Part II

IMAGE AND HANDWRITING

Can we perform character extraction accurately and robustly?
BINARIZATION TECHNIQUES

Following the initial writer identification tests in Chapter 2, this chapter explores a new binarization technique. Due to the severe degradation and damage of the Dead Sea Scrolls (DSS), the intensity-based existing binarization techniques do not work well. Even though many neural network-based approaches may learn the complexity of foreground-background separation in these documents, the problem of acquiring training data remains. This chapter thus tackles both the scarcity of training data and the complexity of learning clear ink-background separation. A new neural network architecture based on the U-Net models is proposed especially for the DSS collection, which proved effective for many other historical manuscripts. Image fusion techniques for multi-spectral images are also explored to achieve better results. Finally, a robust end-to-end model is proposed, and binarization results are presented for the complete collection of the DSS. This chapter’s outcome allows further testing on writer identification and dating in the latter chapters of this thesis. The work also allows image-based material analysis and character-based augmentation of the DSS.

PROVENANCE
This chapter is an adaptation of an article that has been previously published as:

AUTHOR CONTRIBUTIONS
Dhali conceived the complete setup, implemented the models, performed the experiments, performed the analysis, wrote the draft with other co-authors, and wrote the final manuscript.
ABSTRACT Handwritten document-image binarization is a semantic segmentation process to differentiate ink pixels from background pixels of the surface material. It is an essential step toward several handwriting analysis tasks, including character recognition, writer identification, and development estimation of script styles. The binarization task is challenging due to the vast diversity of writing styles among people, writing surfaces, and inks used. It is even more complex for ancient historical manuscripts due to the aging and degradation of these documents over time. One such manuscript collection is the famous Dead Sea Scrolls (DSS), which poses extreme challenges for the existing binarization techniques. This article presents a novel binarization pipeline for DSS images using deep encoder-decoder networks. Although the artificial neural network presented here is primarily designed to binarize the DSS images, it can also be used on many other manuscript collections. Additionally, transfer learning shows the usefulness of the network for a wide range of handwritten documents, making it a unique multi-purpose tool for binarization. Qualitative results and several quantitative comparisons using historical manuscripts and datasets from handwritten document image binarization competitions (H-DIBCO and DIBCO) exhibit the system’s robustness and effectiveness. The best-performing network architecture proposed here, BiNet-derived from the U-Net encoder-decoders, outperforms existing methods by achieving an F-score of 86.7% (±9.4) on fused images created using multi-spectral photos of the DSS collection.

3.1 INTRODUCTION

In a digitized image of a handwritten document, the ink-based pixels result from a physical ink deposition process, where a surface material absorbs the pigment creating the foreground. In contrast, the original unaffected material texture appears as the background. In a typical handwritten document,

\[ N_{\text{ink}} << N_{\text{background}} \]

where, \( N_{\text{ink}} \) is the number of ink pixels and \( N_{\text{background}} \) is the number of background pixels.

The binarization process of handwritten document-image allocates a binary value to each image pixel [40]; 0 for ink and 1 for not-ink. Thus, this process separates the foreground (meaningful information, in general, the ink) from the background (the surface material). The binarized images are compressed and pose significant importance in analyzing the document. It facilitates character recognition, segmentation, and transcription pipeline [7, 47, 172]. Further processing of the handwriting towards writer identification and script-style development also depends on the success of the binarization process itself [65, 120]. Over the years, many techniques have been proposed to perform binarization tasks. However, it is always challenging to obtain good results due to the diversity of handwritten documents. One technique may
perform well for some specific types of documents but fail for other types. The problem becomes challenging regarding historical manuscript binarization [279].

![Figure 3.1](image_url)

Figure 3.1: Three images from the Dead Sea Scrolls collection show the diversity of the materials and their current state with difficult readability due to various degradation. They are all RGB-colored images of the plates (physical arrangements of the fragmented materials on a plane surface). Plate 671-1 contains only one physical fragment, whereas the next two plates (117 and 215) contain multiple physical fragments. Fragments from plate 117 were produced from papyrus, and the repetitive patterns of the fibers are visible in the zoomed-in section of the image. The other two plates contain fragments made from parchments.

Numerous historical manuscripts collections are residing all over the world [179]. Most of them are significant, both culturally and scientifically [8]. The Dead Sea Scrolls (DSS) is one such collection. They are ancient manuscripts discovered in the mid-20th century between Jerusalem and the Dead Sea in the Judaean Desert. Most were written two thousand years ago, over a period of almost four centuries (around 250 BCE to 135 CE), and hold tremendous historical, religious, and linguistic significance [214]. The recent digitization of this collection has opened the door for pattern recognition techniques to be applied to revise existing hypotheses on the writers and dates of these scrolls [65]. However, these documents have diverse document textures and are heavily degraded (see Figure 3.1), primarily due to the materials, natural aging, the preservation processes, and the places they were kept in. Therefore, the images must be preprocessed to perform pattern recognition techniques on the original content (texts). One of the critical steps in this preprocessing is a binarization technique that can keep the actual content of these documents as intact as possible.

There are several challenges in binarizing the DSS images. Similar to many other historical manuscripts, the DSS collection profoundly suffers from document degradation problems. Due to aging and natural causes, the individual glyphs (characters) of the DSS images often show fading effects. Some of the images also show thinning of the characters along with broken (missing or completely faded) parts. Some images often suffer from uneven illumination problems due to the surface material [278]. On top of all these, the most severe issue is the low contrast between ink and background (see Figure 3.2).
3.1 Introduction

3.1.1 Why binarization is still important

Although many modern deep-learning methods in document analysis can be trained end-to-end, directly with a grayscale or color image, this is not desirable in an e-science approach for humanities studies. For instance, a direct end-to-end solution for clustering the DSS collection can be achieved for writer identification and script-style evolution. Still, there is always a risk of getting the solution for the wrong cause, as in the anecdotal story of the ‘Russian (hidden) tank problem’ [71]. The decision of a neural network may be based on spurious correlations with the texture of the background of different materials, such as papyrus and parchment. The fiber statistics of papyrus manufacturing batches may also add to the wrong cause. Additionally, irrelevant materials like the background, rice papers, number tags, scale bars, color calibrators, and other patterns must not be allowed to contribute to the process. So binarization is necessary, and it needs to be precise.

Many of the existing binarization techniques are pixel-intensity based. It is clear that any method working on the pixel intensity will only struggle to produce excellent results for the DSS images. An intelligent binarization method should accommodate all different variability in the DSS collection and still provide superior results. Semi-automatic selection of the region of interest or a manual one-off preprocessing technique may obtain good binarization. Still, it will not be a robust solution for the whole corpus. A non-biasing foreground-background separation is required for objective analysis, including writer identification and dating. Separating foreground-background can be a severe problem but significant for scholarly research. This problem can be illustrated using a suggestion put forward by the eminent palaeographer Ada Yardeni. She ascribed fifty-seven or possibly even ninety-three manuscripts to one scribe ([175, 317]). Figure 3.3 shows two such fragment images. This figure also presents the binarization results from three of the most
famous traditional techniques. In the case of ink separation, the techniques perform considerably well in some regions of the image but fail in most of the remaining areas. Now, for the possibility of writer identification, the palaeographer who is strictly interested in comparing two hand-writings to check the hypothesis from Yardeni would not accept any of the results based on such binarization data. For writer identification, the binarization technique should be capable enough to focus on the original written content only, not the surrounding material, not even the markings and scale bars.

