DeepOtsu

He, Sheng; Schomaker, Lambertus

Published in:
Pattern recognition

DOI:
10.1016/j.patcog.2019.01.025

Pattern Recognition 91 (2019) 379–390

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

DeepOtsu: Document enhancement and binarization using iterative deep learning

Sheng He*, Lambert Schomaker

Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, Groningen, PO Box 407, 9700 AK, The Netherlands

ARTICLE INFO

Article history:
Received 15 August 2018
Revised 6 December 2018
Accepted 13 January 2019
Available online 25 January 2019

Keywords:
Document enhancement and binarization
Convolutional neural networks
Iterative deep learning
Recurrent refinement

ABSTRACT

This paper presents a novel iterative deep learning framework and applies it to document enhancement and binarization. Unlike the traditional methods that predict the binary label of each pixel on the input image, we train the neural network to learn the degradations in document images and produce uniform images of the degraded input images, which in turn allows the network to refine the output iteratively. Two different iterative methods have been studied in this paper: recurrent refinement (RR) that uses the same trained neural network in each iteration for document enhancement and stacked refinement (SR) that uses a stack of different neural networks for iterative output refinement. Given the learned nature of the uniform and enhanced image, the binarization map can be easily obtained through use of a global or local threshold. The experimental results on several public benchmark data sets show that our proposed method provides a new, clean version of the degraded image, one that is suitable for visualization and which shows promising results for binarization using Otsu's global threshold, based on enhanced images learned iteratively by the neural network.

1. Introduction

Extracting useful information from images of historical documents is a challenging problem because these images usually suffer from various degradations [1], such as noise, spots, bleed-through, or low-contrast ink strokes [2]. A modern retrieval system, such as the Monk system [3], which is a web-based search engine for handwritten image collections, can only provide satisfying results on high-quality handwritten images. In addition, most methods for document analysis require preprocessed and clean documents as inputs for performance to be good [4,5]. Document enhancement and binarization are the main pre-processing steps in the document analysis process. Document enhancement addresses the problems involved in improving the perceptual quality of document images and in removing degradations and artifacts present in images [6], with the aim of restoring their original look [7]. Document binarization is the task of separating each pixel of text and background [1]. Enhancement is another pre-processing step for binarization of degraded document images, aimed at removing as much unnecessary noise as possible. Although many document enhancement methods have been proposed in the literature, most of them focus on one specific problem, such as bleed-through correction [6,8–10]. Few existant unsupervised methods can handle multiple degradations in historical documents.

Convolutional Neural Networks (CNNs) [11] have been successfully used in image classification [12], providing significant improvements over traditional methods in various applications. They have also been applied to document binarization [13,14]. Since the output of binarization is the same size as the input image, well-known frameworks of neural networks are often used, such as fully convolutional neural networks (FCNs) [15,16], the holistically nested edge detector (HED) [14,17], and U-Net [18]. Using deep learning provides a wide margin of performance gain compared to traditional methods because millions of parameters in neural networks have been learned in a large training data set.

In this paper, we propose the iterative deep learning framework shown in Fig. 1. We train the network to improve the input images, for example by removing noise or correcting various degradations. Thus, the output of the neural network constitutes the improved version of the input with supervised learning. The neural network learns the difference between the input and the expected output, which might involve noise or other degradations. The output can also be fed into the neural network for refinement using various iterations. After several iterations, the output that constitutes the improved version of the input can be used as the input for final classifiers to improve the performance. The block of

* Corresponding author.
E-mail addresses: heshengxgd@gmail.com (S. He), L.Schomaker@ai.rug.nl (L. Schomaker).

https://doi.org/10.1016/j.patcog.2019.01.025
0031-3203/

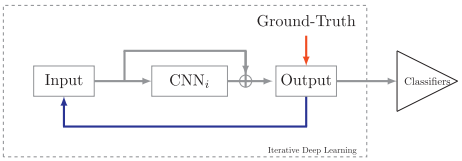


Fig. 1. The proposed iterative deep learning framework. The output of CNN$_i$ is the modified version of the input; thus it can be fed into the network iteratively for fine-tuning using various iterations. The output after several iterations, which is the improved version of the input, can be used as the input for the final classifiers (SVM or Neural networks).

iterative deep learning can be seamlessly integrated into any existant framework, which can be considered as supervised data augmentation pre-processing.

In this paper, we apply iterative deep learning to document enhancement, whereby the final classifier is the traditional binarization method. Unlike traditional binarization methods that train the neural network to predict the label of each pixel, we train the neural network to learn the degradation and correct the degraded document iteratively. As expected, the output of the neural network results in uniform and clean images, rather than binary maps, which then allows the network to learn the degradations, i.e. the differences between the degraded and clean images. Since the output of the neural network is the improved version of the input, it can also be fed into the network for fine-tuning. More precisely, the learned neural network can be used recursively to refine the results, because the output image can also be considered as a slightly degraded image if the learned neural network does not provide a good result in the first iteration. In addition, given the uniform image, which is corrected by the learned neural network, the binarized image can be easily and efficiently obtained using a global threshold, such as Otsu's global threshold [19].

Note that the output of the proposed method is a uniform and clean version of the degraded input image, which ensures acceptable viewing of the degraded images for end-users, such as historians, paleographers, and scholars. What is desirable is that the visualization of the enhanced image should maintain the original appearance as much as possible, while removing the textures and degradations found in the background. Our proposed method is able to provide a better view of the degraded document images by only showing the original text, with the noise and degradations removed. Fig. 2 presents two examples of original degraded images along with the corresponding enhanced images resulting from the proposed method; this shows that the enhanced images are more readable for end-users than the original documents.

The differences between the proposed method and the previous ones [14,16,20,21] can be summarized as follows: (1) Unlike previous methods that train the neural network to learn the labels of each pixel, the output of our method is a latent uniform version of the input images, which represents an internally enhanced version of the image. (2) Our method can then be used iteratively to refine the output, since the output of the method is the improved version of the input. Previous methods were based on intensity probabilities per pixel, which are hard to optimize iteratively. (3) In our approach, we make a distinction between the handling of the degradations and the handling of the binarization. The neural network is trained to correct degradations, while the final binarization is achieved using the efficient Otsu global-threshold method.

The rest of this paper is organized as follows. Section 2 provides a brief summary of a selection of related works. The proposed method is presented in Section 3, and the experimental results are reported in Section 4. Finally, Section 5 provides our conclusions and recommendations for future work.

2. Related work

Binarization is a classic research problem for document analysis, and many document binarization methods have been proposed over the past two decades in the literature. The aim is to convert each pixel in a document image into either text or background. The most popular and simple method is the Otsu method [19], which is a nonparametric and unsupervised method of automatic threshold selection for gray-scale image binarization. It selects the global threshold based on the gray-scale histogram without any a priori knowledge, thus the computational complexity is linear. The Otsu method works very well on uniform and clean images, while producing poor results on degraded document images with nonuniform backgrounds. In order to solve this problem, local adaptive threshold methods have been proposed, such as Sauvola [22], Niblack [23], Pai [24] and AdOtsu [25,26]. These methods compute the local threshold for each pixel based on local statistical information, such as the mean and standard deviation of a local area around the pixel. It should be noted that binarization is not always the goal. Methods such as Otsu can also be used for strong contrast enhancement.

