University of Groningen

Local-LDA

Kasaei, S. Hamidreza; Lopes, Luis Seabra; Tome, Ana Maria

Published in:
IEEE transactions on pattern analysis and machine intelligence

DOI:
[10.1109/TPAMI.2019.2926459](https://doi.org/10.1109/TPAMI.2019.2926459)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2020

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Kasaei, S. H., Lopes, L. S., & Tome, A. M. (2020). Local-LDA: Open-Ended Learning of Latent Topics for 3D Object Recognition. *IEEE transactions on pattern analysis and machine intelligence*, 42(10), 2567-2580. [8753557]. <https://doi.org/10.1109/TPAMI.2019.2926459>

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Local-LDA: Open-Ended Learning of Latent Topics for 3D Object Recognition

S. Hamidreza Kasaei[✉], Luís Seabra Lopes[✉], and Ana Maria Tomé

Abstract—Service robots are expected to be more autonomous and work effectively in human-centric environments. This implies that robots should have special capabilities, such as learning from past experiences and real-time object category recognition. This paper proposes an open-ended 3D object recognition system which concurrently learns both the object categories and the statistical features for encoding objects. In particular, we propose an extension of Latent Dirichlet Allocation to learn structural semantic features (i.e., visual topics), from low-level feature co-occurrences, for each category independently. Moreover, topics in each category are discovered in an unsupervised fashion and are updated incrementally using new object views. In this way, the advantages of both the (hand-crafted) local features and the (learned) structural semantic features have been considered and combined in an efficient way. An extensive set of experiments has been performed to assess the performance of the proposed Local-LDA in terms of descriptiveness, scalability, and computation time. Experimental results show that the overall classification performance obtained with Local-LDA is clearly better than the best performances obtained with the state-of-the-art approaches. Moreover, the best scalability, in terms of number of learned categories, was obtained with the proposed Local-LDA approach, closely followed by a Bag-of-Words (BoW) approach. Concerning computation time, the best result was obtained with BoW, immediately followed by the Local-LDA approach.

Index Terms—Latent dirichlet allocation (LDA), local-LDA, open-ended learning, 3D object learning and recognition, object perception

1 INTRODUCTION

ONE of the primary goals in computer vision is to develop reliable capabilities that will allow robots to robustly recognize objects and interact with their environment. It is still a challenging problem because of ill-definition of objects, large variations in object appearances and concept drift in dynamic environments. To migrate a robot to a new environment one must often completely re-generate the knowledge base that it is running with. Since in open-ended domains the set of categories to be learned is not predefined, it is not feasible to assume that one can pre-program all object categories required by robots. Therefore, autonomous robots must have the ability to continuously execute learning and recognition in a concurrent and interleaved fashion [1]. Furthermore, in order to interact with human users and environments, this process of object recognition cannot take more than a fraction of a second.

Humans learn to recognize object categories ceaselessly over time. This ability allows them to adapt to new

environments, by enhancing their knowledge from the accumulation of experiences and the conceptualization of new object categories. Cognitive scientists demonstrated that, in humans, object recognition skills and the underlying capabilities are developed concurrently [2]. Moreover, they showed humans use hierarchical recognition to handle large-scale object recognition tasks. In hierarchical recognition theories, the human sequentially processes information about the target object leading to the recognition result [2]. Similarly, an autonomous robot, apart from learning from a batch of labelled training data, must process visual information continuously and perform learning and recognition simultaneously and in an open-ended manner. This is important since, no matter how extensive is the training data used for batch learning, a robot might always be confronted with an *unknown* object when operating in an everyday environment. It should be noted that the target set of categories is predefined in standard incremental learning, while open-ended learning implies that the set of object categories to be learned is not known in advance. The training instances are extracted from on-line experiences of a robot, and thus become gradually available over time, rather than being completely available at the beginning of the learning process.

Classical object recognition approaches do not perform well in these scenarios because they are typically designed for static environments i.e., training (offline) and testing (online) are two separated phases. If limited training data is used, this might lead to non-discriminative object representations and, as a consequence, to poor object recognition performance [3]. Therefore, building a discriminative object representation is a challenging step to improve object recognition performance. Most of the traditional approaches use

- S. H. Kasaei is with the Institute of Artificial Intelligence and Cognitive Engineering, University of Groningen, PO Box 407, Groningen 9700, AK, The Netherlands, and also with the IEETA - Instituto de Engenharia Electrónica e Telemática de Aveiro University of Aveiro, Aveiro 3810-193, Portugal. E-mail: hamidreza.kasaei@rug.nl.
- L. S. Lopes and A. M. Tomé are with the IEETA - Instituto de Engenharia Electrónica e Telemática de Aveiro University of Aveiro, Aveiro 3810-193, Portugal, and also with the Department of Electronics, Telecommunications and Informatics, University of Aveiro, Aveiro 3810-193, Portugal. E-mail: {lsl, ana}@ua.pt.

Manuscript received 11 Oct. 2018; revised 14 June 2019; accepted 16 June 2019. Date of publication 2 July 2019; date of current version 2 Sept. 2020.

(Corresponding author: Seyed Hamidreza Kasaei.)

Recommended for acceptance by M. Bennamoun and Y. Guo.

Digital Object Identifier no. 10.1109/TPAMI.2019.2926459

Fig. 1. Proposed data processing layers being tested on a service robot.

hand-crafted features (e.g., [3], [4]). Recently, deep learning approaches, such as Convolutional Neural Networks (CNN) [5], [6], are being explored and used for object recognition tasks. Deep learning has major limitations. Deep learning is exceptionally data hungry. To automate a recognition task with a neural network, one must gather a massive dataset of what information the task requires, and what the task produces as output. The network is trained on this data, attempting to learn how to morph input into a proper output. Furthermore, the decision making processes within a neural network are not easy to interpret. Such limitations and requirements make them difficult to use in open-ended domains and therefore they are not the best option for such domains. Topic modelling [7] is suitable for open-ended learning because, not only it provides short object descriptions (i.e., optimizing memory), but also enables efficient processing of large collections. It is a type of generative latent structure model that represents the underlying structure of data as topics.

This paper proposes a 3D object recognition system capable of learning both object categories as well as the topics used to encode them concurrently and in an open-ended manner. We propose an extension of Latent Dirichlet Allocation to incrementally learn topics for each category independently. Topics in each category are discovered in an unsupervised fashion and updated incrementally using new object views. As depicted in Fig. 1, the approach is designed to be used by a service robot working in an unstructured environment. Tabletop objects are tracked (signaled by different bounding boxes in Fig. 1) and processed through five data processing layers. The first three layers (i.e., feature layer, BoW layer and topic layer) sequentially process a given object view to produce a powerful representation. In the feature layer, a spin-image shape descriptor [8] is used to represent the local shapes of the object in different key points; in the Bag-of-Words (BoW) layer, the given object view is described by histograms of local shape features, as defined in Bag-of-Words models; In the topic layer, we compute a set of visual topics for the given object. Each visual topic is defined as a discrete distribution over visual words. Since the views of an object can vary significantly depending on the perspective, it is crucial to store different views of the object (see Fig. 2, left). Therefore, in the view layer, we save the obtained topic representation for each object view.

Fig. 2. Examples of common challenges in view-based object recognition: (left) mug observed from different viewpoints; (right) intra-class variation among mug instances.

Another important point in 3D object recognition is that objects from the same category can have very different shapes (i.e., high intra-class variation). For instance, the Mug category contains many types of mugs with different shapes (see Fig. 2, left). The *category layer* addresses this point by associating distinctive instances to each category.

This work follows our previous work on hierarchical 3D object representation [9]. The present work aims to offer new insights from the practical point of view and provide benchmarking with the state-of-the-art approaches in both offline and online settings. In particular, we describe and visualize all the required steps by step, from preliminary steps such as key-point selection, spin-images computation and how words are represented to the most advanced step, i.e., visualizing learned topics on several objects. The remainder of this paper is organized as follows. In the next section, related work is discussed. The detailed methodologies are then explained in Sections 4 and 5. The experimental evaluation is presented in Section 6. Finally, conclusions are presented and future directions are discussed in Section 7.

