little blue dolphin at early time, turns to be a white one later, a big red dolphin at last with grey belly, and huge wings. Kun's soul can be revived and return to his world only after he grows up. The setting of Kun indicates breaking rules. Under Chun's help, Kun not only breaks the door to Celestial Lake, but all "doors" which stops his return including the prejudice against human beings, no fish and so on. Kun's success is the achievement for Chun to pursue freedom. These three main characters play the vital role in inheriting traditional culture and they make it. Chinese traditional culture stands out in Big Fish & Begonia.

The main ideas of each animated films could be highlighted as climaxes, no matter where they are, the beginning, the middle parts
or the ending. These climaxes will be helpful to inspire students’ thirst for more information and make them be more active in the class. As a result, this kind of active participation will bring students’ advantages and their nimble wits out. If we could attract and keep students’, especially the naughty ones’ attention on Chinese cultural ideologies for long, finally, more interests will be fostered to these ideologies. To teach with the help of animated films, we should focus on the main ideas of them instead of their results at the box office. Under this precondition, the teaching method which is combined with the excerpts, bearing the major Chinese traditional cultural ideologies, will become the most effective way to realize Attraction Teaching.

References

Abstract—In the era of Web3.0, technology is no longer just a carrier or a platform, but also manifests itself as a kind of message, as Marshall McLuhan, the Canadian famous communication scholar, pointed out, "The medium is the massage." Web3.0 has brought changes in the way of information exchange, altering the way people exist, but also caused a change in the way of thinking, that is, the rise of Internet thinking. In teaching Chinese as a foreign language, teachers should not only take advantage of the convenience brought by the technological innovation in the Web3.0 era, but also actively explore the teaching of the second language through Internet thinking so as to obtain a multiplier effect. Open, interactive and sharing are the core values of Internet thinking. Comparing to the traditional way of thinking, Internet thinking also possesses the characteristics of being changeable and innovative from time to time, and puts the experience of participants in an important position, which means the teachers will be able to design more targeted and forward-looking teaching program, by sorting out and analyzing the learners’ performance and feedback timely. This thesis is aimed to explore the positive role of Internet thinking in teaching Chinese as a foreign language, but also tries to avoid the negative impact on Chinese teaching which is caused by the fragmentation and shallowness of Internet thinking. At the same time, some constructive suggestions are made for reference.

Keywords- the medium being the massage, internet thinking, teaching Chinese as a foreign language

I. Introduction

Teaching Chinese as a foreign language has entered the Web3.0 era of Internet technology. At the very beginning of internet appearing, Nicholas Negroponte, the well-known American futurist, made a sharp prediction that the calculation is no longer just about calculations, it will determine our survival. What the Internet brings to people is not only changes in the way information is communicated, but also changes the way we live and how we look at the world. The Internet has experienced the portal era of Web.1.0, and the Web2.0 search/social age and now enter the Web3.0 era. The typical characteristic of the Web 1.0 era is the display of information which is basically one-way flow and spread. The content is similar to the wonderful scenery on the top of the mountain from where the view is highly visible but the viewer’s participation is very low and in a passive state. At this stage of teaching Chinese as a foreign language, the teacher just place related knowledge information and methods on web pages, and the learners browse, which is based on traditional one-way teaching. The typical feature of the Web2.0 era is UGC which means the user generating content, which make the two-way flow of information and the real time interaction between users and producers become true. The educators and learners can communicate and interact by posting content and expressing opinions.

The Web3.0 is an intelligent era based on big data, cloud computing, and the Internet of Things. The typical feature in Web3.0 is the ability to achieve many-to-many communications and interactions. This interaction is not limited to people, but also includes people and machines, and the interactions and communications among multiple terminals as well. The Web3.0 is an era under the guidance of people-oriented Internet thinking in which individuals can reach the moment of networking as long as he/she wishes, and they can acquire the desired and real-time interactive state. This has brought great opportunities for teaching Chinese as a foreign language, and the educators and learners will be able to use Internet thinking to interfere with the daily teaching/learning work to adapt to this great internet age.