![Image](image_url)

Figure 3.3: An illustration of popular binarization techniques applied directly to two DSS fragment images. Sub-figures (a) and (f) show the original IR images (captured in 924 nm wavelength of light). Sub-figures (b) and (g) show the corresponding manually labeled ground truths by human experts. The red-circled areas show the parts where the human experts ignore the irrelevant contents of the images, such as the color-calibration bars, scales, and numbers. The binarization results of techniques proposed by Otsu [203], Niblack [193], and Sauvola [248] are presented for both the fragment images. Unlike a human expert, these three methods fail to provide output images focusing only on the original contents.

3.1.2 Goals

Intensity alone is not a sufficient heuristic in the binarization task. Rather than using a single filtering technique, a bank of trainable filters is needed to solve this problem. The system should be able to ignore irrelevant information during binarization and should be able to include everything which is part of the original content. Furthermore, a system needs to be adequately intelligent to focus on writing like a human does with reasonable accuracy (see Sub figures 3.3b and 3.3g). An artificial neural network with suitable architecture requiring a small amount of training data can be the right solution as a multi-filter trainable method for these diverse materials. Towards both robustness and obtaining optimal results, this article proposes BiNet, an unbiased automatic end-to-end binarization approach for handwritten documents based on the U-Net architecture ([245]) and inspired by the works of ‘pix2pix’, a
general-purpose solution for image-to-image translation problems using conditional adversarial networks ([131]). Configurations of hidden layers and empirically chosen hyperparameters are proposed with a specific loss function to achieve the best outcome. Skip connections are added between the contracting and the expansive path by simply concatenating all channels from one layer to the other. This concatenation circumvents the bottleneck issue at the deepest layers of the encoder and ensures the precise positioning of the foreground-background pixels. Figure 3.8 provides a simple illustration of the proposed network.

This study demonstrates the effectiveness of the proposed model on the binarization task using the collection of the DSS images. Both the RGB-colored images and the pseudo-colored images are used. Pseudo-colored images are fused from grayscale intensity images of different spectral bands. A simple and new technique is proposed for generating ground-truth images of the DSS to train the network. Similar to many historical manuscripts, the DSS collection also lacks ground-truth labels. It is time-consuming and tedious to create ground truth at the pixel level. So the work in this paper ensures that the training data is precise, includes sufficient variability, and the network can perform well with a minimal amount of these training images. So many state-of-the-art complex models are not helpful in this particular case of DSS images. On top of this, transfer learning techniques are introduced to make the proposed model usable for different collections.

The Handwritten Document Image Binarization Competition (H-DIBCO) datasets are also tested to exhibit the system’s effectiveness. Quantitative and qualitative results are presented to compare it with other techniques. BiNet, the proposed model, shows better performance with robustness to the variability of the data. Overall, this article makes the following contributions:

• BiNet: A complete and effective binarization framework for the Dead Sea Scrolls images that allows further analysis of the original contents of this ancient historical collection.
• The network can learn from a few training images to differentiate between relevant and irrelevant information during training, thus providing intelligent and useful binarized outputs.
• An in-depth analysis of the proposed binarization tool, BiNet, using comparative studies and quantitative analysis on different datasets.
• Multi-purpose usability of the binarization tool for different manuscript collections (including H-DIBCO images) using effective transfer learning techniques.
• A useful technique to generate precise ground truths (training images) for the DSS collection that can be extended to degraded historical manuscripts.
• A unique way to generate pseudo-color images by image fusion technique on multi-spectral band images to yield more information than any of the individual infrared band images.
3.2 RELATED WORKS

Figure 3.4: The evidence for ink versus background resides only partly in the intensity of an individual pixel. The evidence is also heavily present in external features (of papyrus and parchment). The figure shows four RGB-color images of full plates from the DSS collection (from left to right: 463A, 464, 1080, and 1082). The first two have papyrus as a surface material for writing, and the latter have parchment. Both materials show distinctive degradation and decaying of characters (inks). The binarization tool should be able to explicitly find the separation of ink from the surface materials.

Document image binarization is one of the most common research problems addressed numerous times in document analysis. Some of these methods have achieved great success in many applications and have become popular over time. Otsu [203] is one of the most commonly used methods. This unsupervised and non-parametric method automatically selects a global threshold based on the grayscale histogram of a given image with no prior information. The Otsu method is one of the simplest binarization methods that perform well when the image is qualitatively clean with a uniform background. Unfortunately, most historical manuscripts do not contain a consistent background or a clear bimodal pattern. Thus, a global thresholding approach is unsuitable for these types of documents [146]. Instead, a gradual change in the uniformity of the background can be handled using small local patches of the target image by local adaptive thresholding. Several local thresholding methods have been developed, such as Niblack [193], Sauvola [248], local max-min by Su et al.[277], and AdOtsu [185]. Descriptive statistics (mean and standard deviation) are calculated on the local area of a pixel to obtain the local threshold. This local thresholding technique is then performed over the whole target image. It performs well compared to the global thresholding techniques but often shows poor performance in the case of historical manuscripts where the documents are highly degraded with extremely non-uniform backgrounds. In cases similar to the DSS collections, both the global and local thresholding methods fail to provide valuable results (an example of this can be found in Figure 3.3 from the previous section).

Several image-processing techniques are used as an enhancement part of the document-processing pipeline along with the binarization itself to improve the results of threshold-based binarization. Shi et al. used the mathematical morphological operator and region-growing techniques [258]. To compute the final threshold, a
3.2 Related Works

Wiener filter \cite{306} is used for the background surface by Gatos et al. \cite{87}. Instead of using the Wiener filtering, robust regression is used by Vo et al. for document binarization of noisy and non-uniform background \cite{302}. Phase-derived features are used for ancient document image binarization in the works of Nafchi et al. \cite{191}. To enhance and reconstruct degraded documents, a method using non-local patch means (NLPM) is proposed by Moghaddam et al. \cite{184}. Bio-inspired models have already been used for text detection in natural images \cite{318}. Similarly, models based on the OFF-center ganglion cells of the human visual system are used to improve document enhancement and binarization \cite{319}. Different contrast enhancements are performed to adjust local grayscale contrast to enhance the binarization results compared to traditional threshold-based techniques on DIBCO datasets \cite{173}.