2 RELATED WORK

Object representation is one of the most challenging tasks in robotics because it must provide reliable information in real-time to enable the robot to physically interact with the objects in its environment. For this purpose, the descriptor should contain enough information enabling to recognize the same or similar objects seen from different perspectives. Global descriptors encode the entire 3D object, while local descriptors represent a small area of an object around a specific key-point. Examples include Spin Images (SI) [3], Signature of Histograms of Orientations (SHOT) [4], Fast Point Feature Histogram (FPFH) [10], Hierarchical Kernel Descriptors [11] and Global Orthographic Object Descriptor (GOOD) [12], [13]. Local descriptors tend to handle occlusion and clutter better when compared to global descriptors. However, comparing 3D object views based on their local features tends to be computationally expensive [14]. To address this problem, machine learning approaches such as Bag of Words (BoW) [15], Latent Dirichlet Allocation (LDA) [7], [9] and deep learning [16], [17] can be used for representing objects in a uniform format, highly compact and distinctive way.

Towards this goal, Riesenhuber and Poggio [2] proposed a hierarchical approach for object recognition consistent with physiological data, in which objects are modelled in a hierarchy of increasingly sophisticated representations. Sivic et al. [18] proposed an approach to discover objects in images using Probabilistic Latent Semantic Indexing (pLSI) modelling [19]. Blei et al. argued that the pLSI is incomplete in that it provides no probabilistic model at the level of documents [7], [20]. They extended the pLSI model calling the

approach Latent Dirichlet Allocation (LDA). Similar to pLSI and LDA, we discover topics in an unsupervised fashion. Unlike our approach, pLSI and LDA do not incorporate class information.

Several works have been presented to incorporate a class label in the generative model [21]. Blei et al. [22] extend LDA and proposed Supervised LDA (sLDA), which was initially used for supervised text prediction. Later, Wang et al. [23] extended sLDA to classification problems. Another popular extension of LDA is the classLDA (cLDA) [24]. Similar to our approach, the only supervision used by sLDA and cLDA is the category label of each training object. However, there are two main differences. First, the learned topics in sLDA and cLDA are shared among all categories, while we propose to learn specific topics per category. Second, the sLDA and cLDA approaches follow a standard train-and-test procedure (i.e., the set of classes is known in advance and train and test data are available in advance), our approach can incrementally update topics using new observations and the set of classes is continuously growing. There are some topic-supervised approaches e.g., Labeled LDA [25] and semiLDA [26] that consider class labels for topics. On one hand, these approaches need tens of hours of manual annotation. On the other hand, it is hard for a human to provide a specific category label for a 3D local shape description (e.g., a spin-image [8]).

There are some LDA approaches that support incremental learning of object categories [27], [28] but not open-ended learning. The difference between incremental and open-ended learning is that the set of classes is predefined in incremental learning, while in open-ended learning the set of classes is continuously growing. Banerjee et al. [29] proposed online LDA (o-LDA) that is a simple modification of batch collapsed Gibbs sampling. It first applies the batch Gibbs sampler to the full dataset and then samples new topics for each newly observed word using information observed so far. Canini et al. [30] extended o-LDA and proposed an incremental Gibbs sampler for LDA (here referred to as I-LDA). The I-LDA does not need a batch initialization phase like o-LDA. In o-LDA and I-LDA, the number of categories is fixed, while in our approach the number of categories is growing. Moreover, o-LDA and I-LDA are used to discover topics shared among all categories, while our approach is used to discover specific topics per category. Geo et al. [31] and Zhang et al. [32] proposed approaches for fine-grained image categorization by learning category-specific dictionaries and a shared dictionary for all the categories. The works of Geo et al. [31] and Zhang et al. [32] are similar to ours in that they learned specific representation for each category. However, unlike our approach, the representations of known categories do not change after the training stage.

Unlike the above approaches, some researchers have recently adopted Deep Neural Networks (DNN) algorithms for 3D object representation, learning and recognition [16], [17]. However, there are several limitations to use DNN in open-ended domains. In general, DNN is incremental in nature but not open-ended, since the inclusion of novel categories enforces a restructuring in the topology of the network. These approaches also require a large collection of images to learn a compact and discriminative representation. As it was pointed out in [17], training a DNN for 3D object representation requires a large collection of 3D object

views to provide accurate representations and typically involves long training times. Schwarz et al. [5] used DNN for 3D object category learning. They clearly showed that the performance of DNN degrades when the size of the dataset is reduced.

3 OBJECT DETECTION IN CROWDED SCENES

Real 3D scenes generally consist of several objects that are often close to each other or stacked together. It is a challenging task to robustly detect multiple objects in a domestic environment due to severe occlusions and clutter. Moreover, some captured data can be inaccurate due to sensor noise. Throughout this work, we try to impose as few constraints as possible for the object detection. We assume objects are situated on a planar surface, as this is the common case in domestic environments, and transparent objects (e.g., glass) are not considered. We do not impose any other further constraints on the object's appearance.

Towards this end, the planar surface is first detected by finding the dominant plane in the point cloud. This is done using the RANSAC algorithm [33]. The algorithm starts by generating plane hypotheses based on three unique non-collinear points. For each plane hypothesis, distances from all points in the point cloud to the plane are computed. The plane hypotheses are then scored based on counting the number of points whose distance to the plane falls below a user-specified threshold, T . The RANSAC algorithm is repeated for a certain number of iterations, η . In the current implementation, $T = 20\text{mm}$ and $\eta = 200$. After detecting the table, it is possible to extract the points which lie directly above it. The mechanism we use to do this is called extraction of polygonal prisms.¹ After this step, we have a point cloud where all the objects that are on top of the table are included. Therefore, the extracted points are assumed to potentially belong to objects. The obtained point cloud is then segmented into individual clusters using the euclidean Cluster Extraction (ECE) algorithm.² Each small group of points will be treated as an object candidate. Two examples of object detection in isolated scenarios are shown in Figs. 3a and 3b).

In case of multiple objects touching each other (e.g., in a pile of objects or a messy dinning table), the ECE algorithm is not enough to appropriately detect object candidates and further processing is required (see Fig. 3c). In such situations, a hierarchical clustering procedure is used to segment the extracted point cloud using geometric data, surface normal and color information. A region of the given point cloud is considered as an object candidate whenever the points inside the region are continuous in both the space of orientations of surface normals and the space of depths. The depth continuity between every point and its neighbors is computed. If the distance between points is lower than a threshold, then the two points belong to the same region. A region growing segmentation algorithm³ is also applied on medium-size hypotheses. The purpose of this algorithm is

1. http://docs.pointclouds.org/1.0.0/classpcl_1_1_extract_polygonal_prism_data.html

2. http://www.pointclouds.org/documentation/tutorials/cluster_extraction.php

3. http://pointclouds.org/documentation/tutorials/region_growing_segmentation.php

Fig. 3. Examples of the object detection in two different scenarios: (a and b) result of the table detection in isolated objects scenarios; Detected object candidates are shown by different bounding boxes and colors. The red, green and blue lines represent the local reference frame of the objects; (c) As shown in this figure, euclidean clustering algorithm is not enough for such a crowded scene. Therefore, the extracted region is dispatched to a hierarchical clustering procedure to detect multiple object candidates; (d) two highly crowded scenes and their corresponding segmentation results.

to merge the points that are close enough in terms of the smoothness and color constraints. Each cluster of points is finally treated as an object candidate, O . It should be noted that the number of clusters is not pre-defined and varies for different viewpoints. Fig. 3d illustrates the results of the segmentation process in two different pile scenarios. Note that the point clouds of each object have different colors, meaning that they have been correctly segmented. In the following section, we assume that an object has already been segmented from the point cloud of the scene, and we will focus on detailing the Local-LDA model. For additional details on the object segmentation, we refer the reader to our previous works on 6D object pose estimation [34] and 3D object recognition [35], [36]. Note that, in the absence of full objects models, it is not possible to segment all objects precisely in crowded scenarios. In such situations, for further improving the object detection result, the robot could interact with the objects and verify which detected clusters belong to the same object [37], [38]. However, such functionality is out of the scope of this paper.

4 OBJECT REPRESENTATION LEARNING

Comparing 3D objects based on their local features is computationally expensive. To address this problem, we propose a multi-layered perception architecture that builds an increasingly complex object representation (see Fig. 1). Particularly, a statistical model is used to get structural semantic features from low-level feature co-occurrences. The basic idea is to represent the objects as random mixtures over local topics (i.e., latent variables), where each topic is characterized by a distribution over visual words (i.e., observed variables). It must be pointed out that we are using shape features and topic modeling to find out the hidden semantic structure of the data in each object category.