II. The Internet brings about a paradigm shift

In The Third Industrial Revolution, Jeremy Rifkin considered that if the first industrial revolution that took place in the 19th century brought about earth-shaking changes in the world, the second industrial revolution in the 20th century was creating a new world for human beings, then the beginning of the 21st century, that is, the third industrial revolution we are experiencing, has fundamentally changed our lives and work. The real-time communication and sharing of global information, the 3A revolution composed of production automation, office automation and home life automation, and the formation of various man-machine control systems have profoundly affected the way in which the human beings exist. According to Viktor Mayer-Schönberger, author of The Big Data Era, big data has triggered a major change in the times, and this change is first reflected in thinking. Schönberger believes that the core of the era of big data is to give up the concern about causality that people have been accustomed to for thousands of years and turn their attention to related relations. What people care about is ‘what is’ and no longer ‘why’. This shift has overturned the long-established thinking paradigm of mankind. A new thinking paradigm, that is, Internet thinking in response to the era of big data has emerged.

The English expression 'paradigm' is derived from Greek and has the meaning of 'co-displaying'. The scientific philosopher Thomas S. Kuhn believes in the book The Structure of Scientific Revolutions that every type of
conventional science is a paradigm. In Kuhn’s view, the paradigm generally contains: the first is ‘the way to see the problem’, which is expressed in the principles of thinking, methodology and world view, the second is the basic theory and the major scientific achievements that have been obtained, manifested in the symbolic system of logic and mathematical calculus, the third is instrumentations and mode of operation, the last is a set of typical examples of solutions based on scientific achievements. In conventional science, members of the scientific community perform scientific or academic research activities within the framework of the paradigm. However, when the conventional scientific research activities develop to a certain period under the guidance of the paradigm, they will face the emergence of ‘unexpected new phenomena’ which could not be explained or resolved under the guidance of the existing paradigm. Once this anomaly has accumulated over time, beyond the scope of the original paradigm framework, a scientific crisis has emerged. In order to solve this crisis, the scientific community needs to boldly examine and abandon the original paradigm of the conventional solution to the problem and propose a new paradigm framework to solve the crisis. Then the new paradigm is established, and the new conventional science or academic activity will be kept performing in the new paradigm until the next possible crisis comes.

Marshall McLuhan, the famous Canadian communication scholar, proposed the ‘media is the message’. He believes that the new technology is the driving force of the revolution and its influence in the human social production and the change in the way people exist are more than the specific information content that they load and deliver. The digital revolution we are experiencing is not only limited to the innovation of the physical world, but also ‘invented and applied with cloud computing, cloud applications, e-commerce, Internet of things, big data, smart cities, virtual technologies, and 3D printing technologies. It created a new revolution in production methods, industrial conditions, business models, and management models as well [1]. That the Internet’s thinking paradigm comes into being is just adapting to this change. It can be said that Internet thinking is the inevitable result of the scientific revolution and the inevitable outcome of the way of human survival and lifestyle change’ [2].

III. What the paradigm of internet thinking means.

As mentioned above, as being a new paradigm to break the traditional way of thinking, Internet thinking is based on the extensive and in-depth use of digital technology in the society. Compared to previous thinking, Internet thinking is mainly reflected in the following aspects.

Openness. Being open to all is an important feature of the Internet, and as the number of mobile terminals is rapidly increasing, the openness of the Internet is growing. From the report of Cisco: Interpretation of the Global Mobile Internet Development Trends Report 2014–2019, by the year 2019, mobile network users in the world will increase to 5.2 billion, an increase of approximately 21% compared to 2014. Meanwhile the mobile devices will exceed 11 billion, and the speed of connecting networks will reach 4Mbps, an increase of 140% over 2014. According to this growth rate and development trend, it is not difficult to imagine that in this ‘pervasive dissemination’ age, few people can escape from the Internet world and stay alone. The Internet is open to everyone and everyone can reach the Internet. This is an open world. No matter where and when you are, the earth has completely turned into a village. The openness of the physical world inevitably brings openness to thinking.

Decentralization. The Internet’s connection is a network structure, which means there is no central node, and it is not a hierarchical structure. Although different connection points have different weights, none of them are absolute centers and there is no absolute authority. In other words, the innate structure of Internet technology already presupposes that its inner spirit is to decentralize and is distributed and equal, which makes the traditional society more flat and more ecological from a multilevel vertical structure. Decentralized thinking on democracy and equality is the basic principle of the Internet.