Many previous works on binarization have exploited the prior knowledge of texts in the document. The edge pixels of the texts can be extracted by techniques similar to the Canny edge detector \cite{44}. This technique is already proposed by Chen et al. in their double threshold image binarization method \cite{48}. One generalization of edge pixels is transition pixels with extreme transition values. These pixels are calculated on a small neighborhood using the intensity difference, and then the gray-intensity threshold is calculated from the statistical information of the transition set \cite{234}. On the contrary, structural symmetric pixels (SSPs) are used to determine local thresholds in a neighborhood, and a voting system is utilized for multiple thresholds for the binarization task \cite{135}. Automatic parameter tuning can be done using a global energy function inspired by a Markov random field model by incorporating edge discontinuities \cite{127}. All these methods are primarily based on traditional image processing and pattern recognition techniques and may have promising characteristics. However, they are designed to attain good results on certain types of documents and lack in addressing the diversified degradation problems similar to the DSS collection with its broad spectrum of writing-surface materials.

With the success of deep learning in sophisticated image understanding \cite{96}, several neural network architectures have been proposed for handwritten document binarization and analysis. A fully convolutional network (FCN) is proposed, which operates at multiple image scales starting from full-resolution \cite{281}. A convolutional encoder-decoder is used by Peng et al. \cite{208} on the LRDE document binarization dataset \cite{159,160}. In the recent works of Calvo-Zaragoza et al., a selectional autoencoder is used for the binarization task \cite{40} on a couple of DIBCO datasets \cite{198,223}, Balinese palm leaf manuscripts \cite{39}, Persian documents from PHI \cite{12}, and music notations from SAM and ES \cite{41}. Afzal et al. have proposed using the recurrent neural network for document binarization \cite{5}. This work has been extended by Westphal et al. \cite{305} by using Grid Long Short-Term Memory (Grid LSTM) networks \cite{141} for the binarization task. A hierarchical deep supervised network (DSN) architecture is proposed, which performs better than the Grid LSTM on DIBCO datasets \cite{303}. In the recent work of He et al., an iterative deep learning technique is proposed for document enhancement and binarization \cite{117} on several DIBCO datasets, the
Bickley-diary dataset [62], the PHIDB dataset [190], and the Synchromedia Multi-spectral dataset [122]. Several new works also use DIBCO images and their specific datasets to propose improved binarization techniques. For example, Souibgui et al. proposed DE-GAN to restore severely degraded documents [270], Dang et al. used Stroke Boundary Feature Guided Network [58], Akbari et al. used CNN [6], De et al. proposed Dual Discriminator Generative Adversarial Network (DD-GAN) [60]. Lin et al. also used GAN and discrete wavelet transform on the DIBCO dataset [166]. Yand et al. used vision transformer networks on different public datasets from DIBCO and HDIBCO [313].

These neural network-based techniques are helpful and present improved performances in many cases. However, most of the time, the datasets used do not pose instances of extreme degradation along with diverse material textures like parchment and papyrus (see Figure 3.4). Texture modeling can be performed using the Markov random field (MRF) [56], but it will be highly complicated in the case of DSS images. Explicit foreground and background modeling have been proposed by Sriman et al. in the classification of text blocks in the scene images [273]. Still, the binarization task needs precise localization of each ink pixel. On top of this, in the case of the fragment images of the DSS collection, the unnecessary elements and modern number tags should be ignored in the outcome of the binarization (as discussed in the previous section; see Figure 3.3). On the one hand, the desired binarization tool should be robust enough to handle highly degraded historical manuscripts written in parchments and papyrus, like the DSS collection, and, on the other hand, should perform well in general document cases, including the DIBCO images.

Although many previous works have already provided us with different tools for benchmark datasets, a robust mechanism is yet to be designed that performs not only outstanding binarization for severe cases like the DSS collection but also shows consistent performance in general cases. The implementations of U-Net [245] and pix2pix [131] methods are particularly relevant here. Though U-Net was initially designed for biomedical-image segmentation, this has already been used for accurate pixel classification in one of the recent competitions on document image binarization (DIBCO 2017 [224]). On the other hand, pix2pix is proposed initially as a general-purpose solution to image-to-image translation problems using conditional adversarial networks [93, 182] and is not designed to perform document binarization tasks. However, the inspiration lies in the performance of the pix2pix method on image-to-image translation tasks with highly-structured graphical outputs by learning a loss adapted to the task. The proposed method in this article is based on a similar idea where the results are precise and straightforward representations of highly complex inputs.
3.3 METHODOLOGY

This section briefly explains the proposed model and the complete methodology. We first present the dataset along with image fusion techniques. We then propose the BiNet framework, the network architecture with technical details, hyperparameters, and the transfer learning techniques.

3.3.1 Dataset

In this study, we use the high-resolution multi-spectral images provided by the Israel Antiquities Authority (IAA), which derive from their Leon Levy Dead Sea Scrolls Digital Library project. These images are offered to scholars and the general public on their website [10]. For each of the scrolls fragments, the IAA produces one color image and several multi-spectral images on both recto and verso in 28 different exposures with a resolution of 1215 pixels per inch (PPI) at 1:1 ratio [261]. In addition to the fragment images, there are also color images of the full plates where the fragments are physically preserved (see Figure 3.5). Depending on the arrangement, a full plate may contain one fragment or several different fragments. In this study, we first use the fragment images to train and test the model. Later, we use the plate images through the transfer learning technique (as described in 3.3.6).

The **fragment images** have a dimension of $5412 \times 7216$ or $7216 \times 5412$ pixels, depending on the orientation of the physical fragment. We scale this to 50% to speed up the training and testing process. The resulting dimensions are, therefore,
2706 × 3608 or 3608 × 2706 pixels, respectively. The proposed network, BiNet, takes input images with a dimension of 256 × 256. We, therefore, divide the input image into small images of 256 × 256 pixels. This way, we end up with 165 small images per original image. As the dimensions of the images are not divisible by 256, the cuts from the edges of the images have a smaller size. We change these images to size 256 × 256 by padding them. This exact procedure can accommodate any image size at the network’s input. We train three models with different inputs of fragment images, so we can see which of the three images gives the best binarization result:

- RGB-color images (captured in visible light; 445 nm wavelength)
- Grayscale intensity images (captured in infrared; 924 nm wavelength)
- Fused images (details in Subsection 3.3.2)

The plate images have variable dimensions of approximately 3000 × 4000 pixels. They all are RGB-color images. We follow the same procedure as in the case of fragment images to produce small images of 256 × 256 from the plate images.

In addition to the DSS images, we also use the publicly available (H-)DIBCO datasets from the document image binarization competitions of years 2009 through 2018 [88, 198, 220–225]). Finally, to check the system’s robustness, we use grid images from several historical manuscripts (non-DSS) from the Monk system [253].