Although the global or local threshold methods mentioned above are very efficient, their results are still not satisfactory for highly degraded and poor quality document images. Therefore, document enhancement methods are usually used in the form of preprocessing in order to remove degradations or noise in document images. Several image-processing techniques, such as the mathematical morphological operator and region-growing method, are used in [27] for document enhancement and binarization. Gatos et al. [28] use a Wiener filter to estimate the background surface that is involved in the final threshold computation. Similarly, in [29], the background surface is estimated by a robust regression method, and the document is binarized by a global thresholding operation. Su et al. [30] propose an adaptive con-

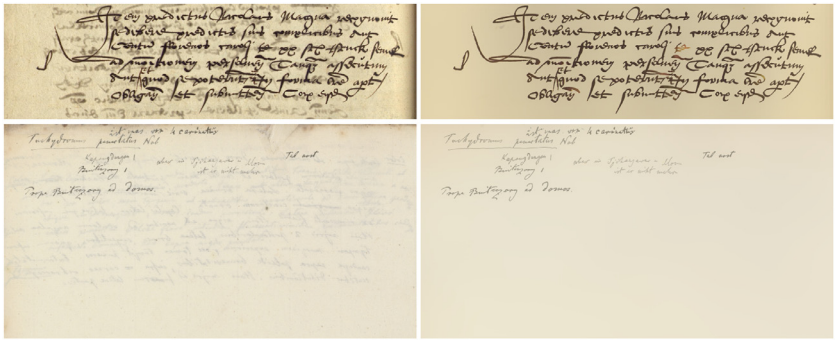


Fig. 2. Demonstration of document image enhancement on two images from the Monk system [3]. The first column shows the original degraded images, and the second column shows the corresponding enhanced images resulting from the proposed method.

trast map for text edge detection, and the local threshold is estimated based on the mean and standard deviation of pixel values on the detected edges in a local region. Nafchi et al. [31] introduce a robust phase-based binarization method that involves image denoising coupled with phase preservation. For bleed-through correction on degraded documents, a new variational model is introduced in [6] based on wavelet shrinkage or a time-stepping scheme. A patch-based, non-local restoration and reconstruction method is proposed in [32] for degraded document enhancement. In [10], a new conditional random field (CRF)-based method [33] is proposed to remove the bleed-through from the degraded images. The bio-inspired model using the off-center ganglion cells of the human vision system is used in [34] for document enhancement and binarization. All the methods mentioned above use traditional techniques for document enhancement, and each of them can only handle a certain type of degradation in document images.

Other a priori knowledge of text is also exploited for binarization, such as the edge pixels extracted by edge detectors. For example, the Canny edge detector is used to extract edge pixels in [35], and then the closed image edges are considered as seeds to find the text region. The transition pixel that is a generation of the edge pixel is introduced in [36] and is computed based on the intensity differences in small neighbor regions, and the statistical information of these pixels is used to compute the threshold. In [37], structural symmetric pixels around strokes are used to compute the local threshold. Howe [38] proposes a promising method that can tune the parameters automatically, with a global energy function as a loss, which incorporates edge discontinuities (the Canny detector is used).

Convolutional neural networks achieve good performance in various applications and can also be applied to document analysis. For example, the winner of the recent DIBCO event [39] uses the U-Net convolutional network architecture for accurate pixel classification. In [16], the fully convolutional neural network is applied at multiple image scales. Deep encoder-decoder architecture is used for binarization in [20,21]. A hierarchical deep supervised network is proposed in [14] for document binarization, which achieves state-of-the-art performance on several benchmark data sets. In [40], the Grid Long Short-Term Memory (Grid LSTM) network is used for binarization. However, its performance is lower than with Vo's method [14].

3. Proposed method

In this section, the problem of document enhancement is discussed, based on iterative deep learning. We first present the formulation of learning degradations for document enhancement, and then the structure of CNN, used for evaluation of the proposed model, is introduced.

3.1. Problem formulation

An original clean or uniform document (ground truth) is assumed to be degraded by various degradations, such as bleed-through or other artifacts. In the image enhancement formulation, the value of each pixel in the degraded images is supposed to be the sum of the original value and the degraded value, which can be expressed by:

$$\mathbf{x} = \mathbf{x}_u + \mathbf{e} \tag{1}$$

where $\mathbf{x}$ is the degraded image, $\mathbf{x}_u$ is the latent clean or uniform image, and $\mathbf{e}$ is the degradation. The probability density of the $\mathbf{e}$ depends on the type of degradation. Fig. 3 gives a visual example of this model.

Most methods for historical document analysis require a clean or uniform image $\mathbf{x}_u$ as input to extract the text edges or contours.

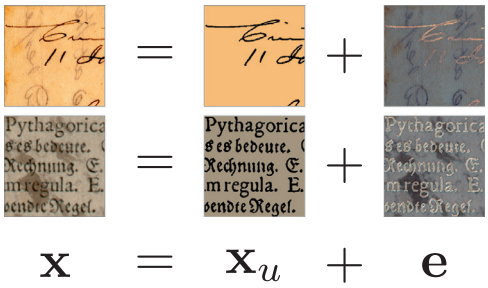


Fig. 3. Schematic description of the proposed degradation model. A degraded pattern $\mathbf{x}$ in the degraded image is assumed to be the sum of an ideal (uniform) pattern $\mathbf{x}_u$ and the degradation $\mathbf{e}$.

Therefore, recovering the clean image $\mathbf{x}_u$, given the degraded image $\mathbf{x}$, is a classic document-enhancement problem. When a uniform $\mathbf{x}_u$ is available, document binarization is quite simple, which can be computed by a global threshold, such as Otsu [19].

The Convolutional Neural Network (CNN) [11] has been successfully used in image classification [12], but it can be applied to document binarization as well [13,14]. However, traditional methods directly apply the CNN to the degraded image $\mathbf{x}$ to compute the binary image $\mathbf{x}_b$ by:

$$\mathbf{x}_b = \mathrm{CNN}(\mathbf{x}) \tag{2}$$

which in fact requires the CNN to implicitly learn the latent uniform image $\mathbf{x}_u$, the degradation $\mathbf{e}$, and also the threshold on the latent uniform image $\mathbf{x}_u$.

In this paper, we train the neural network to predict the uniform image of the input by:

$$\mathbf{x}_u = \mathrm{CNN}(\mathbf{x}) + \mathbf{x} \tag{3}$$

Here the function $\mathrm{CNN}(\mathbf{x}) = -(\mathbf{x} - \mathbf{x}_u) = -\mathbf{e}$ represents the degradations (negative), that is, the difference between the degraded and clean images, which has been learned by the neural network. If the input $\mathbf{x}$ is a uniform and clean image, the neural network does not need to learn any information, and the output of the neural network is close to zero. Since the output $\mathbf{x}_u$ is the improved version of the input $\mathbf{x}$, it is possible to improve the output $\mathbf{x}_u$ iteratively by using the neural network if we set $\mathbf{x} = \mathbf{x}_u$ in the next iteration.