Spirits of being free and sharing. Tangible atoms constitute the main unit of an industrial society, while the Internet world consists of intangible bits as the basic medium. This means that in the era of industrial civilization the scarcity economics is the main economics existing. In the era of Internet civilization, it is abundant economics. According to Moore’s Law and other theories, the three major elements that make up the Internet world—servers, storage, and bandwidth—will all point to unlimited free. In the Internet world, the monopoly of production and sales and the dissemination of knowledge information will no longer exist. In this purely net-like information society, information content determines value. The implication is that the creation of value is determined by the breadth and depth of the connection points. The wider and deeper an individual or organization is connected, the greater the value. Sharing then becomes an inevitable way of the human being’s life because sharing can gain more connections and will bring more values.

Mechanism of real time interaction. As described in the opening section of the article, the era of Internet intelligence based on big data, cloud computing, and the Internet of Things has gotten rid of the one-way flow of information and is no longer limited to the simple interaction of information, but has enabled many-to-many participation and Interactions, and the interactions have also transcended the interaction between people, that between people and machines. Far beyond that, the interaction between multiple terminals has also become a reality. However, the era of singing monologues is far away. In this age the public voices have become the norm, but it does not affect the function of big data forecasting, in which feedback and interaction will make the forecasting more accurate.

In the period of agricultural civilization, land was the
IV. There should be Internet thinking in teaching Chinese as a foreign language

Differences between environment and experience have shaped people's way to live. Since the current social development has entered the Internet age, no one is able to escape the era in which he is placed in, having Internet thinking is one of the ways of human survival in the era of big data. In particular, the post-2000 generation is completely known as the ‘digital natives.’ Since this generation is accompanied by the rise and rapid growth of the Internet, and they are proficient in various digital media such as computer equipment, game consoles, mobile terminals, and networks. In the other word, the digital media has far-reaching influence on their cognition, their behaviors and attitudes towards the world. ‘For digital natives, the network is life and the virtual reality is the world’ [4]. The teaching of Chinese as a foreign language should have Internet thinking and inspire learners’ Interests, improve teaching effectiveness, and expand the horizons of learners’ understanding of Chinese language and culture.

First of all, as far as possible, a reciprocal network structure needs to be constructed between educators and learners of Chinese as a foreign language. This decentralized, authoritarian, and flattened form can encourage each learner to effectively participate in teaching. On the other hand, the potential of learners can be exerted as much as possible to contribute their own strength in the entire structure, which not only highlights personal values, but also enhances the overall learning atmosphere of learners, inspiring educators’ and learners’ motivation to be engaged in the teaching process. Actually, this also fits with the teaching philosophy of ‘teaching and learning to promote each other’ proposed by Confucius more than 2,000 years ago. To a certain extent, the flipped classroom popular in this time is also a reflection of Internet thinking, in the process of which the students cultivate the ability to self-learning and self-dispose learning material outside the classroom, and actively discover and solve the problems. In the classroom, the teacher organizes students to conduct more discussions and explorations, and explores and solves difficult problems. Outside of the classroom, the learner’s motivation and imagination and activity are exerted. In the classroom, educators provide the humanized and differentiated guidance designed according to student characteristics. Educators and learners are in a net-like structure on an equal place. Compared to learners, educators only have slightly different weights in the process.