3.3.2 Image fusion

We use the grayscale image resulting from the light intensity at each pixel in the wavelength of 924 nm. This image is an interesting choice from our end. We selected this particular wavelength because the resulting grayscale image shows the maximum visible contrast between the ink and the background compared to any other wavelengths. In addition, we take advantage of other multi-spectral band images to extend our work further and improve the ink-background separation. We propose here an image-fusion technique to create a pseudo-color image. We take grayscale intensity images from three separate wavelengths: 595 nm, 638 nm, and 924 nm. With these three images, we produce a new fused image (pseudo-color image) with three channels. Image with wavelength 595 nm goes to the R-channel, 924 nm to the G-channel, and 638 nm to the B-channel. Figure 3.6 shows an example of three grayscale images and the resulting pseudo-color RGB-image from image-fusion. By doing this, we hope to capture more details that emerge from various lighting conditions to improve the binarization result.

3.3.3 Ground truth

One of the biggest challenges in working with DSS images is the lack of ground truths. Besides, unlike many other historical manuscripts, the DSS collection is not
a structured or complete dataset. To use trainable networks, we first need labeled train images. We use GIMP, a free and open-source raster graphics editor (version 2.8.16 [145]), to create the ground truths. To establish the credibility of the ground truths, palaeographic experts labeled the images.

Figure 3.7: An illustration of the procedure to create ground-truth data from the DSS fragment images using the GIMP tool. The image is from fragment 9 of plate 497 (column 2, row 1).

We propose a simple method for the labeling task using the GIMP tool. First, a transparent layer is created on the top of the fragment image to train the network. This layer is of the same dimension as the original image. Now, by zooming into the vicinity of the characters, the palaeographic experts mark the inks-pixels in red, capturing the characters from the bottom layer (original image) to the transparent
layer (new image: the ground truth) with a pen of 1 pixel accuracy. This work is similar to creating the carbon copy of a document, but the other way around. In this task, the palaeographic expert overwrites the original content on the transparent layer. Due to choosing a pen size of 1 pixel, we ensure that only the ink pixels are marked red and everything else remains transparent. Once the task is complete, the transparent layer is removed and saved as a separate image where the red pixels are converted to black and everything else to white (ground truth). As the task is done using a computer mouse, it is simple but time-consuming. It should be noted that the palaeographic experts were cautious about producing the ink-pixel labels as accurately as possible, conservatively avoiding marking any non-ink pixels. This labeling task took around 4-8 hours for each image. We have selected 51 fragment images to be labeled for this experiment. The selection was made carefully to accommodate maximum diversities to represent the whole collection (a complete list is attached in Table 3.5 in the Appendix).

For the plate image, we use the already labeled fragment images (for that plate) by manually putting them on a transparent layer created on top of it. We have used 17 plates to re-train the model (a complete list is attached in Table 3.6 in the Appendix at the end of this chapter).

![Figure 3.8: The proposed network architecture shows the encoder (contracting path) at the left half and the decoder (expanding path) at the right half of the image. Each step in the decoder part receives a concatenation with the corresponding feature map from the encoder part through the skip connections. This concatenation circumvents the bottleneck issue at the deepest layers of the encoder and ensures the precise localization of the foreground-background pixels.](image)

### 3.3.4 BiNet

The document image binarization task is simply a two-class classification problem. For handwritten documents, we define the two classes as foreground and background. The simple solution is to decide on labels for each pixel of an image, one by one. The background study already shows several methods working in this direction. However, this is a problematic direction for the cases of highly degraded historical manuscripts, even with local thresholding in a small neighborhood. Therefore, we need a trainable network that learns the target content and classifies each pixel by
considering its neighbors in a local region. In this article, we propose a new approach utilizing a network architecture, the BiNet, that works end-to-end, providing binarization in just one step considering the original content’s knowledge. Rather than classifying each pixel separately, the BiNet efficiently works on the whole input image to generate a binarized output image of the same size.

The implementation of BiNet is a variant of the U-Net architecture [245] capable of doing complex binarization tasks. The original U-Net was designed for biomedical image segmentation with precise localization and can be trained end-to-end with very few images. Due to these two traits, we follow the typical shape of a U-Net with skip connections to build our specific model that can handle diverse manuscript images with sparse information, unlike biomedical image data. Thus, our model becomes an image generator (which shares similarity with pix2pix [131] but differs in the adversarial parts) that produces a binarized image from an input DSS image. The original content (the texts; ground truth) of a DSS image is accompanied by several factors, including the texture of the writing surface, various degradation, and irrelevant materials (numbers, scale bars, and the surface of the platform). The BiNet learns a mapping from the input image $x$ to output image $y$:

$$x = y + \delta$$  \hspace{1cm} (3.1)

where $x$ is the DSS image as it is with degradation and other factors, $y$ is the latent original content (ink from the original writing; the ground truth), and $\delta$ is the noise that comprises all the information except the original content. The network is trained by minimizing the classification error from the $L1$-loss function:

$$L_1 = \frac{1}{n} \sum_{i=1}^{n} |y_{true} - y_{predicted}|$$  \hspace{1cm} (3.2)

where $n$ is the number of pixels in the input image, $y_{true}$ is the ground-truth, and $y_{predicted}$ is the prediction from the network. As the binarized output is less complex than the input image, simply using the $L1$ regression is enough. Additionally, the $L1$ loss function is less affected by the outliers, making it a preferable choice over the $L2$ loss function for the DSS collection. The $L2$ loss tends to produce the border effects.

### 3.3.5 Network architecture

The network architecture is illustrated in Figure 3.8. The technical details of each of the layers can be found in Table 3.1. In the encoder part, we have Convolution-BatchNorm-LeakyReLU layers with a different number of filters. The decoder consists of Convolution-BatchNorm-Dropout-LeakyReLU with a 50% dropout in the first three layers; then, the remaining layers consist of a Convolution-BatchNorm-LeakyReLU structure. We used the activation function of the leaky rectified linear
3.3 Methodology

Table 3.1: Detailed description of the BiNet architecture. Here, $\text{Conv}(f, h, w)$ denotes a convolutional operator with $f$ filters and $h \times w$ kernel sizes; $\text{BNorm}()$ refers to batch normalization; $\text{Dropout}(r)$ denotes dropout operation with a ratio of $r$ connections at each time; $\text{LeakyReLU}$ refers to the leaky rectified linear unit; $\text{Tanh}$ refers to Tanh activation.