When the uniform $\mathbf{x}_u$ image is obtained after several iterations, the binarization map can be computed by:

$$\mathbf{x}_b = \mathcal{B}(\mathbf{x}_u) \tag{4}$$

where $\mathcal{B}$ can be any existant binarization method or learned neural network. Because the $\mathbf{x}_u$ is the enhanced and clean image, a simple and efficient method can be used for binarization, such as Otsu's global threshold [19].

Our new reformulation proposed in Eq. (3) is motivated by the residual learning [41]. However, the proposed method applies the CNN directly to the input image, which allows the neural network to learn degradations and correct the degraded images iteratively. The CNN structure in Eq. (3) can be any neural network, including the residual network [41].

The proposed model has the following advantages: (1) The neural network only learns the degradation of the image, without fitting the latent uniform images, such as the distribution of the background. (2) The proposed method has a new intermediate output $\mathbf{x}_u$, which can be considered as the enhanced image version of the degraded input $\mathbf{x}$. (3) The model can be seamlessly integrated with other methods by Eq. (4). For example, any existing binarization method can be applied to the estimated uniform image $\mathbf{x}_u$ learned by the neural network.

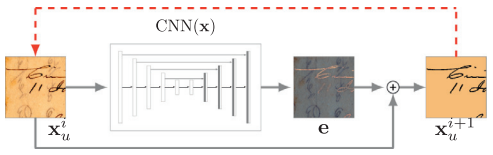


Fig. 4. The recurrent refinement (RR) diagram of the i-th iteration. The red dashed line denotes that the output of the neural network can be used as input for iterative fine tuning with different iterations. $\mathbf{x}_u^0 = \mathbf{x}$ is the original degraded image at the beginning when $i = 0$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The proposed model learns the uniform image $\mathbf{x}_u$ directly from the original image. However, the learned $\mathbf{x}_u$ might not be perfect; it can also be considered as a degraded image $\mathbf{x}$ if the network does not provide good results. Thus, the learned $\mathbf{x}_u$ can be refined or enhanced recursively if we set $\mathbf{x} = \mathbf{x}_u$ for the neural network.

If we obtain the $\mathbf{x}_u$ from the neural network, there are two ways to refine it iteratively: (1) feed it into the same neural network for fine enhancement, referred to as "Recurrent Refinement (RR)," and (2) train a new network (with the same or different structures), referred to as "Stacked Refinement (SR)." The recurrent refinement (RR) method is defined as:

$$\mathbf{x}_u^i = \mathrm{CNN}(\mathbf{x}_u^{i-1}) + \mathbf{x}_u^{i-1} \tag{5}$$

where $\mathbf{x}_u^i$ is the i-th output of the neural network and $\mathbf{x}_u^0 = \mathbf{x}$, which is the original degraded image. Note that there is only one neural network that is trained to refine the results iteratively. Fig. 4 shows the diagram of the RR framework. The advantage of the RR method is that, once the neural network is trained, it can be used iteratively with many iterations. However, this also requires the neural network to learn different levels of degradations in document images. For example, it needs to remove noise from the background and recover the weak ink trace on the text region using the same network.

The stacked refinement (SR) method is defined as:

$$\mathbf{x}_u^i = \mathrm{CNN}_i(\mathbf{x}_u^{i-1}) + \mathbf{x}_u^{i-1} \tag{6}$$

which is similar to Eq. (5), except that the new network CNN_i is trained during the i-th iteration to refine the input $\mathbf{x}_u^{i-1}$, which is the output of the $(i-1)$-th iteration on the CNN_{i-1} neural network. Fig. 5 gives an example of two stacked neural networks with the same structure. However, the neural network structure can also be different in different iterations. The SR method is better than the RR method, because the new network is trained iteratively to learn the degradations in different levels. For example, the neural network can learn the distribution of the background in the first iteration and the distribution of ink traces in the second (iteration). Therefore, background-noise removing and ink-trace recovering can be performed in different networks.

Ideally, the RR and SR methods can be mixed. For example, the neural network of the RR method can also be a stack of networks used in the SR method, which is trained iteratively. However, due to time and memory costs, we will evaluate the proposed RR and SR methods separately in this paper.

3.2. Network architectures

Although any neural network can be used, in this paper, we will adopt the basic U-Net [18] to learn degradations in historical documents, similar to image segmentation. The architecture consists of two paths: a contracting path and an expansive path. In the contracting path, there are five convolutional layers with the kernel size 3×3, each followed by a leaky-ReLU [42] ($\lambda = 0.25$) and a 2×2 max. pooling layer with stride 2. In the expansive path, the deconvolutional operation is used to upsample the feature maps, and then it is concatenated with the corresponding feature maps on the contracting path, followed by a convolutional layer. The filter numbers in five convolutional layers are set to [16,32,64,128,256], respectively. The output layer has the same size as the input image, which makes the additive operation possible (shown in Eq. (3)).

The network is trained by minimizing the following loss function in each iteration:

$$L^i = \frac{1}{n} \sum |\mathbf{x}_t - \hat{\mathbf{x}}_u^i| \tag{7}$$

where n is the number of pixels in the image, $\mathbf{x}_t$ is the ground truth, and $\hat{\mathbf{x}}_u^i$ is the prediction from the neural network with the input $\mathbf{x}_u^i$ in the i-th iteration ($\mathbf{x}_u^0 = \mathbf{x}$, which is the original degraded image at the beginning). The network in each iteration is trained using the loss defined in Eq. (7), with the degraded image and the uniform ground truth $\mathbf{x}_t$. The network of the RR and SR models can be trained separately and jointly on each iteration. In this paper, we will train the network jointly, and the combined loss is defined as

$$L_{total} = \frac{1}{m} \sum_i L^i \tag{8}$$

where m is the number of iterations and L^i is the loss on the i-th iteration, which is defined in Eq. (7).

4. Experiments

In this section, we will present the experimental results of the proposed methods for document enhancement and binarization. The training data sets are constructed based on several public benchmark data sets. We will also introduce a new bleed-through data set, called the Monk Cuper Set (MCS), where the historical documents are collected from the Cuper book collection of the Monk system [3].

4.1. Dataset

There are several public data sets for document binarization, such as the (H-)DIBCO data sets, which are used for document binarization competitions. Similar to practice in [14], we select images on DIBCO 2009 [43], H-DIBCO 2010 [44], and H-DIBCO 2012 [45] for training. The training set also includes documents on the Bickely-diary dataset [46], PHIDB [47], and the Synchromedia Multispectral dataset [48]. The documents on DIBCO 2011 [49], DIBCO 2013 [50], H-DIBCO 2014 [51], and H-DIBCO 2016 [52] are selected for evaluation. Four evaluation metrics, which are used in

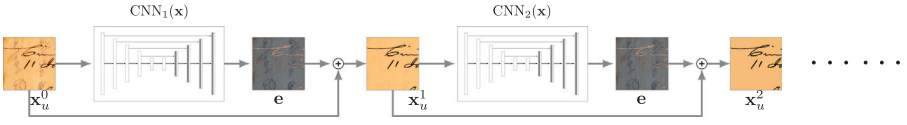


Fig. 5. The stacked refinement (SR) diagram. The neural networks are stacked together to refine the results. Note that different neural networks with the same structure (CNN_1, CNN_2, ...) are trained in this example.