Secondly, educators of Chinese as a foreign language must strive to explore the relevance of learners and use an open perspective to view the relationship between learners and Chinese language and culture. As mentioned earlier, the way of thinking in the era of big data has broken the traditional concern for causality and shifted to the pursuit of relationships. After all, in this era, everyone is a node in a huge network that is inextricably linked to other people. In the age of the Internet, humans returned to the tribal era in which people know each other and do business with each other. In other words, in daily teaching, the relevance of educators concerns not only the content of teaching, but also the language learners and educators themselves. Broadly speaking, both learners and educators are all in a kind of social production relationship respectively. They are the epitome of a particular culture and are all in a web of meaning. To put it in another way, it is the ability to think across boundaries, which is the "universal wisdom" of the Internet age. Investor Charlie Munger hailed cross-border thinking as a "hammer." With this hammer, he would immediately rush to the barrier between different industries, and then use a broader perspective to look for ideas that could be ‘linked’. In the teaching of Chinese as a foreign language, educators and learners must often utilize this ‘hammer’ of relevance to every nail that is linked to what they have learned. Because the understanding of linguistic culture ‘is a process in which the subject of linguistic and action abilities is unified’ [5], and any language acquisition can only be linked to the internal and external environment, then it may become a possible reality. This related language acquisition, in Habermas's words, is ‘the background knowledge of acceptability conditions expressed through language standardization... This background knowledge is an implicit background knowledge’ [6]. The learner's newly acquired language and its background can only be grasped by integrating the whole structure into the well-known background knowledge of the living world and supporting each other. If learners do not realize it as they wish, it is impossible to master and use this new acquisition language.

In addition, we make full use of the means provided by the Internet to enrich teaching activities. As mentioned earlier, the new generation of learners belonged to digital natives and adapted to the survival of the visual culture era that was full of audiovisual language, and their 'spiritual world carries a mythological spiritual trait, emotional, intuitive, fantasy, romantic, full of passion and vitality'[7]. ‘Media is an extension of people.’ McLuhan's another famous assertion. In his view, the print media's text mobilizes people's abstract thinking ability, ignoring the perception of people's senses; the broadcasting media emphasizes auditory cognition. Although television media mobilizes people's sensory system, it is slightly that fleeting feature ignores people's abstract thinking ability. As a brand-new media integration of print media, broadcasting media, and television media, the Internet once again integrates the perceptions of human senses and focuses on people who are fully developed. Moreover, with the wide application of VR technology, the forms of teaching Chinese as a foreign language will become more abundant. It is of great significance to mobilize the interest of cultivating learners and stimulate the recognition of all aspects of Chinese language and culture.

Conclusion

For the learner who learns Chinese as a second language, the internet age is definitely the best time in which there are all kinds of free, shared, and open learning resources. For the educator who teaches Chinese as a second language, how to use internet thinking and maximize the advantages of this era will also be a new problem faced by it. After all, the internet has its disadvantageous aspects as well. Its fragmentation, shallowness, and unqualified and weak characteristics are also...
easily detected during being used. This requires the educators during the daily teaching of Chinese keep vigilant to the side effect of using internet and guide the learners timely, help them select and filter the information, and correct the wrong parts of the problematic sources to avoid mistakes existing as right ones.