4.1 Data Processing Layers

This approach consists of five data processing layers including feature layer, BoW layer, topic layer, object view layer and category layer. The *feature layer* involves two main phases: keypoint selection and spin image computation for each keypoint. For keypoint selection, a voxelized grid approach⁴ is used to obtain a smaller set of points by taking only the nearest neighbor point for each voxel center (see Figs. 4a and b). Afterwards, the spin-image descriptor is used to encode the surrounding shape in each keypoint using the original point cloud (i.e., *feature layer*) (see Figs. 4c and 4d). Therefore, each object view, O , is described by a set of spin-images [8], $\{s_1, s_2, \dots, s_N\}$, where N is the number of keypoints. Spin-images are pose invariant, and therefore suitable for 3D perception in autonomous robots. To compute spin-images, a function from the Point Cloud Library (PCL) is used.⁵ Subsequently, the spin-images go “up” to the *BoW layer*, where an object is represented as a set of visual words, $\{w_1, w_2, \dots, w_N\}$, where each entry represents the index of one of the V words of the dictionary (i.e., *BoW layer*). It is worth to mention that spin-images can be replaced by any other hand-crafted or deep learning based local shape descriptor. Furthermore, if a set of data is available at the beginning of the learning process, we can learn the visual word dictionary using deep Sparse Autoencoders [39], [40].

The obtained representation is then presented as input to the *topic layer*. In this layer, each word, $w_i, i = 1, \dots, N$, is mapped to one of the topics of the category to which the object belongs. The object is therefore described as a set of topics $z = \{z_1, z_2, \dots, z_N\}$, where each element of z represents the index of one of K category-specific topics. Towards this goal, a new *Local-LDA* model is presented. The details of dictionary construction and statistical model will be discussed in the following subsections.

4.2 Dictionary Construction

This approach requires a dictionary with V visual words. In this work, the dictionary is created via off-line clustering of training data. We create a pool of objects using 75 percent of the training data. Previously, we showed how a robot can

Fig. 4. Object representation for a coffee cup : (a) point cloud of a coffee cup (b) key-point selection (blue points); spin-image descriptor is used to encode the surrounding shape in each key-point; (c) a schematic of how spin-image is computed for a keypoint p ; (d) the computed spin-image.

4. http://docs.pointclouds.org/trunk/classpcl_1_1_voxel_grid.html

5. http://docs.pointclouds.org/1.7.0/classpcl_1_1_spin_image_estimation.html

Fig. 5. A sample set of visual words of a dictionary: spin-images compiled with IW = 4 bins and SL = 0.05m.

create a pool of objects by exploring the environment [9], [41]⁶ and how the dictionary can be updated in an online fashion [42]. To construct a pool of features, the spin-image descriptor [8] is used to encode the surrounding shape in each point. Finally, the dictionary is constructed by clustering the pool of features using the k-means algorithm. The centers of the V extracted clusters are defined as visual words, $\mathbf{w}_i$ ($1 \leq i \leq V$). Fig. 5 shows a sample set of visual words of a dictionary.

4.3 Local-LDA Model

We use a statistical model to extract structural semantic features from low-level feature co-occurrences. The basic idea is to represent the objects as random mixtures over visual topics (i.e., latent variables), where each topic is characterized by a distribution over visual words (i.e., observed variables). Both random distributions are defined using multinomial distributions and their parameters are characterized by a Dirichlet distribution. It must be pointed out that we are using shape features rather than semantic properties to encode the statistical structure of object categories [44]. It is easier to explain the details using an example. We start by selecting a category label, for example *Mug*. To represent a new instance of *Mug*, a distribution over *Mug* topics is drawn that will specify which topics should be selected for generating each visual word of the object. According to this distribution, a particular topic is selected out of the mixture of possible topics of the *Mug* category for generating each visual word in the object. For instance, a *Mug* usually has a handle, and a “handle” topic refers to some visual words that frequently occur together in handles. The process of drawing both the topic and visual word is repeated several times to choose a set of visual words that would construct a *Mug*. We use statistical inference techniques for inverting this process to automatically find out a set of topics for each category from a collection of instances. In other words, we try to learn a model for each category (a set of latent variables) that explains how each object obtains its visual words.

Towards this end, the obtained representation from the *BoW layer* is presented as input to the *topic layer*. The LDA model contains parameters at three levels: including category-level parameters (i.e., α), which are sampled once in the process of generating a category of objects; object-level variables (i.e., θ_o), which are sampled once per object, and word-level variables (i.e., $\mathbf{z}_{o,n}$ and $w_{o,n}$), which are sampled every time a feature is extracted. The variables θ , ϕ and $\mathbf{z}$ are latent variables that should be inferred. Assume everything is observed and a category label is selected for each object; i.e., each object belongs to one category. The joint distribution of all hidden and observed variables for a category is defined as follows:

6. A sample video of creating a pool of objects by exploring an office environment is available online at: <http://youtu.be/MwX3J6aoAX0>.

$$p^{(c)}(\mathbf{w}, \mathbf{z}, \theta, \phi | \alpha, \beta) = \prod_{z=1}^K p^{(c)}(\phi_z | \beta) \prod_{o=1}^{|c|} p^{(c)}(\theta_o | \alpha) \prod_{n=1}^N p^{(c)}(\mathbf{z}_{o,n} | \theta_o) p^{(c)}(w_{o,n} | \mathbf{z}_{o,n}, \phi), \quad (1)$$

where α and β are Dirichlet prior hyper-parameters that affect the sparsity of distributions. Although, α and β could be vector-valued, we assume symmetric Dirichlet priors, with α and β each having a single value. K is the number of topics, $|c|$ is the number of known objects in the category c and N is the number of words in object o . Each $\theta_o^{(c)}$ represents an instance of category c in topic-space as a histogram (i.e., *topic layer*), $\mathbf{w}$ represents an object as a vector of visual words, $\{w_1, w_2, \dots, w_N\}$, where each entry represents the index of one of the V words of the dictionary (i.e., *BoW layer*). Next, the object is described as a set of topics $\{z_1, z_2, \dots, z_N\}$, where each element of $\mathbf{z}$ represents the index of one of the K topics and then the object is represented as a histogram of topics. ϕ is a $K \times V$ matrix, which represents word-probability matrix for each topic, where $\phi_{i,j} = p^{(c)}(w_i | z_j)$; thus, the posterior distribution of the latent variables given the observed data is computed as follows:

$$p^{(c)}(\mathbf{z}, \theta, \phi | \mathbf{w}, \alpha, \beta) = \frac{p^{(c)}(\mathbf{w}, \mathbf{z}, \theta, \phi | \alpha, \beta)}{p^{(c)}(\mathbf{w} | \alpha, \beta)}, \quad (2)$$

As [20] showed, the denominator of the Equation (2) is *intractable* and cannot be computed exactly due to a large number of possible instantiations of the topic structure. Griffiths et al., [45] proposed an analytical solution based on collapsed Gibbs sampling [46] that can be used. Since θ and ϕ can be derived from z_i , they are integrated out from the sampling procedure. In this work, for each category, an incremental LDA model is created. Whenever a new training instance is presented, the collapsed Gibbs sampling is employed to update the parameters of the model. The collapsed Gibbs sampler is used to estimate the probability of topic z_i being assigned to a word w_i , given all other topics assigned to all other words:

$$p^{(c)}(z_i = k | \mathbf{z}_{-i}, \mathbf{w}) \propto p^{(c)}(z_i = k | \mathbf{z}_{-i}) \times p^{(c)}(w_i | z_i, \mathbf{w}_{-i}) \propto \frac{n_{o,k,-i}^{(c)} + \alpha}{\sum_{k=1}^K [n_{o,k}^{(c)} + \alpha] - 1} \times \frac{n_{w,k,-i}^{(c)} + \beta}{\sum_{w=1}^V n_{w,k}^{(c)} + \beta}, \quad (3)$$

where $n_{\cdot}^{(c)}$ shows the relative counters of category c ; $\mathbf{z}_{-i}$ means all hidden variables except z_i , and $\mathbf{z} = \{z_i, \mathbf{z}_{-i}\}$, and $n_{o,k}^{(c)}$ is the number of times topic k from category c is assigned to some visual word in object o and $n_{w,k}^{(c)}$ shows the number of times visual word w is assigned to topic k . After the iterative sampling procedure, the multinomial parameter sets $\theta^{(c)}$ (i.e., object-topic matrix) and $\phi^{(c)}$ (i.e., topic-word matrix) can be estimated using the following equations:

$$\theta_{k,o}^{(c)} = \frac{n_{o,k}^{(c)} + \alpha}{n_o + K\alpha}, \quad \text{and} \quad \phi_{w,k}^{(c)} = \frac{n_{w,k}^{(c)} + \beta}{n_k^{(c)} + V\beta}. \quad (4)$$

Fig. 6. Visualization of five sample topics learned from the Washington RGB-D dataset [43].

where $n_k^{(c)}$ is the number of times a word assigned to topic k in category c and n_o is the number of words in the object o . It is worth mentioning since we compute a model for each category independently, we assumed that in a certain category, object views have an almost similar size. Therefore, the size of object is not important and the denominator of $\theta^{(c)}$ can be omitted. It is also possible to consider the size of instances in each category. An intuitive approach is to use small alpha for small object and vice versa. This is equivalent to use different symmetric Dirichlet prior on objects with different sizes, i.e., $\alpha_o = K \times |\mathbf{O}|$ where $|\cdot|$ defines the size of object and K stands for number of topics.