<table>
<thead>
<tr>
<th>Original input</th>
<th>Input at Encoder</th>
<th>Encoding layers</th>
<th>Decoding layers</th>
<th>Final output</th>
</tr>
</thead>
<tbody>
<tr>
<td>256x256</td>
<td></td>
<td>Conv (64,4,4,2)</td>
<td>Conv (512,4,4,2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>128x128</td>
<td></td>
<td>Conv (128,4,4,2)</td>
<td>Conv (1024,4,4,2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>[0, 255]^{256 \times 256}</td>
<td></td>
<td>Conv (256,4,4,2)</td>
<td>Conv (1024,4,4,2)</td>
<td>[0, 1]^{256 \times 256}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>[0, 255]^{256 \times 256 \times 3}</td>
<td>or,</td>
<td>Conv (512,4,4,2)</td>
<td>Conv (1024,4,4,2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>[0, 255]^{256 \times 256 \times 3}</td>
<td></td>
<td>Conv (512,4,4,2)</td>
<td>Conv (512,4,4,2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>[0, 255]^{256 \times 256 \times 3}</td>
<td></td>
<td>Conv (512,4,4,2)</td>
<td>Conv (512,4,4,2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>[0, 255]^{256 \times 256 \times 3}</td>
<td></td>
<td>Conv (512,4,4,2)</td>
<td>Conv (256,4,4,2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>[0, 255]^{256 \times 256 \times 3}</td>
<td></td>
<td>Conv (512,4,4,2)</td>
<td>Conv (128,4,4,2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
<tr>
<td>[0, 255]^{256 \times 256 \times 3}</td>
<td></td>
<td>Conv (512,4,4,2)</td>
<td>Conv (1,4,4)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BNorm ()</td>
<td>BNorm ()</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actv (LeakyReLU 0.2)</td>
<td>Actv (LeakyReLU 0.2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dropout (0.5)</td>
<td></td>
</tr>
</tbody>
</table>

The convolutions are $4 \times 4$ spatial filters applied with stride 2 and padding 1. The hyperparameters are set empirically through grid search. No max-pooling layer is used. Experimental results show that BiNet is one of the optimal topologies for DSS image binarization. The model works with a fixed image size of $256 \times 256$, accepts both color (3 channel) and grayscale (1 channel) images, and always outputs a $256 \times 256$ binary image. Please note that the document image...
can be much larger and of variable sizes. The implementation processes all the input images by dividing them into equal pieces of $256 \times 256$ patches (padding is performed if necessary) to provide the input images to the network and combine the individual outputs to get the full binarized image.

Instead of ReLU, we used LeakyReLU to avoid collapsing gradient. Parametric ReLU has the same advantage with one difference the slope of the output for negative inputs is a learnable parameter, while in the Leaky ReLU, it is a hyperparameter. Convolutional layers replace max-pooling with increased stride making it a subsampling step. The network is large enough for the DSS images to be trained on, and it can learn all the necessary invariances without using max-pooling layers, without any loss in accuracy [271].

3.3.6 Transfer learning

The BiNet structure is initially proposed for binarizing the DSS fragment images. In the case of full plate images, the whole model pre-trained from scratch using DIBCO images can be retrained to update the network weights. This retraining can be done using a small number of ground-truth plate images due to the high similarity of the plate images to the DIBCO images. A BiNet architecture can also be used in different historical manuscript collections using this simple transfer learning technique with only a small amount of training data.

3.4 experiments

This section briefly discusses different aspects of the experiment, including the training procedures and the evaluation matrices for quantitative analysis.

3.4.1 Training

From the labeled fragment images, we use 40 images for training and 11 for testing and evaluation. Each of these images contains one fragment and has been manually binarized. We train this network to minimize the $L1$ loss. We use the Adam optimizer with an initial learning rate of 0.0002. The model is trained for 200 epochs. We train both the proposed framework (BiNet) and the original CGAN model (from pix2pix [131]) on color, grayscale, and fused fragment images separately (from scratch) as described in Subsection 3.3.1. Additionally, we train our model on the DIBCO datasets from scratch. We use the DIBCO datasets from 2009 to 2014 as training data and those from 2016 to 2018 as test data. We use a pre-train model for plate images and updated the weights by re-training it for another 200 epochs using the 16 manually labeled plate images. The system runs on a personal workstation with a
single GPU (NVIDIA GTX 1060 with 6GB memory; details can be found in Table 3.8 and Table 3.9 in the Appendix).

3.4.2 Evaluation measures

For quantitative analysis, we use evaluation metrics that are commonly used in the (H-)DIBCO [88, 198, 220–224]. The metrics are suitable in the context of document analysis. We will use four metrics: F-measure \( (F) \), pseudo-F-measure \( (F_{ps}) \), peak signal-to-noise ratio \( (PSNR) \), and distance reciprocal distortion \( (DRD) \). Brief descriptions for each of them are provided in the following subsections.

3.4.2.1 F-measure

F-measure (also F1-score or F-score) measures a test’s accuracy. It is defined as:

\[
F_{\text{meas}} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}
\]  

(3.3)

where, \( \text{Recall} = \frac{TP}{TP + FN} \) and \( \text{Precision} = \frac{TP}{TP + FP} \); \( TP, FP, FN \) refer to the True Positive, False Positive, and False Negative values, respectively.

3.4.2.2 pseudo-F-measure

Pseudo F-measure follows the same formula of F-measure (Eq. 3.3), but it uses pseudo-Recall and pseudo-precision [197].

\[
pseudoF_{\text{meas}} = \frac{2 \times \text{pseudoRecall} \times \text{pseudoPrec.}}{\text{pseudoRecall} + \text{pseudoPrec.}}
\]  

(3.4)

Both these pseudo metrics use distance weights concerning the contour of the ground-truth characters. In the case of pseudo-Recall, the weights of the ground-truth inks are normalized according to the local stroke width. These weights are defined between [0, 1]. In the case of pseudo-Precision, the weights are constrained within an area that expands to the background of the ground truth, considering the stroke width of the nearest ground-truth component. Inside this area, the weights are more significant than one (generally between [1, 2]); outside this area, they are equal to one.
3.4.2.3 Peak Signal-to-Noise Ratio (PSNR)

The peak signal-to-noise ratio is the measure of how close an image is to another one. The higher the value of PSNR, the higher the similarity of the two images being compared.

\[
PSNR = 10 \log\left( \frac{C^2}{MSE} \right)
\]

where, \(C\) is the difference between foreground and background, and \(MSE\) is the mean squared error and defined as:

\[
MSE = \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} (I_{bin}(x,y) - I'_{bin}(x,y))^2}{MN}
\]

where, \(M \times N\) is the dimension of the image, \(I_{bin}\) is the original ground-truth image and \(I'_{bin}\) is the test-output of the ground-truth from the model.

3.4.2.4 Distance Reciprocal Distortion (DRD)

The distance reciprocal distortion metric correlates with the human visual perception system and measures the distortion for all the \(S\)-flipped pixels as:

\[
DRD = \frac{\sum_{k=1}^{S} DRD_k}{NUBN}
\]

where, \(DRD_k\) is the distortion of the \(k\)-th flipped pixel that is calculated using 5 x 5 normalized weight matrix. The weight matrix \(W_{Nm}\) is defined by Lu et al. [174], where they used the DRD to measure the visual distortion in binary document images. \(NUBN\) is the number of non-uniform 8 x 8 blocks in the ground-truth image.

3.5 Results

This section presents the experimental results based on four different evaluation measures presented in Subsection 3.4.2. The results are obtained using three different test sets. The first test set consists of 11 fragment images of the DSS collection for which the corresponding ground truths were built manually, as described in Subsection 3.3.3. The test images are selected to accommodate maximum diversity and different degradation (a list is attached in Table 3.5 in the Appendix). The second test set contains 40 images from H-DIBCO 2016, DIBCO 2017, and H-DIBCO 2018 datasets. Both H-DIBCO 2016 and 2018 have ten handwritten document images each, and DIBCO 2017 has twenty document images: ten machine-printed and ten
handwritten. Finally, the third test set consists of 3 RGB-colored full plate images from the DSS collection.