References


# List of Papers

<table>
<thead>
<tr>
<th>#</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The existence of oscillatory solutions for a complex-valued neural network model with delay</td>
<td>333</td>
</tr>
<tr>
<td>5</td>
<td>Distance Estimation Based Energy Efficient Protocol for Wireless Sensor Network</td>
<td>339</td>
</tr>
<tr>
<td>7</td>
<td>LS- SVM’s no-reference video quality assessment model considering the network packet loss under the internet of things</td>
<td>344</td>
</tr>
<tr>
<td>8</td>
<td>SAF-AFIS: SIFT-based Alignment Free Approach for an Automatic Fingerprint Identification System</td>
<td>350</td>
</tr>
<tr>
<td>9</td>
<td>Unsupervised Anomaly Detection in Sewer Images with a PCA-based Framework</td>
<td>354</td>
</tr>
<tr>
<td>10</td>
<td>Survey of Issues with Text to Speech Synthesis of Multilingual Indian Texts</td>
<td>360</td>
</tr>
<tr>
<td>11</td>
<td>An overview on on-line handwriting analysis for the assessment of AD and PD</td>
<td>14</td>
</tr>
<tr>
<td>12</td>
<td>Identification of Bruised Apples Using Deep Learning and 3D Near-infrared Imaging</td>
<td>245</td>
</tr>
<tr>
<td>13</td>
<td>Gesture Synthesis for Human-Computer Interaction</td>
<td>758</td>
</tr>
<tr>
<td>14</td>
<td>Multi-label Pixelwise Classification for Reconstruction of Large-scale Urban Areas</td>
<td>195</td>
</tr>
<tr>
<td>16</td>
<td>A Robust Palmprint Matching Method</td>
<td>372</td>
</tr>
<tr>
<td>17</td>
<td>Evaluation of trackers for Pan-Tilt-Zoom Scenarios</td>
<td>170</td>
</tr>
<tr>
<td>18</td>
<td>AI-based Automated Japanese Short-answer Scoring and Support System</td>
<td>138</td>
</tr>
<tr>
<td>21</td>
<td>Illumination Invariant Face Recognition Via Dual-Tree Complex Wavelet Transform in Logarithm Domain</td>
<td>49</td>
</tr>
<tr>
<td>24</td>
<td>Training Quantized Nets with Adaptive Shared Exponents Based on Statistical Distributions</td>
<td>83</td>
</tr>
<tr>
<td>25</td>
<td>A New Approach to Train Convolutional Neural Networks for Monocular Visual Odometry</td>
<td>66</td>
</tr>
<tr>
<td>26</td>
<td>Facial Expression Recognition Using a Multi-level Convolutional Neural Network</td>
<td>217</td>
</tr>
<tr>
<td>27</td>
<td>Some observations on lognormality and motor control in handwriting</td>
<td>732</td>
</tr>
<tr>
<td>28</td>
<td>Objective Identification of Bullets Based on 3D Pattern Matching and Line Counting Scores</td>
<td>290</td>
</tr>
<tr>
<td>29</td>
<td>Visual Detection of Fake Coins Using Fisher Vectors</td>
<td>296</td>
</tr>
<tr>
<td>31</td>
<td>A novel female facial beauty predictor</td>
<td>378</td>
</tr>
<tr>
<td>33</td>
<td>Counterfeit Coin Detection Based on Image Content By Fuzzy Association Rules Mining</td>
<td>285</td>
</tr>
<tr>
<td>34</td>
<td>Restoring height-map images of shiny coins using spline approximation to detect counterfeit coins</td>
<td>383</td>
</tr>
<tr>
<td>41</td>
<td>Automatic High-Speed Compressive Tracking with Motion Prediction</td>
<td>388</td>
</tr>
<tr>
<td>42</td>
<td>Dissimilarity-based representation for radiomics applications</td>
<td>53</td>
</tr>
<tr>
<td>43</td>
<td>Image Quality Enhancement for Document Extraction with Mobile Devices</td>
<td>19</td>
</tr>
<tr>
<td>50</td>
<td>The Handwritten Chinese Character Recognition use Convolutional neural networks with the GoogLenet</td>
<td>2</td>
</tr>
<tr>
<td>51</td>
<td>Off-line Handwritten Chinese Recognition and Improvement Based on Caffe Framework</td>
<td>394</td>
</tr>
<tr>
<td>52</td>
<td>Attention-Based Recurrent Neural Networks in Chinese Short Text Classification</td>
<td>399</td>
</tr>
<tr>
<td>53</td>
<td>Quaternion Linear Regression for Color Face Recognition</td>
<td>403</td>
</tr>
<tr>
<td>56</td>
<td>Computational Optimization for Fast and Robust Automatic Segmentation in Virtual Microscopy using Brute-force-based Feature Selection</td>
<td>407</td>
</tr>
<tr>
<td>59</td>
<td>Improving Multiple Object Tracking with Optical Flow and Edge Preprocessing</td>
<td>158</td>
</tr>