In Local-LDA, what happens next depends only on the current state of the system and not on the sequence of previous states. Therefore, whenever a new training instance is taught by an instructor (i.e., supervised learning), the collapsed Gibbs sampler is employed to first estimate a topic for each word of the given instance and then update the parameters of the model including $n_k^{(c)}$ and $n_{w,k}^{(c)}$ incrementally (i.e., unsupervised learning). Fig. 6 represents the five sample topics learned from the Washington RGB-D object dataset [43]. The proposed procedure is summarized in Algorithm 1.

5 OBJECT CATEGORY LEARNING AND RECOGNITION

Learning methods used in most of the classical object recognition systems are not designed for open-ended domains, since they do not support an incremental update of the internal robot's knowledge based on new experiences and they are not able to extend the set of categories as needed in real-world scenarios. Our approach is interactive and open-ended. We let the user identify the need for interactively teaching a new category or for providing corrective feedback when s/he notices recognition errors in robot's responses [47], [48]. In particular, the user can perform the following basic actions:

- *point*, used to bring the robot's attention to a particular object,
- *teach*, for teaching the category of the target object,
- *ask*, for asking the robot what is the category of the target object, and
- *correct*, for providing corrective feedback in case of misclassification.

Concerning category formation, an instance-based learning (IBL) approach is adopted, in which a category is represented by a set of views of instances of the category:

$$c = \{\mathbf{O}_1, \dots, \mathbf{O}_n\}, \quad (5)$$

where $\mathbf{O}_i$ are the constituent views. An advantage of the IBL approach is that it facilitates incremental learning in an open-ended fashion. Moreover, the instance-based approach is a

baseline approach to evaluate object representations for the purpose of object recognition. The agent only stores an instance when it fails to recognize its category and the teacher provides a correction.

To assess the dissimilarity between a target object view and stored instances of a certain category c , the target object should be described based on the learned topics of the category $\theta_T^{(c)}$ as a histogram. To this end, the target object and a temporary copy of the $n_{w,k}^{(c)}$ and $n_k^{(c)}$ counters are first loaded into the Gibbs sampling procedure. Then, a topic for each word of the target object is estimated. Finally, the target object is represented as a topic frequency histogram. The Symmetric Kullback Leibler Divergence, $D_{SKLD}(\theta_T^{(c)}, \theta_S^{(c)})$, is used as a dissimilarity measure. Afterwards, the minimum distance between the target object and all stored instances of the category c , is considered as the Object-Category Distance (OCD) and the target object is finally classified based on the minimum OCD(.).

Algorithm 1. Local LDA Learning Procedure

Input:

$W \leftarrow$ dictionary of visual words

$V \leftarrow$ size of dictionary

$K \leftarrow$ number of topics

$G \leftarrow$ number of Gibbs sampling iterations

$\alpha, \beta \leftarrow$ Dirichlet prior hyper-parameters

object_view o is given as a sequence of N visual words, $\{w_1, \dots, w_N\}$

Input/Output:

count_matrices $n_{o,k}, n_{w,k}, n_k$

Initialization:

for all visual words $w_i : i \in \{1, \dots, N\}$ **do**

$k \leftarrow z_i$ $\triangleright$ sample a topic index randomly

increment $n_{o,k}, n_{w,k}, n_k$

end for

Inference:

for $it \in \{1, \dots, G\}$ **do** $\triangleright$ Gibbs sampling

for all visual words $w_i : i \in \{1, \dots, N\}$ **do**

$k \leftarrow z_i$ $\triangleright$ sample a topic index

decrement $n_{o,k}, n_{w,k}, n_k$

for $k \in \{1, \dots, K\}$ **do**

$p[k] \leftarrow p[k-1] + (n_{o,k} + \alpha) \times \frac{n_{w,k} + \beta}{n_k + V\beta}$

end for

$u \leftarrow$ draw from uniform $[0, 1]$

for $k \in \{1, \dots, K\}$ **do** $\triangleright$ sample the topic index

if $u < p[k]/p[K]$ **then**

$z_i \leftarrow k$

stop

end if

end for

increment $n_{o,k}, n_{w,k}, n_k$ $\triangleright$ update the counters

end for

end for

TABLE 1
Average Object Recognition Performance of Local-LDA in Batch Experiments for Different Parameter Values

Parameters	VS (m)			IW (bins)		SL (m)				V (visual words)					K (topics)		
Values	0.01	0.02	0.03	4	8	0.02	0.03	0.04	0.05	50	60	70	80	90	30	40	50
Avg. Accuracy(%)	81	82	86	84	83	78	81	83	84	82	82	83	85	85	88	85	84

6 EXPERIMENTAL EVALUATION

Three types of experiments were carried out to evaluate the proposed approach. In this section, we first describe our experimental setup and then discuss the obtained results.

6.1 Datasets and Baselines for Comparison

The experimental evaluations were carried out on two 3D object datasets, namely the Restaurant Object dataset [36] and the Washington RGB-D Object Dataset [43]. For the off-line evaluation (i.e., 10-fold cross-validation), we mainly use the Restaurant Object dataset since it has a small number of classes (10 categories) with the significant intra-class variation that is suitable for performing extensive sets of experiments. The Washington RGB-D Object Dataset is used for open-ended evaluation since it is a large scale dataset concerning the number of images. It consists of 250,000 views of 300 common household objects taken from multiple views and organized into 51 categories. We also report on a real-world demonstration using Washington RGB-D Scenes dataset v2 [49]. This is a suitable dataset for this test since the scenes have been captured under different levels of occlusion and contain objects with similar shapes and objects with very different shapes.

For comparison, we have selected three 3D object category learning and recognition approaches, including the RACE approach [50], a BoW approach [15] based on the nearest neighbour classification rule, and the standard smoothed LDA approach [7]. For the proposed approach and for the three selected baseline approaches, batch and incremental/open-ended implementations were developed. All tests were performed with an i7, 2.40 GHz processor, and 16 GB RAM.

6.2 Offline Evaluation

The proposed approach has five parameters that must be well selected to provide a good balance between recognition performance and memory usage. To examine the performance of different configurations of the proposed approach, 10-fold cross-validation has been used. A set of 360 experiments was performed for different values of five parameters of the system, namely the voxel size (VS), which determines the number of keypoints extracted from each object view, the image width (IW) and support length (SL) of spin images, the dictionary size (V) and the number of topics (K). Three parameters including IW, V and K have a different influence on both memory usage and recognition performance. IW defines the size of the spin-image descriptor, which will be $(IW + 1) \times (2 \times IW + 1)$ float (4 bytes). The dictionary size (V) determines the number of words in the dictionary. K is the number of topics. We tried to find a good value for these parameters to obtain a good balance between recognition performance, memory usage, and processing speed. A summary of all experiments is reported in Table 1. The parameters that obtained the best average accuracy was selected as the

default configuration: VS = 0.03, IW = 4 and SL = 0.05, V = 90 and K = 30. As mentioned before, we assumed symmetric Dirichlet priors; therefore, a high alpha-value means that each object is likely to contain a mixture of most of the topics, and not any single topic specifically. Likewise, a low beta-value means that a topic may contain a mixture of just a few of the words (see Fig. 6). Griffiths and Steyvers [45] suggest a value of $50/K$ for alpha and 0.1 for the beta. Apart from this suggestion, we ran several experiments and concluded that 30 iterations is a proper choice for the number of iterations for Gibbs sampling parameter and α and β parameters should be set to 1.0 and 0.1 respectively. The accuracy of the proposed system in topic layer with the default configuration was 0.91. Therefore, this configuration displays a good balance between recognition performance and memory usage. The remaining results were obtained using this configuration. The computation time for a complete experiment (including both learning and recognition phases) on average was around 348.50 seconds. Examples of object-topic representations for eight categories of Restaurant Object Dataset [36] are depicted in Fig. 7.