Fig. 6. Training samples (red box) with their corresponding ground-truth (blue box) images. Each pixel in the ground truth is the average of pixels with the same label (text or background) within the patch. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the (H-)DIBCO contests, are adopted in this section for quantitative evaluation and comparison, including the F-measure, pseudo F-measure (F_{ps}), distance reciprocal distortion metric (DRD), and the peak signal-to-noise ratio (PSNR).

Following contest reports [43–45,50–52], these evaluation metrics are defined as follows:

1. F-measure (FM):

$$FM = \frac{2 \times Recall \times Precision}{Recall + Precision} \tag{9}$$

where $Recall = \frac{TP}{TP+FN}$, $Precision = \frac{TP}{TP+FP}$, TP, FP, FN denote the True positive, False position, and False Negative values, respectively.

2. pseudo F-measure (F_{ps}):

$$F_{ps} = \frac{2 \times pRecall \times Precision}{pRecall + Precision} \tag{10}$$

where $pRecall$ is the percentage of the skeletonized ground-truth image described in [44].

3. distance reciprocal distortion metric (DRD):

$$DRD = \frac{\sum_k DRD_k}{NUBN} \tag{11}$$

where DRD_k is the distortion of the k-th flipped pixel, which is calculated using a 5×5 normalized weight matrix, and where $NUBN$ is the number of the non-uniform 8×8 blocks in the ground-truth image (see details in [51]).

4. peak signal-to-noise ratio (PSNR):

$$PSNR = 10\log\left(\frac{C^2}{MSE}\right) \tag{12}$$

where $MSE = \frac{\sum_{x=1}^{M}\sum_{y=1}^{N}(I_{bin}(x,y)-\hat{I}_{bin}(x,y))^2}{MN}$, C denotes the difference between the text and background.

We will also construct a new data set for evaluation, the Monk Cuper Set (MCS), which contains 25 pages sampled from real historical collections. The documents in this set have heavy bleed-through degradations and a textural background, making them very difficult for information retrieval by a computer and even more difficult for end-users to read. Several examples are shown in Fig. 9. This data set is available for academic usage on the author's website.

4.2. Implementing details

Data preparation. We trained our networks with small image patches sampled from the document images with a sliding window. The basic patch size was set to 256×256 (as suggested in [21]). Data augmentation is very important for boosting the performance of the neural network, so we also applied augmentation methods (scale and rotation) to create more training samples. For scale augmentation, we sampled patches with the scale factor {0.75,1.25,1.5}, based on the input of the neural network, and resized them to 256×256. For rotation augmentation, we rotated each patch with a rotation angle 270. Overall, more than 120,000 training patches were created for training.

Ground-truth construction. Since the output of the neural network consists of the uniform images of the input, each pixel value on the ground-truth image is computed as the average pixel value with the same label within the patch. The text and background label are obtained from the ground truth of the binary maps. For example, for the patches that do not contain any text strokes or ink traces, the ground truth is the average image of the patch, which is helpful for removing noise in the background regions of document images. Fig. 6 shows the training samples used in this paper.

Training. The training batch size was set to 5 due to the limitation of the memory. The learning rate was set to 10^{-4}, and the number of training iterations was 110,000. The system ran on a PC platform with a single GPU (NVIDIA GTX 960 with 4G memory).

4.3. Document enhancement

Given an image, the patches with the same size as the training patches were sampled with a sliding window strategy. The values of each pixel in the enhanced image are the average values of the overlapping patches computed from the trained neural networks. Fig. 7 shows a visual example of two documents (in RGB color space) on the DIBCO 2013 data set, with different iterations using the SR and RR methods. It shows that with more iterations,

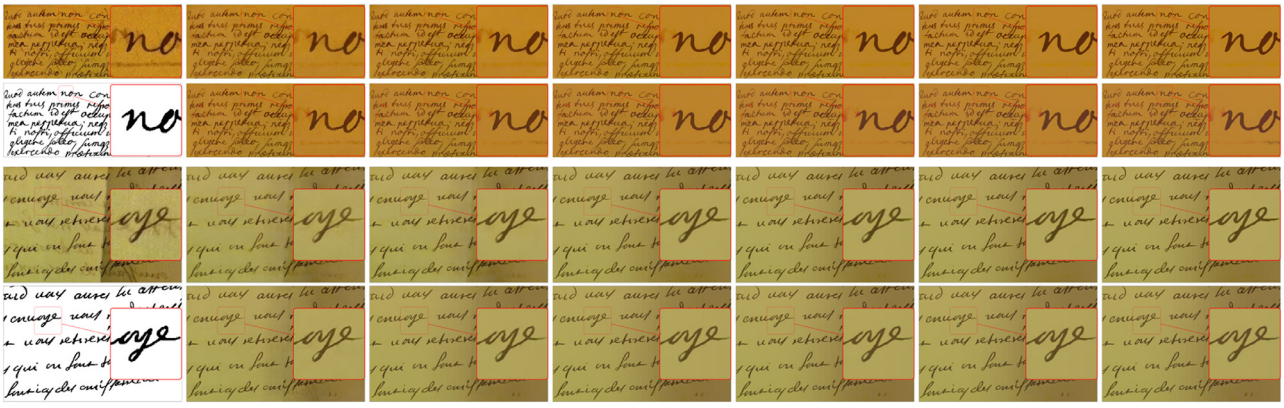


Fig. 7. Examples of enhancement with different iterations of the SR (top row) and RR (bottom row) methods. The first column shows the original images (top) and the corresponding binary maps (bottom row). Images from the second column to the last are the corresponding results of i-iterations where $i = 1, 2, \ldots, 6$.