![Figure 3.9](image1)

Figure 3.9: A comparative illustration of test results using different traditional methods and the proposed BiNet model on fragment 1 of plate 1082. Please note the successful removal of the color-calibrator strip and machine-printed number tag.

![Figure 3.10](image2)

Figure 3.10: Results of BiNet and traditional methods presented on a zoomed-in part of fragment 1 of plate 1082. The zoomed-in (enlarged) part is taken from the pixel position of (460, 150) with a window size of 780 x 780 pixels. A visual inspection shows that the BiNet (fused) output is the closest match to the ground truth.

For the first test-set, we trained two different models: the proposed BiNet and the conditional adversarial network (CGAN) as presented in pix2pix (the image-to-image translation work [131]). The reason behind training a CGAN is to evaluate the potential of the adversarial network for learning complex tasks with a smaller number of training images. Additionally, it is worthwhile to present the results of CGAN as the BiNet shares similar generative architecture. The CGAN and BiNet
models are trained on three different types of images: grayscale, color, and fused (pseudo-color). Thus, we tested the first set of images on six different networks (2 models trained from scratch on three different types of training data).

To present a quantitative evaluation, we use four traditional thresholding methods to perform binarization: Otsu [203], Niblack [193], Sauvola [248], and local implementation of Otsu with $70 \times 70$ windows. The quantitative results for the first set, the DSS images, are presented in Table 3.2. The CGAN models show promising performance compared to the traditional methods. But all the BiNet models outperform the rest. BiNet on fused is the best-performing model in all four evaluation measures.

Table 3.2: Detailed evaluation results on the DSS fragment images. The results are presented as $mean \pm std.dev.$ on the whole test set. The proposed BiNet outperforms other methods (best performance in bold).

<table>
<thead>
<tr>
<th>Method</th>
<th>F-measure</th>
<th>pF-measure</th>
<th>PSNR</th>
<th>DRD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Otsu (global) on grayscale</td>
<td>7.3 ± 3.5</td>
<td>7.3 ± 3.5</td>
<td>1.6 ± 0.5</td>
<td>1236 ± 644.8</td>
</tr>
<tr>
<td>Niblack on grayscale</td>
<td>15.7 ± 8.8</td>
<td>15.9 ± 9.1</td>
<td>4.9 ± 1.0</td>
<td>589.4 ± 317.7</td>
</tr>
<tr>
<td>Sauvola on grayscale</td>
<td>19.1 ± 10.3</td>
<td>19.5 ± 10.7</td>
<td>6.3 ± 1.1</td>
<td>429.5 ± 228.1</td>
</tr>
<tr>
<td>Otsu (local) on grayscale</td>
<td>51.3 ± 13.4</td>
<td>54.9 ± 13.6</td>
<td>14.8 ± 1.2</td>
<td>53.7 ± 25.8</td>
</tr>
<tr>
<td>CGAN on grayscale</td>
<td>63.6 ± 16.6</td>
<td>65.1 ± 15.7</td>
<td>17.3 ± 2.2</td>
<td>29.8 ± 20.2</td>
</tr>
<tr>
<td>CGAN on fused</td>
<td>68.4 ± 18.2</td>
<td>70.5 ± 16.7</td>
<td>17.8 ± 2.3</td>
<td>28.6 ± 23.1</td>
</tr>
<tr>
<td>CGAN on color</td>
<td>71.5 ± 8.7</td>
<td>72.9 ± 8.4</td>
<td>17.2 ± 2.0</td>
<td>28.9 ± 15.2</td>
</tr>
<tr>
<td>BiNet on grayscale</td>
<td>80.3 ± 20.7</td>
<td>82.6 ± 19.2</td>
<td>20.5 ± 3.8</td>
<td>18.7 ± 24.6</td>
</tr>
<tr>
<td>BiNet on color</td>
<td>83.5 ± 9.9</td>
<td>85.8 ± 9.0</td>
<td>20.3 ± 2.9</td>
<td>15.2 ± 14.4</td>
</tr>
<tr>
<td><strong>BiNet on fused</strong></td>
<td><strong>86.7 ± 9.4</strong></td>
<td><strong>89.3 ± 8.3</strong></td>
<td><strong>21.3 ± 3.4</strong></td>
<td><strong>13 ± 14.8</strong></td>
</tr>
</tbody>
</table>

For the second test set of (H-)DIBCO images, we used the BiNet model only. The quantitative results of the second set are shown in Table 3.3. For the (H-)DIBCO results, we also present the performances of the winning methods from each year. Though our model is designed initially for DSS images, it shows high performance in the case of (H-)DIBCO images. For the datasets of 2016 and 2017, BiNet outperforms the best-performing method in a couple of evaluation measures. For the cases of the 2018 dataset, BiNet is almost as good as the best-performing method (Table 3.3).

For the third test set of RGB-colored full plate images, we used the transfer learning technique on the BiNet model initially trained on (H-)DIBCO images. The results are presented in Table 3.4. The BiNet outperforms the traditional methods by significant differences and obtains high evaluation measures. This performance shows the excellent usability of the system on different datasets with a small number of training data by transfer learning techniques.
Table 3.3: Detailed evaluation results on (H-)DIBCO datasets. The BiNet performs equally well with the best-performing methods from each competition (best performance in **bold**). The output images are attached in Tables 3.10, Tables 3.11, and Tables 3.12 in the Appendix.