Another set of experiments was carried out to evaluate the performance of LDA. Similar to the Local-LDA, a set of

Fig. 7. Examples of object-topic representation for eight categories of Restaurant Object dataset [36]: each object view is represented as a distribution over topics; each topic is shown with its top frequent visual words (i.e., visual words with more than 25 percent probability) using a unique color. It should be note, colours have different meaning in each category.

TABLE 2
Average Object Recognition Performance of LDA in Batch Experiments for Different Parameters Values

Parameters	VS (m)			IW (bins)		SL (m)				DS (dictionary size)					K (topics)		
Values	0.01	0.02	0.03	4	8	0.02	0.03	0.04	0.05	50	60	70	80	90	30	40	50
Avg. Accuracy	0.78	0.78	0.75	0.79	0.79	0.64	0.73	0.78	0.81	0.81	0.82	0.82	0.82	0.85	0.84	0.84	0.81

TABLE 3
Average Object Recognition Performance of the RACE Approach in Batch Experiments for Different Parameter Values

Parameters	VS (m)			IW (bins)		SL (m)				CT		
Values	0.01	0.02	0.03	4	8	0.02	0.03	0.04	0.05	1.00	1.50	2.00
Avg. Accuracy	0.34	0.40	0.41	0.43	0.34	0.22	0.29	0.46	0.56	0.35	0.39	0.40

Parameters	VS (m)			IW (bins)		SL (m)				CT		
Values	0.03	0.04	0.05	4	8	0.07	0.08	0.09	0.10	1.00	1.50	2.00
Avg. Accuracy	0.75	0.76	0.74	0.72	0.78	0.71	0.75	0.77	0.78	0.65	0.80	0.81

TABLE 4
Average Object Recognition Performance of the BoW Approach in Batch Experiments for Different Parameter Values

Parameters	VS (m)			IW (bins)		SL (m)				V (visual words)				
Values	0.01	0.02	0.03	4	8	0.02	0.03	0.04	0.05	50	60	70	80	90
Avg. Accuracy	0.83	0.74	0.65	0.74	0.74	0.58	0.71	0.81	0.85	0.74	0.74	0.74	0.73	0.73

360 experiments was performed for different values of the system parameters. For all experiments, we also set α and β and the number of Gibbs sampling iterations to 1.0 and 0.1 and 30 respectively. The obtained results are summarized in Table 2 which shows the average accuracies obtained with different configurations. Similar to the previous approach, the parameter configuration that obtained the best average accuracy and leads to better efficiency was selected as the default configuration: VS = 0.02, IW = 4, and SL = 0.05, DS = 90, and K = 30. The performance of the topic layer with LDA (i.e., shared topics among all categories) was carried out using this configuration and its accuracy was 0.88.

In the case of RACE approach [50], a set of 72 experiments was performed for different values of the four system parameters including VS, IW, SL and classification threshold (CT). Results are presented in Table 3 (*top row*). One important observation is that by increasing the value of SL parameter, the classification performance is also increased. It is due to this fact that support length parameter determines the amount of cylindrical space swept out by a spin-image, which will have a radius of SL and height of $2 \times SL$. Therefore, selecting a large value to the SL parameter causes spin image behaves like a global descriptor, which is out of the scope of this work. To prove this observation, we ran another set of experiments. Results are summarized in Table 3 (*bottom row*). In these experiments the following parameters obtained the best average accuracy and set as the default system parameters: VS = 0.04, IW = 8, SL = 0.10 and CT = 2. The accuracy of the proposed system with the default parameters was 0.87. Results show that the overall performance of the recognition system is promising. Since we investigate object category learning and recognition using topic modelling representation, global features (i.e., spin-image with large support length parameter) are not suitable. This is the reason why we preferred to use the small value for the SL parameter. An entire experiment for the RACE approach in average took around 1757.20 seconds.

The descriptiveness of the BoW approach [15] was also evaluated. Towards this end, a set of 120 experiments has been performed. Experiments were repeated for different values of the four parameters of the system including (VS, IW, SL, V). Results are summarized in Table 4. The default system configuration for this approach: VS = 0.01, IW = 4, SL = 0.05 and DS = 70. The accuracy of the system with the default configuration was 89 percent. The average computation time for an experiment was around 195.60 seconds.

The RACE, BoW and LDA approaches obtained acceptable recognition performances (i.e., 0.87, 0.88 and 0.89 respectively) and the proposed Local-LDA obtained 0.91 accuracy rate. The Local-LDA provides a good balance between memory usage and descriptiveness with 30 floats (i.e., NT = 30). The length of the BoW is around three times larger than the representation of the Local-LDA. The RACE is the less compact representation. One important observation is that the overall performance of the recognition system based on topic modelling is promising and the proposed Local-LDA representation is capable of providing distinctive representation for the given object. In addition, independent topics for each category provides better representation than shared topics for all categories. Furthermore, it is observed that the discriminative power of LDA (shared topics) [7] depends on the order of introduction of categories.

In terms of experiment time, the BoW approach [15] achieved the best performance that is around 1.78 and 8.98 times smaller than the Local-LDA and RACE [50] approaches respectively. The underlying reason is that there is a Gibbs sampling procedure in the LDA-based approaches which makes the difference. It can be concluded that the RACE approach [50] is computationally expensive. The best results for each approach are reported in Table 5.

6.3 Open-Ended Evaluation

Off-line evaluation methodologies (e.g., k-fold cross validation, leave-one-out, etc.) do not comply with the simultaneous nature of learning and recognition in autonomous

TABLE 5
Object Recognition Performance in Batch Experiments for the Best Parameter Configuration in Each Approach

Approach	Accuracy	Exp. Time (s)
RACE [50]	0.87	1757.20
BoW [15]	0.89	195.60
LDA (shared topics)[7]	0.88	227.50
Local-LDA (our approach)	0.91	348.50

robots. Moreover, they assume that the number of categories is predefined. An evaluation protocol for open-ended learning systems was proposed in [47], [51], [52]. The idea is to emulate the interactions of a learning and recognition system with the surrounding environment over long periods of time. This protocol can be followed by a human teacher performing the *teach*, *ask* and *correct* actions described in Section 5. However, replacing a human teacher with a simulated one makes it possible to conduct systematic, consistent and reproducible experiments for different approaches. Having a simulated teacher allows the possibility to perform multiple experiments and explore different experimental conditions in a fraction of time a human would take to carry out the same task. A *simulated teacher* was therefore developed to follow the teaching protocol and autonomously interact with an agent (i.e., a learning and recognition system).

The idea is that, for each newly taught category, the simulated teacher repeatedly picks unseen object views of the currently known categories from a dataset and presents them to the system. It progressively estimates the recognition accuracy of the system and, in case this accuracy exceeds a given threshold ($\tau = 0.67$, meaning accuracy is at least twice the error rate), introduces an additional object category. In case the agent can not reach the classification threshold after a certain number of iterations, the teacher infers that the agent is not able to learn more categories and terminates the experiment (breakpoint). This way, the system is trained, and at the same time the accuracy of the system is continuously estimated. The simulated teacher is connected to an object dataset. In this work, the simulated teacher was connected to the Washington RGB-D Object Dataset consisting of 250,000 views of 300 common household objects, organized into 51 categories [43].

When an experiment is carried out, learning performance is evaluated using several measures, including: (i) number of learned categories at the end of an experiment (#LC), an indicator of *how much the system is capable of learning*; (ii) the number of question / correction iterations (#QCI) required to learn those categories and the average number of stored instances per category (AIC), indicators of *time and memory resources required for learning*; (iii) Global Classification Accuracy (GCA), an accuracy computed using all predictions in a complete open-ended learning experiment, and the Average Protocol Accuracy (APA), an average of all accuracy values used to control the application of the teaching protocol in an open-ended learning experiment. GCA and APA are indicators of *how well the system learns*.