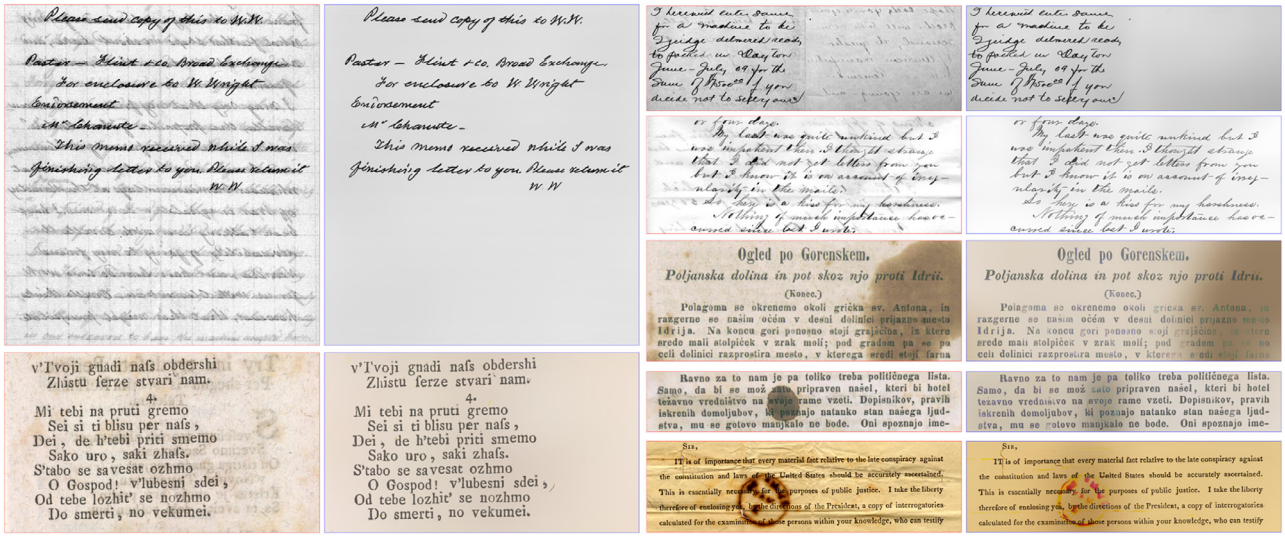


Fig. 8. Examples of enhancement document based on the DIBCO2013 data set and using the SR method.

Fig. 9. Examples of enhancement document on the challenging MCS data set and using the SR method.

the bleed-through and noise on the background are removed and the ink traces are enhanced.

Figs. 8 and 9 show results of the enhancement documents computed by the SR method and based on the DIBCO 2013 and MCS data sets, respectively. From the figures we can see that: (1) The outputs are very clean. The noise and bleed-through degradations on the background have been removed. (2) The large smears in the document images could not be removed completely but could be smoothed out because the input of the neural network was a small patch and the ground truth of the proposed method was constructed based on this small patch, rather than the global images.

In order to quantitatively evaluate the enhancement performance and following [34], we applied two simple threshold methods to the original and the enhanced images to compute the binary maps: Otsu's global threshold [19] and Sauvola's local threshold [22]. Tables 1 and 2 show the increased performance of the binarization maps computed based on the original images (OI) and the enhanced images using the proposed SR and RR methods on the DIBCO 2013 and MCS datasets, respectively. From these two tables, we can see that significant improvement is achieved by using the proposed deep enhancement methods as a preprocessing step. Particularly the use of Sauvola's threshold along with SR method offers the best performance on the two data sets; since our model works on small patches, the background of the whole image does not become uniform.

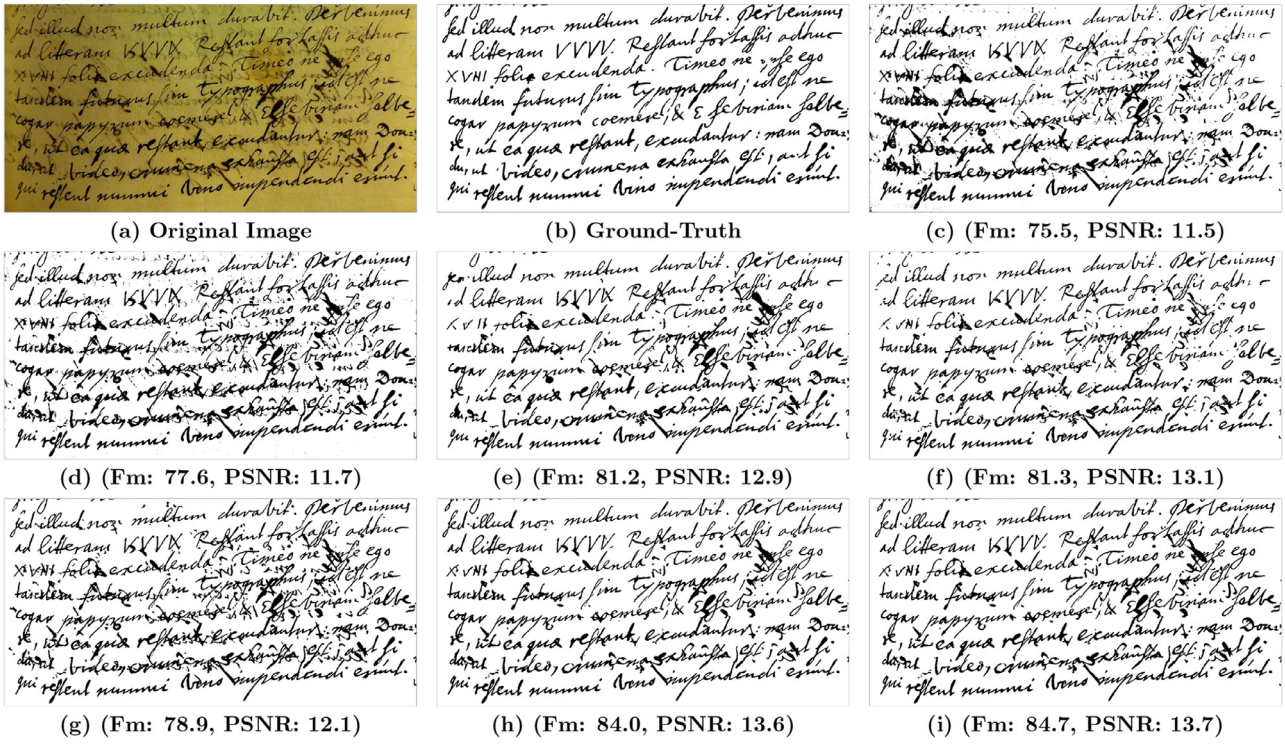
Table 3 shows the performance of different methods in the MCS data set. Sauvola's threshold, based on the enhanced images produced by the SR method, provides a better performance than other traditional methods in this data set. Fig. 10 presents the binary

Table 1
Performance of binarization methods in the original and enhanced images from the DIBCO 2013 data set. Bracketed values show the performance improvements of the binarization results in the enhanced images (using SR and RR methods) as compared to the results in the original images (OI).

Methods	F-measure	F_{ps}	PSNR	DRD
OI-Otsu [19]	80.01	82.82	16.62	11.00
SR-Otsu	90.00(+9.99)	91.68(+8.86)	20.25(+3.63)	6.71(−4.29)
RR-Otsu	88.90(+8.89)	91.19(+8.37)	19.62(+3.00)	7.07(−3.39)
OI-Sauvola [22]	81.23	83.55	16.60	11.39
SR-Sauvola	91.90(+10.67)	93.79(+10.24)	20.65(+4.05)	2.60(−8.79)
RR-Sauvola	90.48(+9.35)	93.63(+10.08)	19.97(+3.37)	2.91(−8.48)

Table 2
Performance of other binarization methods in the original and enhanced images from the MCS data set. Bracketed values show the performance improvements of the binarization results in the enhanced images (using SR and RR methods) as compared to the results in the original images (OI).