<table>
<thead>
<tr>
<th>Method</th>
<th>F-measure</th>
<th>pF-measure</th>
<th>PSNR</th>
<th>DRD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>H-DIBCO 2016</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Otsu (global)</td>
<td>86.7</td>
<td>90</td>
<td>17.8</td>
<td>5.5</td>
</tr>
<tr>
<td>Niblack</td>
<td>56.2</td>
<td>56.3</td>
<td>9.6</td>
<td>57.8</td>
</tr>
<tr>
<td>Sauvola</td>
<td>79.9</td>
<td>81.7</td>
<td>14.8</td>
<td>11.4</td>
</tr>
<tr>
<td>Best method at H-DIBCO’16 [223]</td>
<td><strong>87.6</strong></td>
<td><strong>91.3</strong></td>
<td>18.1</td>
<td>5.2</td>
</tr>
<tr>
<td>BiNet</td>
<td>85.6 (-2.0)</td>
<td>90.7 (-0.6)</td>
<td><strong>18.3 (+0.2)</strong></td>
<td><strong>4.9 (+0.3)</strong></td>
</tr>
<tr>
<td><strong>DIBCO 2017</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Otsu (global)</td>
<td>77.7</td>
<td>80.1</td>
<td>13.8</td>
<td>15.8</td>
</tr>
<tr>
<td>Niblack</td>
<td>57.4</td>
<td>57.6</td>
<td>8.8</td>
<td>44</td>
</tr>
<tr>
<td>Sauvola</td>
<td>80</td>
<td>83</td>
<td>14.3</td>
<td>9.6</td>
</tr>
<tr>
<td>Best method at DIBCO’17 [224]</td>
<td><strong>91.0</strong></td>
<td><strong>92.9</strong></td>
<td>18.3</td>
<td><strong>3.4</strong></td>
</tr>
<tr>
<td>BiNet</td>
<td>90.9 (-0.1)</td>
<td><strong>93.3 (+0.4)</strong></td>
<td><strong>18.3 (+0.0)</strong></td>
<td>3.6 (+0.2)</td>
</tr>
<tr>
<td><strong>H-DIBCO 2018</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Otsu (global)</td>
<td>51.4</td>
<td>53.4</td>
<td>9.7</td>
<td>59.5</td>
</tr>
<tr>
<td>Niblack</td>
<td>45.7</td>
<td>45.9</td>
<td>7.7</td>
<td>80.8</td>
</tr>
<tr>
<td>Sauvola</td>
<td>56.3</td>
<td>58.7</td>
<td>10.9</td>
<td>36</td>
</tr>
<tr>
<td>Best method at H-DIBCO’18 [225]</td>
<td><strong>88.3</strong></td>
<td><strong>90.2</strong></td>
<td><strong>19.1</strong></td>
<td><strong>4.9</strong></td>
</tr>
<tr>
<td>BiNet</td>
<td>84.7 (-3.6)</td>
<td>87.1 (-3.1)</td>
<td>17.4 (-1.7)</td>
<td>7.5 (+2.6)</td>
</tr>
</tbody>
</table>

Table 3.4: Detailed evaluation results on the DSS full plate images. The results are presented as mean ± std.dev. on the whole test set (best performance in **bold**). The output images are attached in the Appendix in Figure 3.15.

<table>
<thead>
<tr>
<th>Method</th>
<th>F-measure</th>
<th>pF-measure</th>
<th>PSNR</th>
<th>DRD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Otsu</td>
<td>27.2 ± 5.8</td>
<td>27.2 ± 5.8</td>
<td>9.5 ± 2.1</td>
<td>206.4 ± 101.8</td>
</tr>
<tr>
<td>Niblack</td>
<td>7.3 ± 5.9</td>
<td>7.3 ± 5.9</td>
<td>1.9 ± 1.4</td>
<td>1375.4 ± 872.7</td>
</tr>
<tr>
<td>Sauvola</td>
<td>47.9 ± 16.1</td>
<td>48 ± 16.2</td>
<td>13.8 ± .08</td>
<td>80.9 ± 43.6</td>
</tr>
<tr>
<td>BiNet</td>
<td><strong>86.9 ± 3.1</strong></td>
<td><strong>91.1 ± 2.4</strong></td>
<td><strong>22.9 ± 2.6</strong></td>
<td><strong>6.3 ± 0.2</strong></td>
</tr>
</tbody>
</table>

Figure 3.9 presents the resulting images from different methods. A zoomed-in portion of the methods can be found in Figure 3.10 for a better qualitative analysis. BiNet is extremely good at binarizing the original content and labeling everything else as background. Inside the area of the original content, BiNet is remarkably similar to the ground truth (Figure 3.10). An interesting finding in the results from the fused images can be seen in Figure 3.12. During the labeling process of the
images, the human expert only labeled the visible inks in the fragments. However, ink parts might become visible in the fused images, making them extractable during binarization. Thus a fused image binarization can reveal more characters than what is visible in a color image (Figure 3.12). Please note that the phenomenon might reduce the quantitative performance as we compare the output with the ground truth. Nevertheless, this is extraordinary and useful in real applications.

Finally, to test the robustness of the model, we collected some additional materials from different collections to perform the binarization using BiNet. One such image is of the famous Nash Papyrus. We used the pretrained BiNet model from the DSS color images to binarize the Nash papyrus, and the result is illustrated in Figure 3.13. The binarization result shows another case where the BiNet can extract the original content by segmenting all the irrelevant materials as background pixels. We also test some grid images from several historical manuscripts (non-DSS) from the Monk system. These results are presented in Figure 3.16 in the Appendix.
3.6 Conclusions

In this article, we proposed a complete framework, the BiNet, to efficiently binarize one of the most degraded historical manuscript collections, the Dead Sea Scrolls. The method can work with the full-plate color images and the grayscale intensity images of the individual fragments. The network we used can learn what is essential and irrelevant. Compared to traditional methods, the proposed BiNet can focus on and binarize the original written content of the document with remarkably high performance, which is crucial for getting as close as possible to the original writer to be able to perform writer identification and document dating. All non-ink information must be removed from these applications for a strictly handwriting-based prediction. One of the significant features of the network is the ability to segment everything except the original writing contents into the background semantically. The binarization results of BiNet demonstrate the robustness and multi-purpose usability of the network on different degraded manuscripts of diverse document textures and layouts.

To facilitate the training of the neural network, we utilized a simple and effective ground truth labeling technique. We proposed an image fusion technique to produce a pseudo-color image from multi-spectral image bands. This technique improved the binarization results. In several cases, these improved results from fused images could extract more of the original contents than the ground truth itself. Though this phenomenon might lead to lower performance on the quantitative analysis, this additional extraction is much more desirable in real-world applications. Like BiNet, the fusion technique can also be used in different collections if multi-spectral images are available. Though our framework was initially designed for the DSS images, we...
3.6 CONCLUSIONS

have also performed experiments with (H-)DIBCO datasets. The BiNet delivered excellent results for these datasets with high-performance measures, similar to the best-performing methods. This success shows the prospect of using BiNet as a one-off tool for manuscript binarization.

In this article, we have worked on the binarization part only. Additionally, both our ground-truth labeling and binarization results were restrictive. We were careful enough to avoid automatic filling or smoothing of the characters. This conservative approach made the output of BiNet crisp and accurate to the ink of each character. As most historical manuscripts show extreme degradation even on the character level, a reconstruction-based binarization needs to be explored in the future. More research can be done on the image fusion technique by fusing more than three channels. In cases of a larger image, we divided the images into smaller ones ($256 \times 256$). During the binarization process, the border areas sometimes lack in obtaining perfect binarization. This problem can be improved by further work in the area and might be solved by using overlapping patches of images. The design of BiNet took into account the low amount of training data with high complexity. If more training data becomes available, a more in-depth network similar to ResNet [107] or even a dense network similar to DenseNet [129] can be explored in the future. Though we propose a complete binarization framework, for now, additional research can always be performed for further improvement of the technique.

Finally, this chapter enables us to perform in-depth experiments on the difficult and diverse images of the DSS, both in the writer identification and dating domain, in the following parts of the thesis. The outcome of BiNet also enabled us to apply character reconstruction and image-based material analysis of the DSS.