Similar to the previous evaluation, in addition to the proposed approach, we have used the RACE approach [50], a Bag-of-Words (BoW) approach [15], and the standard smoothed LDA approach [7], modified to be open-ended. The RACE and BoW approaches completely discard structural

information, i.e., information related to the co-occurrence of local object features. The LDA-based approaches capture structural information and were therefore included to assess the added-value, of structural information. Since the order introduction of the categories may have an effect on the performance of the system, 10 experiments for each approach were carried out in which categories were introduced in random sequences. It should be noted that the random sequences were the same for all approaches and the default parameters in the respective implementations were used. Therefore, a set of 40 experiments was performed to evaluate the performance of the selected approaches, concerning their scalability with respect to increasing number of categories. Results are reported in Fig. 8.

The left column of Fig. 8 provides a detailed summary of the obtained results. By comparing all approaches, it is visible that the agent, on average, learned more categories (#LC) using the Local-LDA approach than using other approaches. Although, RACE and BoW approaches stored fewer instances per category (AIC), on average, than Local-LDA, their discriminative power is lower and their performance dropped quickly as the number of categories increased (see the right column of Fig. 8). It was also observed that the discriminative power of shared topics depends a lot on the order of introduction of categories.

From Fig. 8, we also see that on average the longest experiments were observed with Local-LDA and the shortest ones were observed with LDA. In the case of LDA, the agent on average learned 14.40 categories using 262.60 question/correction iterations. Local-LDA on average continued for 613 question/correction iterations and the agent was able to learn 28.50 categories. By comparing all the experiments based on Local-LDA, it is visible that in the first experiment, the number of iterations required to learn 32 object categories was greater than in other experiments. The maximum number of categories that the agent could learn with Local-LDA was 32 categories (i.e., Exp. #1 and #4). These experiments respectively took 811 and 696 question/correction iterations for the agent to acquire that number of categories. It can be observed that, on average, Global Classification Accuracy (GCA) of Local-LDA is higher than the other approaches.

The center column of the Fig. 8 illustrates how fast the learning occurred in each of the experiments. It shows the number of question/correction iterations required to learn a certain number of categories. The right column of Fig. 8 shows the global classification accuracy obtained by the selected approaches as a function of the number of learned categories. One important observation is that accuracy decreases in all approaches as more categories are introduced. This is expected since a higher number of categories known by the system tends to make the classification task more difficult. BoW and Local-LDA achieved the best accuracies with stable performance. It is also clear that BoW achieved better APA than the other approaches. This is expected since BoW learned fewer categories, and it is easier to get better APA in fewer categories. By comparing all experiments, it is visible that the agent learned more categories when using Local-LDA. In particular, it learned around 7 categories more than BoW approach and nine and 14 categories more than RACE and LDA approaches respectively. Local-LDA provides an appropriate balance between all parameters.

EXP#	#QCI	#LC	AIC	GCA (%)	APA (%)
1	223	12	10.33	0.60	0.74
2	347	20	8.40	0.68	0.81
3	183	12	8.83	0.61	0.73
4	852	35	10.09	0.70	0.78
5	646	30	9.70	0.69	0.73
6	454	26	7.42	0.74	0.82
7	194	13	7.69	0.68	0.79
8	269	17	8.23	0.66	0.77
9	414	20	9.55	0.68	0.80
10	239	14	8.64	0.66	0.79
Avg.	382.10	19.90	8.88	0.67	0.78

(a) Summary of experiments using RACE approach

EXP#	#QCI	#LC	AIC	GCA (%)	APA (%)
1	320	18	8.44	0.69	0.82
2	384	23	7.26	0.74	0.84
3	381	21	8.42	0.70	0.78
4	519	27	8.29	0.72	0.82
5	941	33	11.27	0.70	0.79
6	198	14	8.14	0.63	0.78
7	326	21	7.38	0.71	0.82
8	279	19	6.89	0.73	0.84
9	505	24	8.91	0.71	0.83
10	265	18	7.00	0.72	0.84
Avg.	411.80	21.80	8.20	0.71	0.82

(b) Summary of experiments using BoW approach

EXP#	#QCI	#LC	AIC	GCA (%)	APA (%)
1	184	10	10.60	0.58	0.76
2	190	14	6.71	0.72	0.85
3	149	7	11.14	0.61	0.76
4	437	20	9.85	0.68	0.82
5	411	22	8.68	0.69	0.78
6	127	9	7.44	0.68	0.85
7	289	19	7.89	0.68	0.83
8	383	17	10.76	0.65	0.76
9	263	15	9.06	0.65	0.73
10	193	11	9.27	0.64	0.82
Avg.	262.60	14.40	9.14	0.66	0.80

(c) Summary of experiments using Open-Ended LDA approach

EXP#	#QCI	#LC	AIC	GCA (%)	APA (%)
1	811	32	10.19	0.71	0.80
2	654	30	9.40	0.70	0.77
3	514	28	8.07	0.72	0.80
4	696	32	8.59	0.74	0.82
5	579	27	9.14	0.71	0.77
6	505	24	8.91	0.71	0.83
7	697	30	9.40	0.72	0.80
8	550	25	9.92	0.68	0.77
9	560	29	8.31	0.72	0.80
10	564	28	8.92	0.70	0.79
Avg.	613.00	28.50	9.08	0.71	0.80

(d) Summary of experiments using Local-LDA approach

Fig. 8. Summary of system performance in open-ended evaluations using Washington RGB-D object dataset [43].

From open-ended evaluations, we conclude that all the selected approaches seem to be able to generalize to a certain number of categories, while the classification accuracy drops by increasing the number of categories (Fig. 8 right). Based on the obtained results, we performed another set of experiments to assess the classification performance of each approach by training for all categories at the very beginning (Train-then-Test schema), instead of using the teaching protocol (Test-then-Train schema). These experiments show the difference between the adopted open-ended learning approach, where categories are gradually introduced and

only distinctive instances are stored, and the batch approach, where all the instances of all categories are available from the beginning. Note that, since the approach in this paper is instance-based, i.e., categories are represented by sets of instances, the batch approach is limited to storing all available instances. Towards this end, we have performed a set of batch experiments for each approach based on the *average number of learned categories* and the *average number of instances per category* as given by the respective open-ended experiments. For example, if an approach at the end of open-ended experiments on average learned x categories with y instances per

TABLE 6
Comparison between the Open-Ended Experiments
and Batch Experiments with the Same Number of
Categories and Instances

Approach	Average #LC	Average AIC	Last Protocol Accuracy	Batch Accuracy
RACE [50]	20	9	0.59	0.34
BoW [15]	22	8	0.66	0.70
LDA	14	9	0.59	0.56
Local-LDA	29	9	0.65	0.61

category, we setup the batch experiments by calculating the round of x and y as x' and y' . Then, for that approach, a set of 10 experiments is carried out. In each experiment, we train the agent by randomly selecting x' categories and y' instances per category and test it using an unseen instance per category. Results are summarized in Table 6.

By comparing the accuracy of batch learning experiments with the teaching protocol accuracy (PA) at breakpoint in the open-ended learning experiments, the benefit/penalty of the adopted incremental learning approach is highlighted. In the case of RACE there is a significant difference. LDA and Local-LDA show, it makes sense that the adopted incremental learning approach works slightly better for the same number of stored instances, because the incremental learning scheme stores distinctive instances, whereas the simple batch approach stores all instances, which may be redundant and not represent all variations in the data. In the case of BoW, experiments show 4 percent penalty for using the incremental approach instead of learning all classes at once.

6.4 System Demonstration

We performed three demonstrations to show the strength of the proposed approach concerning real-time performance, scalability, robustness to occlusion, and ability to cope with context change.

6.4.1 Real-World Scenario

To show the strength of object representation, a real demonstration was performed, in which the proposed approach has been integrated in the object perception system presented in [36], [50]. In this demonstration a table is in front of a camera and two users interact with the system (see Fig. 9 left). Initially, the system only had prior knowledge about the *Vase* and *Dish* categories, learned from batch data (i.e., set of observations with ground truth labels), and there is no information about other categories (i.e., *Mug*, *Bottle*, *Spoon*). Throughout this session, the system must be able to recognize instances of learned categories and incrementally learn new object categories. Note, in the case of training online, the instructor can be a human user, and in the case of batch learning we used a set of ground truth labels. The right side of Fig. 9 illustrates the behaviour of the system:

- Fig. 9 (right-top): the instructor puts object TID6 (a *Mug*) on the table. It is classified as *Unknown* because mugs are not known to the system; Instructor labels TID6 as a *Mug*. The system conceptualizes *Mug* and TID6 is correctly recognized. The instructor places a

Fig. 9. Real demonstration: (left) System setup; (right) two snapshots show the proposed system supports (top) classical learning from a batch of labelled training data and (bottom) open-ended learning from on-line experiences.