<tr>
<td>60</td>
<td>Novel Graph-based Image Segmentation: Application to Medical Imaging</td>
<td>413</td>
</tr>
<tr>
<td>63</td>
<td>Hybrid Deep Neural Network for Visual Phrase Detection</td>
<td>78</td>
</tr>
<tr>
<td>64</td>
<td>Fractal-windowed based Empirical Mode Decomposition Scheme for Protein Sequence Analysis</td>
<td>419</td>
</tr>
<tr>
<td>65</td>
<td>A new ConvNet architecture for heartbeat classification</td>
<td>424</td>
</tr>
<tr>
<td>68</td>
<td>Naive Bayes Classification for Subset Selection in a Multi-label Setting</td>
<td>204</td>
</tr>
<tr>
<td>69</td>
<td>Algorithms for hierarchical segmentation based on the Felzenszwalb-Huttenlocher dissimilarity</td>
<td>108</td>
</tr>
<tr>
<td>71</td>
<td>Fractal Modeling of Big Data Networks</td>
<td>429</td>
</tr>
<tr>
<td>75</td>
<td>Classification of the Levels of Consciousness within Non-Rapid Eye Movement Sleep</td>
<td>315</td>
</tr>
<tr>
<td>76</td>
<td>Real-time interpretation of geometric shapes for digital learning</td>
<td>31</td>
</tr>
<tr>
<td>77</td>
<td>Application of Pattern Recognition in Mineral Segmentation and Identification</td>
<td>433</td>
</tr>
<tr>
<td>No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>78</td>
<td>Sequential Transformation with Geometric Constraints for Matching Oblique Aerial Images</td>
<td>439</td>
</tr>
<tr>
<td>79</td>
<td>Improving on-line signature skillfulness</td>
<td>768</td>
</tr>
<tr>
<td>80</td>
<td>Counterfeit Coin Detection Using Stamp Features and Convolutional Neural Network</td>
<td>273</td>
</tr>
<tr>
<td>81</td>
<td>On dynamic ensemble selection and data preprocessing for multi-class imbalance learning</td>
<td>189</td>
</tr>
<tr>
<td>82</td>
<td>A Novel Way of Identifying Cyber Predators</td>
<td>279</td>
</tr>
<tr>
<td>84</td>
<td>Stochastic Modeling of Camera Errors for Stereo Image Processing</td>
<td>445</td>
</tr>
<tr>
<td>86</td>
<td>Cubes3D: Neural Network based Optical Flow in Omnidirectional Image Scenes</td>
<td>164</td>
</tr>
<tr>
<td>87</td>
<td>Online Semi-Supervised Learning with Adaptive Vector Quantization</td>
<td>461</td>
</tr>
<tr>
<td>88</td>
<td>Recent Developments from Attribute Profiles for Remote Sensing Image Classification</td>
<td>102</td>
</tr>
<tr>
<td>89</td>
<td>Improving Chinese Writer Identification by Fusion of Text-dependent and Text-independent Methods</td>
<td>25</td>
</tr>
<tr>
<td>91</td>
<td>Integrating Learning and Reasoning Services for Explainable Information Fusion</td>
<td>60</td>
</tr>
<tr>
<td>92</td>
<td>Pattern Structure of Human Motion Using Single Channel CW Doppler Radar: An Unsupervised Perspective.</td>
<td>467</td>
</tr>
<tr>
<td>95</td>
<td>Support Vector Machine As Graph Theory Problems</td>
<td>183</td>
</tr>
<tr>
<td>96</td>
<td>Sequential Minimal Optimization Extended to General Quadratic Programming</td>
<td>473</td>
</tr>
<tr>
<td>97</td>
<td>Extending the Sigma-Lognormal Model of the Kinematic Theory to Three Dimensions</td>
<td>748</td>
</tr>
<tr>
<td>98</td>
<td>Cardiac Murmur Classification in Phonocardiograms using Deep Convolutional Neural Networks</td>
<td>261</td>
</tr>
<tr>
<td>99</td>
<td>An Investigation of Discrete Hidden Markov Models on Handwritten Short Answer Assessment System</td>
<td>144</td>
</tr>
<tr>
<td>100</td>
<td>Stacked Kernel Extreme Learning Machine for Hyperspectral Image Classification</td>
<td>481</td>
</tr>
<tr>
<td>101</td>
<td>A Block-based Path Recognition of Slag Removal Using Convolutional Neural Network</td>
<td>487</td>
</tr>
<tr>
<td>102</td>
<td>Local Binary Pattern Mapping on Graph-based Image Representation for Texture Classification</td>
<td>492</td>
</tr>
<tr>
<td>105</td>
<td>Segmentation-Free Cell Phenotype Classification using Deep Residual Neural Networks</td>
<td>72</td>
</tr>
<tr>
<td>106</td>
<td>Developing Image Processing Algorithms in Julia</td>
<td>43</td>
</tr>
<tr>
<td>107</td>
<td>Ontology-driven Acquisition of Verbal and Nominalization Patterns for Criminal Events</td>
<td>498</td>
</tr>