Methods	F-measure	F_{ps}	PSNR	DRD
OI-Otsu [19]	69.28	70.51	11.80	33.96
SR-Otsu	82.77(+13.49)	85.80(+15.29)	15.29(+3.49)	11.32(−22.64)
RR-Otsu	79.80(+10.52)	82.31(+11.80)	14.54(+2.74)	15.84(−18.12)
OI-Sauvola [22]	75.84	76.85	13.08	21.54
SR-Sauvola	87.01(+11.17)	89.86(+13.01)	16.19(+3.11)	6.07(−15.47)
RR-Sauvola	86.71(+10.87)	89.68(+12.83)	16.05(+2.97)	6.03(−15.51)



(a) Original Image (b) Ground-Truth (c) (Fm: 75.5, PSNR: 11.5)
(d) (Fm: 77.6, PSNR: 11.7) (e) (Fm: 81.2, PSNR: 12.9) (f) (Fm: 81.3, PSNR: 13.1)
(g) (Fm: 78.9, PSNR: 12.1) (h) (Fm: 84.0, PSNR: 13.6) (i) (Fm: 84.7, PSNR: 13.7)

Fig. 10. Visual example with the F-measure (**Fm**) and **PSNR** metric values of the binarization results of one document applied to the MSC data set and produced by different methods: (a) original image, (b) ground truth, (c) Otsu [19], (d) Sauvola [22], (e) Howe [38], (f) Su [53], (g) Jia [37], (h) SR-Otsu, and (i) SR-Sauvola.

results of one document on the MCS data set using different methods. Our proposed method provides the best visual quality and the values of evaluation metrics (F-measure and PSNR).

4.4. Document binarization

In this section, we will first describe how to compute the binary maps, given the enhanced images. Then, we will present the performance of the document binarization in three benchmark datasets: DIBCO 2011, H-DIBCO 2014, and H-DIBCO 2016.

4.4.1. Binarization

Given the enhanced images produced by the trained neural networks, the existing method can be directly applied to compute the binary maps, such as Otsu [19], which is very simple and efficient. Fig. 11 shows Otsu's performance based on the enhanced images produced by the proposed RR and SR methods, with different iterations on the DIBCO 2011 data set. From the figure, we can see that the trained neural network can enhance the document iteratively and that performance increases dramatically during the first three iterations. After that, performance improves slightly with more iterations.

Table 3
Comparison of different algorithms applied to the MSC data set.

Methods	F-measure	F_{ps}	PSNR	DRD
Otsu [19]	69.3	70.5	11.8	34.0
Sauvola [22]	75.8	76.9	13.1	21.5
Howe [38]	85.6	89.1	15.8	6.4
Su [53]	82.8	87.4	15.2	16.8
Jia [37]	85.4	88.7	15.8	7.1
SR-Sauvola	**87.0**	**89.9**	**16.2**	**6.1**

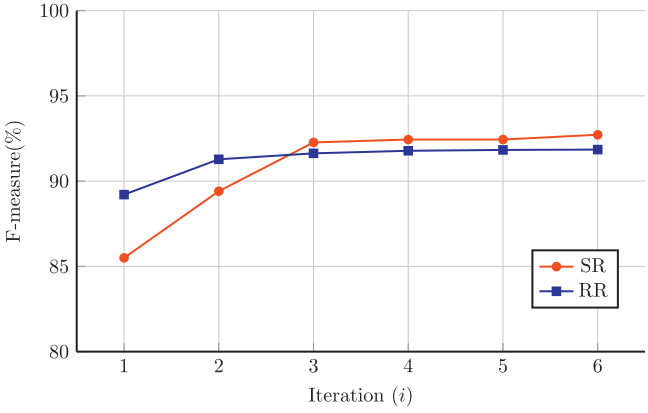


Fig. 11. The performance of 'Otsu's binarization performance/results' based on the enhanced images produced by the proposed RR and SR methods on the DIBCO 2011 data set, with different iterations i (i=1,2,...,6).

In order to use Otsu's global threshold, the background of the enhanced images should be uniform. However, since we were using a small patch as input, the enhanced images were not globally uniform since the background was nonuniform, as can be seen in Fig. 8. To handle this problem, we proposed several refinement steps to improve the performance of the Otsu threshold, based on the enhancement images and using the proposed methods.

Uniform+Otsu: We rescaled the output of each patch to a range of (0, 255) if it contained ink traces, which means the background was set to values increasing to 255 and the text to values decreasing to 0. Otherwise, we used the background value. Sauvola's threshold was used to determine whether the image patch from the output of the trained neural networks contained ink traces or not. Fig. 12 shows an example of the Otsu results, with and without using locally uniform results for a document with a nonuniform background. The locally uniform results were very helpful for handling nonuniform background documents when using Otsu's global threshold.

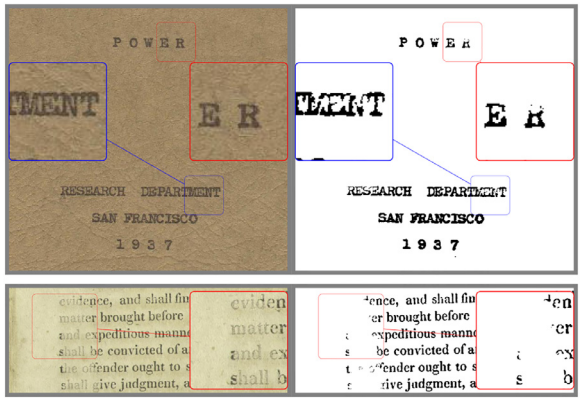


Fig. 13. Two cases of failure in the DIBCO 2011 data set while using the proposed DeepOtsu (SR) method. The method misses the weak strokes in the binary maps.

MS+Uniform+Otsu: We also sampled the patches with different scales on the test images (scale factors 0.75, 1.25, and 1.5, based on the size of neural network input), and resized them into 256 × 256. The binary map is the average of multiple scale outputs from the neural network.

Fusion+MS+Uniform+Otsu: Our proposed method was able to iteratively refine the output of the degraded input. However, weak and thin ink traces ran the risk of being lost at the end of the iteration. Integration of the output of each iteration improved performance. In this paper, we averaged the output of each iteration (six in total) and then used the Otsu to compute the binary maps.

Table 4 shows the results from the DIBCO 2011 data set. From the table, we can see that using locally uniform results in multiscale modeling can improve the performance of both RR and SR methods. Combining the outputs from different iterations can provide a slightly better performance for the SR method and a slightly worse performance in terms of the F-measure for the RR method, due to the fact the SR method contains more convolutional layers than the RR method. We also compared the results of Sauvola's threshold [22] based on the enhanced images computed with different refinements in Table 4. Sauvola's local threshold resulted in a slightly worse performance than Otsu's global threshold [19]. In the following section, we will provide the results for both the SR and RR method, using all the refinement steps (Fusion+MS+Uniform+Otsu), referred to as DeepOtsu.

4.4.2. Performance in the (H)-DIBCO benchmark data set

In this section, we will compare the performance of the proposed methods with other binarization algorithms using three benchmark data sets.