Vase on the table. The system has learned *Vase* category from batch data, therefore, the *Vase* is properly recognized

- Fig. 9 (right-bottom): later, another *Mug* is placed on the table. This particular *Mug* had not been previously seen, but the system can recognize it, because the *Mug* category was previously taught.

In this demonstration for realizing that the target object belongs to an “unknown” category, the robot reasons in terms of distances to the known categories. We used a simple classification threshold (CT) on the $OCD(.)$ to predict an “unknown” category when the target object is too far from all known categories. In particular, if, for all categories, the $OCD(.)$ is larger than a given CT (e.g., 0.75), then the object is classified as *unknown*; If we change CT from 0.75 to 0.5, the system being under-confident means that the target object should be very similar to the instances of known categories to be recognized (and vice versa). This simple rule can be easily replaced by a more advanced unknown/outlier detection schemes. For example, it is possible to consider the ratio of OCD distances between the closest and second closest category. It is worth to mention that only in this demonstration we considered unknown category detection mechanism. This demonstration shows that the system is capable of using prior knowledge to recognize new objects in the scene and learn about new object categories in an open-ended fashion. A video of this demonstration is available at: https://youtu.be/J0QOc_Ide4.

6.4.2 Scene Dataset

Another demonstration has been performed using the Washington RGB-D Scenes dataset v2 [49]. This dataset consists of 14 scenes containing a subset of the objects in the Washington RGB-D Object Dataset, including *Bowls*, *Caps*, *Mugs*, and *SodaCans* and *CerealBoxes*. Initially, the system had no prior knowledge. The four first objects are introduced to the system using the first scene and the system conceptualizes those categories. The system is then tested using the second scene of the dataset and it can recognize all objects except cereal boxes, because this category was not previously taught. The instructor provided corrective feedback and the

Fig. 10. Four snapshots showing object recognition results on eight scenes of Washington scene dataset [49].

Fig. 11. (a) Experimental setup: First, we train the robot using the objects' view extracted from the scene captured from the blue cameras. We then test the system using the remaining ten scenes captured from the red cameras. Three snapshots (b,c,d) showing the system performance during the demonstration: (b) An example of object recognition result for one of the test scene. (c) Later, we move the system to a new context; each scene in this context contains four instances of two object categories with very similar shapes (lipton versus softkings). The robot could recognize all objects correctly by using knowledge learned from previous contexts. (d) To show the robustness of the proposed approach against occlusion, we move the system to another context which is much more crowded than the previous one. The robot could recognize all objects correctly. This shows that the proposed approach is capable of recognizing objects under various level of occlusion in different positions.

system conceptualized the cereal boxes category. Afterwards, all objects are classified correctly in all 12 remaining scenes. Four snapshots are shown in Fig. 10. This evaluation illustrates the process of acquiring categories in an open-ended fashion. A video of this demonstration is online at: <https://youtu.be/pe29DYNolBE>.

6.4.3 Different Camera Poses and Context Change

We report on another demonstration using the Imperial College Domestic Environment Dataset [53]. This is a suitable dataset for this test since all scenes were captured under various level of occlusion and contain several objects. Similar to the previous demonstration, the system initially had no prior knowledge, and all objects are recognized as unknown. Later, a user interacts with the system in an online manner and teaches all object categories including *Amita*, *Colgate*, *Lipton*, *Elite*, *Oreo* and *Softkings* to the system using the objects' views extracted from scenes captured from the blue cameras as shown in Fig. 11a. The system conceptualizes all categories using the extracted object views. Afterwards, the system is tested by the remaining ten scenes captured from different viewpoints (i.e., shown by red cameras). The system could recognize all objects properly by using the knowledge learned from the three training scenes. Some misclassification also occurred throughout the demonstration. The underlying reason was that, at some points, the object tracking could not track the object accurately and the distinctive parts of the object were not included in the object's point cloud. An example of the obtained results is depicted in Fig. 11b.

Later, we moved the system to two new contexts to validate its knowledge using new scenes. The first context contains four instances of two object categories with very similar shapes (*lipton* versus *softkings*). The system could recognize all objects properly (see Fig. 11c). This shows the

descriptive power of the proposed approach. Then, the system is moved to the second context. This context is much more crowded than the previous one. In this case, eight instances of the four categories exist. The robot could recognize all objects correctly by using knowledge from the previous context (see Fig. 11d). This demonstration shows the robustness of the proposed approach against occlusion. A video of this demonstration is available online at: <https://youtu.be/1ONrAKcA79c>.

7 CONCLUSION

This paper presented a multi-layered object representation approach to enhance an open-ended 3D object category learning and recognition system developed for robotic applications. In this work, for optimizing the recognition process and memory usage, each object view was hierarchically described as a random mixture over a set of latent topics, and each topic was defined as a discrete distribution over visual words. This paper focused on open-ended object category learning and recognition using an extension of topic modelling. We transform objects from a bag-of-words space into a local semantic space for providing a powerful representation to deal with the semantic gap between low-level features and high-level concepts.

A set of extensive experiments has been performed to assess the performance of the proposed Local-LDA and compare its performance with other state-of-art approaches with respect to several characteristics including descriptiveness, scalability and experiment time. Experimental results show that the overall classification performance obtained with Local-LDA is comparable to the best performances obtained with the state-of-the-art approaches. Local-LDA outperformed the selected state-of-the-art approaches (i.e., RACE [50], BoW [15]

and LDA [7]), achieving appropriate descriptiveness and scalability with respect to increasing number of categories. Regarding experiment time, the best performance was obtained with BoW [15]. The Local-LDA and RACE [50] approaches led to experiment times 1.78 to 8.98 times higher than BoW [15] respectively. To show all the described properties of the proposed representation, Local-LDA, three real demonstrations were carried out. Results showed that the proposed system supports classical learning from a batch of labelled training data and open-ended learning from online experiences of a robot. In the continuation of this work, we would like to study how to use LDA approaches in fine-grained object recognition scenarios [54] and investigate the possibility of overcoming the mentioned limitations to use deep learning approaches in open-ended domains [55].

ACKNOWLEDGMENTS

This work was partially funded by Portuguese funds through FCT project PEst-OE/EEI/UI0127/2016 and FCT scholarship SFRH/BD/94183/2013.