<tr>
<td>108</td>
<td>Classification of Keyphrases using Random Forest and Latent Semantic Analysis</td>
<td>506</td>
</tr>
<tr>
<td>110</td>
<td>Semi-supervised product quantization for approximate nearest neighbor search</td>
<td>210</td>
</tr>
<tr>
<td>111</td>
<td>Efficient fine-grained road segmentation using superpixel-based CNN and CRF models</td>
<td>512</td>
</tr>
<tr>
<td>113</td>
<td>Bag-of-features for clustering online handwritten mathematical expressions</td>
<td>127</td>
</tr>
<tr>
<td>114</td>
<td>Clustering Offline Handwritten Mathematical Answers for Computer-Assisted Marking</td>
<td>121</td>
</tr>
<tr>
<td>115</td>
<td>A common framework to evaluate Parkinson’s disease in voice and handwriting</td>
<td>795</td>
</tr>
<tr>
<td>116</td>
<td>Image Classification using Collaborative Mean Attraction with Sparse Optimization</td>
<td>518</td>
</tr>
<tr>
<td>117</td>
<td>Deep Error Correcting Output Codes</td>
<td>250</td>
</tr>
<tr>
<td>118</td>
<td>Generalization of Parameter Recovery in Binocular Vision for a Planar Scene</td>
<td>37</td>
</tr>
<tr>
<td>119</td>
<td>Saliency and Object Detection</td>
<td>523</td>
</tr>
<tr>
<td>120</td>
<td>Modeling the Complexity of Biomechanical Tasks using the Lognormality Principle: Applications to Signature Recognition and Touch-screen Children Detection</td>
<td>774</td>
</tr>
<tr>
<td>122</td>
<td>Sequential Backward Spatio-Spectral Filter Optimization for Motor Imagery Classification in Ear-EEG Brain-Computer Interface</td>
<td>227</td>
</tr>
<tr>
<td>124</td>
<td>SDRN: Scalable Deep Rectifier Network for opinion spam detection</td>
<td>535</td>
</tr>
<tr>
<td>126</td>
<td>Enhancing Human Action Recognition through Temporal Saliency</td>
<td>176</td>
</tr>
<tr>
<td>127</td>
<td>Intrinsic Quality Analysis of Binary Partition Trees</td>
<td>0</td>
</tr>
<tr>
<td>128</td>
<td>Finger-vein quality assessment by joint representation learning from grayscale and binary images</td>
<td>232</td>
</tr>
<tr>
<td>129</td>
<td>Automatic video stream selection method by on-air microphone detection</td>
<td>309</td>
</tr>
<tr>
<td>133</td>
<td>Animation Generation with a Low-Dimensional Simplicial Complex</td>
<td>325</td>
</tr>
<tr>
<td>135</td>
<td>Kinematic Reconstruction of Calligraphic Traces from Shape Features</td>
<td>762</td>
</tr>
<tr>
<td>140</td>
<td>Hybrid RUSBoost Versus Data Sampling to Address Data Imbalance for Breast Cancer Cytological Malignancy Grading</td>
<td>545</td>
</tr>
<tr>
<td>Page</td>
<td>Title</td>
<td>Authors</td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>142</td>
<td>Failure modelling and anomaly detection of a propulsion subsystem</td>
<td></td>
</tr>
<tr>
<td>143</td>
<td>Toward Scalable Visual Digital Evidence Visualization and Multimodal Interaction using Computer Vision Techniques</td>
<td></td>
</tr>
<tr>
<td>144</td>
<td>Kinematic Analysis of Fast Pen Strokes in Children with ADHD using the Sigma-lognormal Model</td>
<td></td>
</tr>
<tr>
<td>145</td>
<td>A generalized unified discrete linear method for edge detection by antisymmetric FIR kernels</td>
<td></td>
</tr>
<tr>
<td>146</td>
<td>Machine learning and feature selection for the analysis of Alzheimer Metabolomics Data</td>
<td></td>
</tr>
<tr>
<td>148</td>
<td>Table-based Document Classification in Historical Document Images</td>
<td></td>
</tr>
<tr>
<td>151</td>
<td>Hybrid Image Representation Method based on Bag of Edge Tokens from Octaves of Edge Elements</td>
<td></td>
</tr>
<tr>
<td>152</td>
<td>Deep Learning-Based Corresponding Points Fast Matching</td>
<td></td>
</tr>
<tr>
<td>154</td>
<td>Posture and Fall Detection System Using 3D Motion Sensors</td>
<td></td>
</tr>
<tr>
<td>157</td>