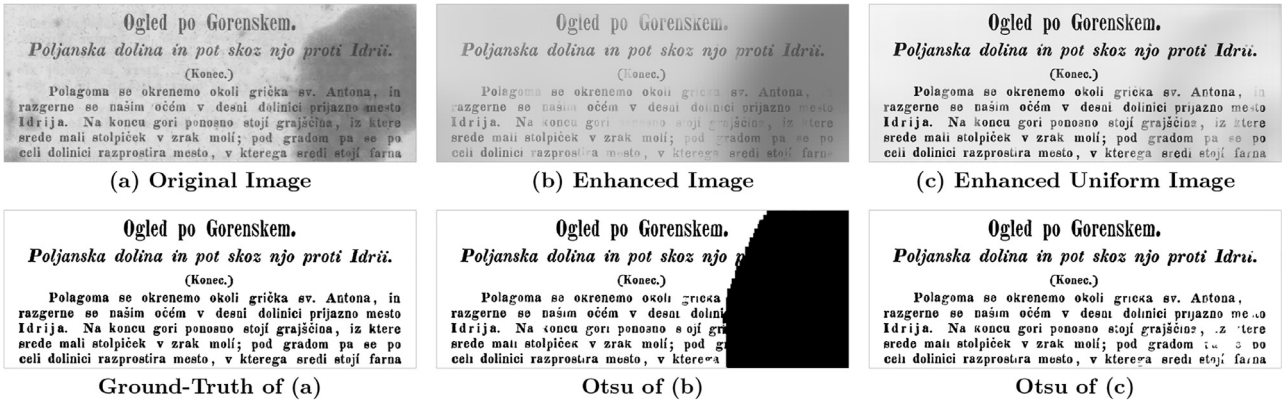


Fig. 12. Example of locally uniform results for the sample document image (PR04) on the DIBCO 2013 data set.

Table 4
Performance of binarization of the CNN output image with different refinement steps in the DIBCO 2011 data set.

Methods	SR				RR			
	F-measure	F_{ps}	PSNR	DRD	F-measure	F_{ps}	PSNR	DRD
Otsu	92.7	95.6	19.7	2.2	91.9	94.9	19.1	2.6
Uniform+Otsu	92.9	95.7	19.9	2.1	92.4	95.3	19.4	2.4
MS+Uniform+Otsu	93.1	95.4	20.0	2.0	93.0	95.3	19.6	2.2
Fusion+MS+Uniform+Otsu	93.4	95.8	19.9	1.9	92.8	95.6	19.5	2.2
Sauvola	90.9	93.8	19.6	2.6	90.8	94.6	18.9	2.8
Uniform+Sauvola	93.1	93.7	19.6	2.2	92.3	92.6	19.1	2.7
MS+Uniform+Sauvola	92.2	92.2	19.1	2.5	91.9	91.7	18.7	2.7
Fusion+MS+Uniform+Sauvola	92.4	92.4	19.1	2.4	92.3	92.4	19.0	2.6

Fig. 14. Examples of the enhanced and binarization results of sample document images from the H-DIBCO 2014 data set. The left column shows the original images, the middle column shows the enhanced images using the proposed SR method, and the right column shows the binarization maps based on the enhanced images.

Table 5 shows the performance in the DIBCO 2011 data set. Otsu's threshold, based on the enhanced images computed by the SR method, offers the best performance in terms of the F-measure and DRD metrics, which shows that our proposed method produces binarization maps with a less distorted visual quality. The method proposed in [14] also uses a hierarchical neural network to predict the binary maps directly from the degraded images, and it provides a slightly better performance in terms of F_{ps} and PSNR. Fig. 13 provides two examples of failure within the DIBCO 2011 data set while using our proposed method; it misses the weak ink strokes because, in the training set, the ground truth of the weak ink strokes is also very weak since it is computed using the average ink pixels in a local patch. Thus, these text regions are missed in the final binary maps computed by Otsu's threshold. This problem might be resolved by training the network with more iterations, using training samples of weak ink strokes.

Table 6 shows the performance of different methods in the historical document competition, i.e. the H-DIBCO 2014 data set. Fig. 14 shows the enhanced and corresponding binarization maps computed by the proposed DeepOtsu (SR) method for this data set. From Table 6, we can see that the performance of our proposed DeepOtsu method is comparable to that of Vo's method [14], which also uses deep learning. The performance of the traditional methods, such as Howe [38], Su [53], and Jia [37], is comparable to that of the deep learning methods, such as Vo [14], which indicates that the binarization problem in the H-DIBCO 2014 data set is less challenging than in other data sets. The best performance is achieved using Vo's method [14], which integrates the outputs of the binary

Table 5
Comparison of different algorithms applied to the DIBCO 2011 data set.

Methods	F-measure	F_{ps}	PSNR	DRD
Otsu [19]	82.1	84.8	15.7	9.0
Sauvola [22]	82.1	87.7	15.6	8.5
Howe [38]	91.7	92.0	19.3	3.4
Su [53]	87.8	90.0	17.6	4.8
Jia [37]	91.9	95.1	19.0	2.6
Vo [29]	88.2	90.3	20.1	2.9
Vo [14]	93.3	**96.4**	**20.1**	2.0
DeepOtsu(RR)	92.8	95.6	19.5	2.2
DeepOtsu(SR)	**93.4**	95.8	19.9	**1.9**

Table 6
Comparison of different algorithms applied to the H-DIBCO 2014 data set.

Methods	F-measure	F_{ps}	PSNR	DRD
Otsu [19]	91.7	95.7	18.7	2.7
Sauvola [22]	84.7	87.8	17.8	2.6
Howe [38]	96.5	97.4	22.2	1.1
Su [53]	94.4	95.9	20.3	1.9
Jia [37]	95.0	97.2	20.6	1.2
Vo [14]	**96.7**	**97.6**	**23.2**	**0.7**
DeepOtsu(RR)	94.3	96.3	20.9	1.9
DeepOtsu(SR)	95.9	97.2	22.1	0.9