REFERENCES

- [1] L. Seabra Lopes and Q. Wang, "Towards grounded human-robot communication," in *Proc. 11th IEEE Int. Workshop Robot Human Interactive Commun.*, 2002, pp. 312–318.
- [2] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature Neuroscience*, vol. 2, no. 11, pp. 1019–1025, 1999.
- [3] S. Jeong and M. Lee, "Adaptive object recognition model using incremental feature representation and hierarchical classification," *Neural Netw.*, vol. 25, pp. 130–140, 2012.
- [4] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 356–369.
- [5] M. Schwarz, H. Schulz, and S. Behnke, "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 1329–1335.
- [6] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3367–3375.
- [7] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.
- [8] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.
- [9] S. H. Kasaei, A. M. Tomé, and L. Seabra Lopes, "Hierarchical object representation for open-ended object category learning and recognition," in *Proc. Advances Neural Inf. Process. Syst.*, 2016, pp. 1948–1956.
- [10] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 3212–3217.
- [11] L. Bo, K. Lai, X. Ren, and D. Fox, "Object recognition with hierarchical kernel descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1729–1736.
- [12] S. H. Kasaei, A. M. Tomé, L. Seabra Lopes, and M. Oliveira, "GOOD: A global orthographic object descriptor for 3D object recognition and manipulation," *Pattern Recognit. Lett.*, vol. 83, pp. 312–320, 2016.
- [13] S. H. Kasaei, L. Seabra Lopes, A. M. Tomé, and M. Oliveira, "An orthographic descriptor for 3D object learning and recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 4158–4163.
- [14] A. Aldoma, Z. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. B. Rusu, S. Gedikli, and M. Vincze, "Tutorial: Point cloud library: Three-dimensional object recognition and 6 DOF pose estimation," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 80–91, Sep. 2012, doi: 10.1109/MRA.2012.2206675.
- [15] H. Kasaei, and M. Oliveira, "An adaptive object perception system based on environment exploration and Bayesian learning," in *Proc. IEEE Int. Conf. Auton. Robot Syst. Competitions*, 2015, pp. 221–226.
- [16] Y. Li, S. Pirk, H. Su, C. R. Qi, and L. J. Guibas, "FPNN: Field probing neural networks for 3D data," *Advances in Neural Inf. Process. Syst.*, 29, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds., Curran Associates, Inc., pp. 307–315, 2016.
- [17] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D shapenets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1912–1920.
- [18] J. Sivic, B. C. Russell, A. Efros, A. Zisserman, W. T. Freeman, et al., "Discovering objects and their location in images," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, 2005, pp. 370–377.
- [19] T. Hofmann, "Probabilistic latent semantic indexing," in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 1999, pp. 50–57.
- [20] D. M. Blei, "Probabilistic topic models," *Commun. ACM*, vol. 55, no. 4, pp. 77–84, 2012.
- [21] F. Rodrigues, B. Ribeiro, M. Lourenço, and F. Pereira, "Learning supervised topic models from crowds," in *Proc. Third AAAI Conf. Human Computation and Crowd Sourcing*, 2015, pp. 160–168.
- [22] J. D. McAuliffe and D. M. Blei, "Supervised topic models," in *Proc. Advances Neural Inf. Process. Syst.*, 2008, pp. 121–128.
- [23] C. Wang, D. Blei, and F.-F. Li, "Simultaneous image classification and annotation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1903–1910.
- [24] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 524–531.
- [25] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2009, pp. 248–256.
- [26] Y. Wang, P. Sabzmejdani, and G. Mori, "Semi-latent Dirichlet allocation: A hierarchical model for human action recognition," in *Human Motion–Understanding, Modeling, Capture and Animation*. Berlin, Germany: Springer, 2007, pp. 240–254.
- [27] M. Hoffman, F. R. Bach, and D. M. Blei, "Online learning for latent Dirichlet allocation," in *Proc. Advances Neural Inf. Process. Syst.*, 2010, pp. 856–864.
- [28] H. Celikkat, G. Orhan, N. Pugeault, F. Guerin, E. Şahin, and S. Kalkan, "Learning context on a humanoid robot using incremental latent Dirichlet allocation," *IEEE Trans. Cognitive Develop. Syst.*, vol. 8, no. 1, pp. 42–59, Mar. 2016.
- [29] A. Banerjee and S. Basu, "Topic models over text streams: A study of batch and online unsupervised learning," in *Proc. SIAM Int. Conf. Data Mining*, 2007, pp. 431–436.
- [30] K. R. Canini, L. Shi, and T. L. Griffiths, "Online inference of topics with latent Dirichlet allocation," in *Proc. Int. Conf. Artif. Intell. Statistics*, 2009, pp. 65–72.
- [31] S. Gao, I. W.-H. Tsang, and Y. Ma, "Learning category-specific dictionary and shared dictionary for fine-grained image categorization," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 623–634, Feb. 2014.
- [32] Y. Zhang, X.-S. Wei, J. Wu, J. Cai, J. Lu, V.-A. Nguyen, and M. N. Do, "Weakly supervised fine-grained categorization with part-based image representation," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1713–1725, Apr. 2016.
- [33] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.
- [34] J. Sock, S. Kasaei, L. Seabra Lopes, and T.-K. Kim, "Multi-view 6D object pose estimation and camera motion planning using RGBD images," in *Proc. Int. Conf. Comput. Vis. 3rd Int. Workshop Recovering 6D Object Pose*, 2017, pp. 2228–2235.
- [35] S. H. Kasaei, J. Sock, L. Seabra Lopes, A. M. Tomé, and T.-K. Kim, "Perceiving, learning, and recognizing 3D objects: An approach to cognitive service robots," in *Proc. 32nd Conf. Artif. Intell.*, 2018, pp. 596–602.
- [36] S. H. Kasaei, M. Oliveira, G. H. Lim, L. Seabra Lopes, and A. M. Tomé, "Interactive open-ended learning for 3D object recognition: An approach and experiments," *J. Intell. Robotic Syst.*, vol. 80, no. 3, pp. 537–553.
- [37] D. Katz, M. Kazemi, J. A. Bagnell, and A. Stentz, "Clearing a pile of unknown objects using interactive perception," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 154–161.
- [38] H. Van Hoof, O. Kroemer, and J. Peters, "Probabilistic interactive segmentation for anthropomorphic robots in cluttered environments," in *Proc. 13th IEEE-RAS Int. Conf. Humanoid Robots*, 2013, pp. 169–176.

- [39] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 1096–1103.
- [40] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. Dec, pp. 3371–3408, 2010.
- [41] S. H. Kasaei, M. Oliveira, G. H. Lim, L. Seabra Lopes, and A. M. Tomé, "Towards lifelong assistive robotics: A tight coupling between object perception and manipulation," *Neurocomputing*, vol. 291, pp. 151–166, 2018.
- [42] M. Oliveira, L. Seabra Lopes, G. H. Lim, S. H. Kasaei, A. D. Sappa, and A. M. Tomé, "Concurrent learning of visual codebooks and object categories in open-ended domains," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 2488–2495.
- [43] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 1817–1824.
- [44] J. G. Kim, I. Biederman, M. D. Lescroart, and K. J. Hayworth, "Adaptation to objects in the lateral occipital complex (LOC): Shape or semantics?" *Vis. Res.*, vol. 49, no. 18, pp. 2297–2305, 2009.
- [45] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proc. National Academy Sci. United States America*, vol. 101, no. suppl 1, pp. 5228–5235, 2004.
- [46] I. Porteous, et al., "Fast collapsed gibbs sampling for latent Dirichlet allocation," in *Proc. 14th Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 569–577.
- [47] L. Seabra Lopes and A. Chauhan, "How many words can my robot learn?: An approach and experiments with one-class learning," *Interaction Stud.*, vol. 8, no. 1, pp. 53–81, 2007.
- [48] L. Steels and F. Kaplan, "Aibo's first words: The social learning of language and meaning," *Evolution Commun.*, vol. 4, no. 1, pp. 3–32, 2000.
- [49] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3D scene labeling," in *Proc. Int. Conf. Robot. Autom.*, 2014, pp. 3050–3057.
- [50] M. Oliveira, L. Seabra Lopes, G. H. Lim, H. Kasaei, A. M. Tomé, and A. Chauhan, "3D object perception and perceptual learning in the RACE project," *Robot. Autonomous Syst.*, vol. 75, Part B, pp. 614–626, 2016.
- [51] A. Chauhan and L. S. Lopes, "Using spoken words to guide open-ended category formation," *Cognitive Process.*, vol. 12, no. 4, pp. 341–354, 2011.
- [52] S. H. Kasaei, L. Seabra Lopes, and A. M. Tomé, "Coping with context change in open-ended object recognition without explicit context information," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1–7.
- [53] A. Doumanoglou, R. Kouskouridas, S. Malassiotis, and T.-K. Kim, "Recovering 6D object pose and predicting next-best-view in the crowd," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3583–3592.
- [54] S. H. Kasaei, "Look further to recognize better: Learning shared topics and category-specific dictionaries for open-ended 3D object recognition," in *Proc. 2019 IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, 2019, pp. 1–6.
- [55] S. H. Kasaei, "OrthographicNet: A deep learning approach for 3D object recognition in open-ended domains," *arXiv preprint arXiv:1902.03057*, 2019.

S. Hamidreza Kasaei is a Faculty of Science and Engineering (FSE) fellow at the Department of Artificial Intelligence, University of Groningen, The Netherlands. His main research interests includes the area of 3D object perception, grasp affordance, and object manipulation. He is interested in developing algorithms for an adaptive perception system based on interactive environment exploration and open-ended learning, which enables robots to learn from past experiences and interact with human users.

Luís Seabra Lopes received the PhD degree in robotics and integrated manufacturing from the New University of Lisbon, Portugal, in 1998. He is associate professor of Informatics with the Department of Electronics, Telecommunications and Informatics of the University of Aveiro, Portugal. He has long standing interests in robot learning, cognitive robotic architectures, and human-robot interaction.

Ana Maria Tomé is an associate professor of electrical engineering with the DET/IEETA of the University of Aveiro. Her research interests include digital and statistical signal processing, independent component analysis, and blind source separation, as well as machine learning applications.

▷ For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.