<td>Object cosegmentation using deep Siamese network</td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>A New Roadmap for Evaluating Descriptive Handwritten Answer type</td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>Fast Context-Annotated Classification of Different Types of Web Service Descriptions</td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>Classification of Human Activity Level Using Single Channel CW Doppler Radar</td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>Comparative Study of a Shape-Based and a Texture-Based Feature Extraction Technique for Mass Classification in Digital Mammograms</td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>Comparative Study of Iterative Back Projection and Discrete Algebraic Reconstruction Techniques for Reconstruction of Low Resolution Images</td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>Distributed Component Forests: Hierarchical Image Representations Suitable for Tera-Scale Images</td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>The Delta-Lognormal model in 2.5D</td>
<td></td>
</tr>
<tr>
<td>166</td>
<td>Exploiting the Lognormality Principle: Three Ongoing Projects</td>
<td></td>
</tr>
<tr>
<td>168</td>
<td>A New Experimental Set-up To Run Neuromuscular Tests</td>
<td></td>
</tr>
<tr>
<td>171</td>
<td>Deep Learning is not a Matter of Depth but of Good Training (Extended Abstract)</td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>Computational Graphology Applied to Handwriting Images</td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>Approximation-based transformation of color signal for heart rate estimation with a webcam</td>
<td></td>
</tr>
<tr>
<td>176</td>
<td>Using optimal circular path method to match piecewise iris templates</td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>Dynamic-HOG Descriptor for Structured Object Recognition: Case Study on Coins</td>
<td></td>
</tr>
<tr>
<td>178</td>
<td>Dataless Black-Box Model Comparison</td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>Increase efficiency of simple images segmentation using detectors based on doubly stochastic random fields</td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>A post-processing method for 3D fluorescence microscopy images</td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>In Defense of Active Part Selection for Fine-Grained Classification</td>
<td></td>
</tr>
<tr>
<td>182</td>
<td>Missing data imputation based on stochastic neighbor embedding</td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>Big Data Application for Smart Features Formation in Medical Diagnostic Tasks</td>
<td></td>
</tr>
<tr>
<td>184</td>
<td>Nonlinearity of iris structure as way to improve recognition methods</td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>Shape of Basic Clusters: Finding Coherent ELR-2s via Hough-type Transform</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>Multimodal image analysis for power line inspection</td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>Precedent-based Low Count Rate Image Intensity Estimation using Maximum Likelihood Distribution Descriptions</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>Development of the filtering algorithm for doubly stochastic images based on models with multiple roots of characteristic equations</td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>Underwater Image Enhancement Algorithm Based on Logarithmic Transform Histogram Matching With Spatial Equalization</td>
<td></td>
</tr>
<tr>
<td>192</td>
<td>Iris Anti-spoofing Solution for Mobile Biometric Applications</td>
<td></td>
</tr>
<tr>
<td>193</td>
<td>Iris Segmentation in Challenging Conditions</td>
<td></td>
</tr>
<tr>
<td>194</td>
<td>A new mathematical method for automated identification of neurons on microscopic images</td>
<td></td>
</tr>
<tr>
<td>195</td>
<td>Separation of arterial and venous vessels images from computer tomography</td>
<td></td>
</tr>
<tr>
<td>Page</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>196</td>
<td>Training a classifier for automatic flash detection in million images from camera-traps</td>
<td>589</td>
</tr>
<tr>
<td>197</td>
<td>Automatic Video Mining in Animal Behavior Study using Statistical Shape Models</td>
<td>607</td>
</tr>
</tbody>
</table>