ACKNOWLEDGEMENTS FOR CHAPTER 3

The authors (from the original article) would like to thank Mladen Popović (Principal investigator of the ERC project, Faculty of Theology and Religious Studies, University of Groningen), Eibert Tigchelaar (Research Professor, Faculty of Theology and Religious Studies, KU Leuven) for their valuable inputs and time in labeling and creating the ground truths of the DSS images. In addition, the authors would also like to thank Drew Longacre (Post-doctoral fellow, ERC project), Gemma Hayes (Ph.D. candidate, ERC project), Ayhan Aksu (Ph.D. candidate, NWO/FWO project), Marwin Van Dijk (Research assistant, ERC project), and Christine van der Veer (Research assistant, ERC project) for their time and effort in labeling a number of the ground-truth images. The authors would also like to thank Ben Outhwaite of the Genizah Research Unit at Cambridge University Library for the Nash Papyrus. This work has been supported by an ERC Starting Grant of the European Research Council (EU Horizon 2020): The Hands that Wrote the Bible: Digital Palaeography and Scribal Culture of the DSS (HandsandBible # 640497).
This section contains additional information and results related to the article. Figure 3.14 illustrates the box plots for evaluation metrics. The complete list of train and test images can be found in Table 3.5, Table 3.6, and Table 3.7. A brief configuration of the computing system is provided in Table 3.8. Figure 3.15 contains the binarization results of three test images (full-plates) of the DSS collection using BiNet with transfer learning. The BiNet is additionally tested on several different manuscript collections, which is illustrated using the binarization of grid images from Monk in Figure 3.16. Finally, the binarization results of DIBCO/H-DIBCO datasets are presented at the end of this section.

Figure 3.14: Box-plots showing the distribution of test data (DSS fragment-images) for the four different evaluation metrics.
Table 3.5: List of train & test images and material types (for DSS fragment images).

<table>
<thead>
<tr>
<th>Material Type</th>
<th>Train-images</th>
<th>Test-images</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parchment</td>
<td>38</td>
<td>9</td>
<td>47</td>
</tr>
<tr>
<td>Papyrus</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Total</td>
<td>40</td>
<td>11</td>
<td>51</td>
</tr>
</tbody>
</table>

Table 3.6: List of train (for transfer-learning) & test images (for DSS full-plate images).

<table>
<thead>
<tr>
<th>Image Description</th>
<th>Train-images</th>
<th>Test-images</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train-images (columns 1-3)</td>
<td>25 107 207 587 807 1081</td>
<td>386 366 386 386 386 386</td>
<td>1997</td>
</tr>
<tr>
<td>Test-images</td>
<td>25 107 207 587 807 1081</td>
<td>386 366 386 386 386 386</td>
<td>1997</td>
</tr>
</tbody>
</table>

Table 3.7: List of train & test images (from DIBCO/H-DIBCO [88, 198, 220–225]).

<table>
<thead>
<tr>
<th>Image Description</th>
<th>Train-images</th>
<th>Test-images</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of images:</td>
<td>Train: 75, Test: 40</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3.8: Brief configuration of the work-station.

<table>
<thead>
<tr>
<th>CPU</th>
<th>Memory</th>
<th>Display card</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel(R) Core(TM) i5-4590 @ 3.3GHz</td>
<td>775MB</td>
<td>GeForce GTX 1060 6GB</td>
</tr>
<tr>
<td>size: 3.3GHz</td>
<td>size: 3.3GHz</td>
<td>vendor: NVIDIA Corporation</td>
</tr>
<tr>
<td>description: System memory</td>
<td>vendor: NVIDIA Corporation</td>
<td>width: 64 bits; clock: 33MHz</td>
</tr>
<tr>
<td>capacity: 3750MHz, width: 64 bits</td>
<td>width: 64 bits; clock: 33MHz</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.9: Time needed to binarize one of the test images (fragment 1, plate 1082).

<table>
<thead>
<tr>
<th>Image Type</th>
<th>Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grayscale image</td>
<td>0 min 28.905 sec</td>
</tr>
<tr>
<td>RGB-color image</td>
<td>0 min 34.991 sec</td>
</tr>
<tr>
<td>Pseudo-color image</td>
<td>1 min 3.987 sec</td>
</tr>
</tbody>
</table>
Figure 3.15: Binarization results of three test images (DSS full-plate images) using BiNet (trained on DIBCO images, then updated by transfer learning using sixteen manually labeled plate images).
Figure 3.16: An illustration of BiNet output of four different test images (grid-images) created with various manuscript collections from Monk [253]. The BiNet model used here is trained on DIBCO images. Results from Otsu (global) and Sauvola are presented as well.
Table 3.10: Figures of the H-DIBCO 2018 testing dataset along with the binarization results from BiNet.

<table>
<thead>
<tr>
<th>Original image</th>
<th>BiNet output</th>
<th>Ground-truth</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
<td><img src="https://via.placeholder.com/150" alt="Image" /></td>
</tr>
</tbody>
</table>
Table 3.11: Figures of the DIBCO 2017 testing dataset (the machine-printed ones) along with the binarization results from BiNet.

<table>
<thead>
<tr>
<th>Original image</th>
<th>BiNet output</th>
<th>Ground-truth</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.jpg" alt="Image" /></td>
<td><img src="image2.jpg" alt="Image" /></td>
<td><img src="image3.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image4.jpg" alt="Image" /></td>
<td><img src="image5.jpg" alt="Image" /></td>
<td><img src="image6.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image7.jpg" alt="Image" /></td>
<td><img src="image8.jpg" alt="Image" /></td>
<td><img src="image9.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image10.jpg" alt="Image" /></td>
<td><img src="image11.jpg" alt="Image" /></td>
<td><img src="image12.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image13.jpg" alt="Image" /></td>
<td><img src="image14.jpg" alt="Image" /></td>
<td><img src="image15.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image16.jpg" alt="Image" /></td>
<td><img src="image17.jpg" alt="Image" /></td>
<td><img src="image18.jpg" alt="Image" /></td>
</tr>
</tbody>
</table>

3.6 Conclusions
Table 3.12: Figures of the H-DIBCO 2016 testing dataset along with the binarization results from BiNet.

<table>
<thead>
<tr>
<th>Original image</th>
<th>BiNet output</th>
<th>Ground-truth</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image 1" /></td>
<td><img src="image2.png" alt="Image 2" /></td>
<td><img src="image3.png" alt="Image 3" /></td>
</tr>
<tr>
<td><img src="image4.png" alt="Image 4" /></td>
<td><img src="image5.png" alt="Image 5" /></td>
<td><img src="image6.png" alt="Image 6" /></td>
</tr>
<tr>
<td><img src="image7.png" alt="Image 7" /></td>
<td><img src="image8.png" alt="Image 8" /></td>
<td><img src="image9.png" alt="Image 9" /></td>
</tr>
<tr>
<td><img src="image10.png" alt="Image 10" /></td>
<td><img src="image11.png" alt="Image 11" /></td>
<td><img src="image12.png" alt="Image 12" /></td>
</tr>
</tbody>
</table>

Note: Images are not visible in this text-based format.