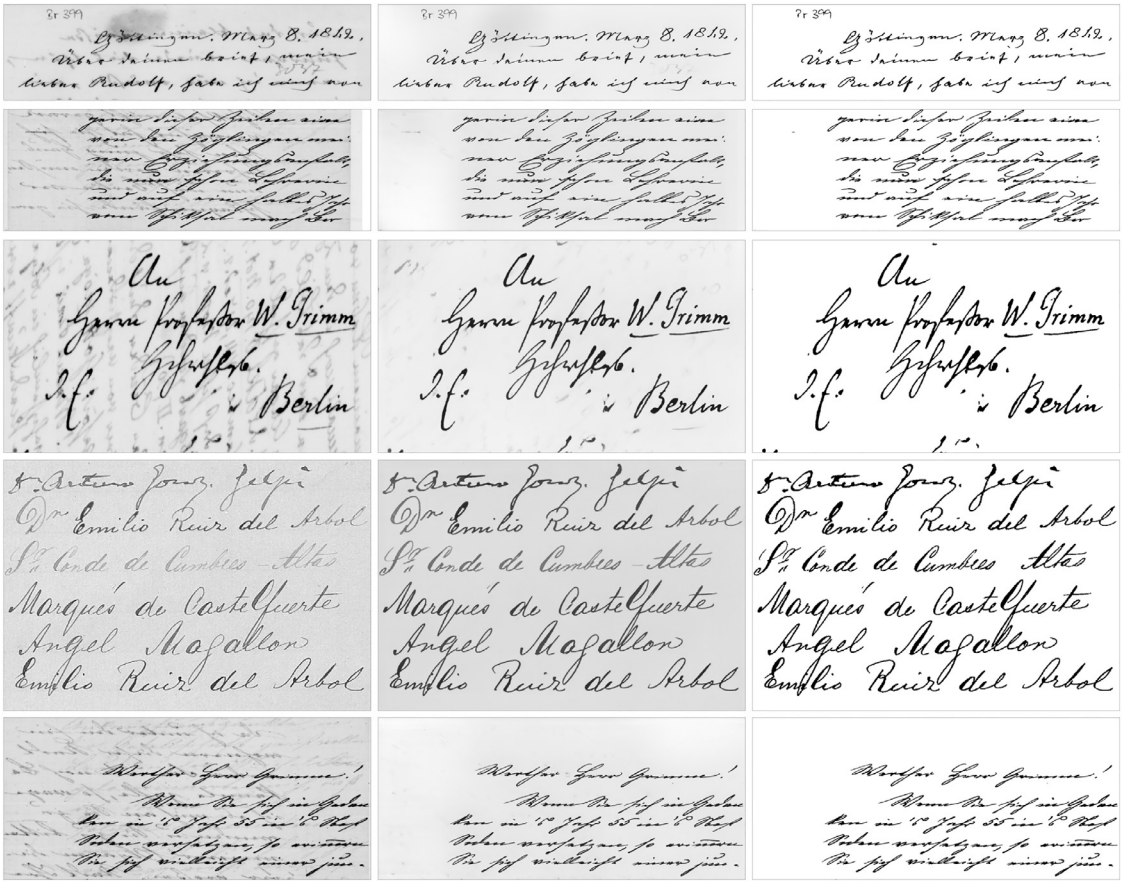


Fig. 15. Examples of the enhanced and binarization results of sample document images from the H-DIBCO 2016 data set. The left column shows the original images, the middle column shows the enhanced images using the proposed SR method, and the right column shows the binarization maps based on the enhanced images.

predictions of three different networks and computes the final binary map using a local and global threshold that is learned based on a separate data set.

Table 7 presents the performance of different binarization methods in the H-DIBCO 2016 data set. Our proposed method provides better results than the traditional threshold methods and surpasses other deep learning methods, such as the recurrent neural network [40] and the hierarchical deep supervised network [14]. Fig. 15 shows the enhanced and corresponding binarization maps computed by the proposed DeepOtsu (SR) method for this data set, showing that the enhanced images are uniform and clean, without noise and textures on the background.

From the above analysis, one can see that the proposed method works much better on degraded document images with bleed-through and noise, offering an improved version of these documents for visualization. Since the input of our method is a small patch, the large smears cannot be removed completely (Fig. 8). This problem can be resolved, however, by training a neural network with a large input patch. The trained neural network will focus on removing the degradations; the risk, however, is that it will consider thin or weak strokes as degradations (Fig. 13). If reconstruction of thin strokes is needed, this must be reflected in the composition of the training set; in other words, it must contain a sufficient number of such patterns.

Table 7
Comparison of different algorithms applied to the H-DIBCO 2016 data set.

Methods	F-measure	F_{ps}	PSNR	DRD
Otsu [19]	86.6	89.9	17.8	5.6
Sauvola [22]	84.6	88.4	17.1	6.3
Howe [38]	87.5	92.3	18.1	5.4
Su [53]	84.8	88.9	17.6	5.6
Jia [37]	90.5	93.3	19.3	3.9
Vo [29]	87.3	90.5	17.5	4.4
Vo [14]	90.1	93.6	19.0	3.5
Westphal [40]	88.8	92.5	18.4	3.9
DeepOtsu(RR)	90.9	93.9	19.4	3.1
DeepOtsu(SR)	**91.4**	**94.3**	**19.6**	**2.9**

4.4.3. Computing time analysis

The input patch size of the method that we propose is fixed, so that it can be computed in a very efficient way by GPU. The training required for this takes about 24 hours for both the SR and RR method on a single GPU (NVIDIA GTX 960 with 4GB of memory). For testing, the computing time of the neural network for each patch is around 0.02 s, and the processing uniform for binarization takes around 0.0008 s. The patches on one image can be processed in parallel on different GPUs. The training and testing times can be reduced if a more powerful computer system with more GPUs and memory is used.

5. Conclusion

We have proposed a novel model for document enhancement and binarization based on iterative deep learning. Given a small patch sampled from an image, the proposed enhancement model iteratively predicts the uniform image in two possible ways: through recurrent refinement or stacked refinement. The enhanced image produced by the proposed method results in a very good visual experience for end-users since it is

clean, locally uniform, and does not contain any undesirable textures in the background. We evaluated the method that we proposed using real historical collections from the Monk system as well as several public benchmark data sets. The experimental results have demonstrated that our method shows a promising performance.

In this paper, we have used the basic U-Net neural network to learn degradations in document images. More complicated neural networks, such as ResNet [41] and DenseNet [54] could be adopted in future work, as well as DSN, which is used in [14]. In addition, different networks could also be used in the various iterations: A more complicated neural network could perform the initial iterations, while a lighter neural network could be used in the final iterations for fine-tuning.

Acknowledgements

This work has been supported by the Dutch Organization for Scientific Research NWO Digging into data grant 'Global Currents' (Project no. 640.006.015). The authors would like to thank Zhenwei Shi to label the MCS data set.

References

[1] K. Ntirogiannis, B. Gatos, I. Pratikakis, Performance evaluation methodology for historical document image binarization, IEEE Trans. Image Process. 22 (2) (2013) 595–609.
[2] A. Tonazzini, Color space transformations for analysis and enhancement of ancient degraded manuscripts, Pattern Recognit. Image Anal. 20 (3) (2010) 404–417.
[3] T. Van der Zant, L. Schomaker, K. Haak, Handwritten-word spotting using biologically inspired features, IEEE Trans. Pattern Anal. Mach. Intell. 30 (11) (2008) 1945–1957.
[4] S. He, P. Samara, J. Burgers, L. Schomaker, A multiple-label guided clustering algorithm for historical document dating and localization, IEEE Trans. Image Process. 25 (11) (2016) 5252–5265.
[5] M. Stauffer, A. Fischer, K. Riesen, Keyword spotting in historical handwritten documents based on graph matching, Pattern Recognit. 81 (2018) 240–253.
[6] R.F. Moghaddam, M. Cheriet, A variational approach to degraded document enhancement, IEEE Trans. Pattern Anal. Mach. Intell. 32 (8) (2010) 1347–1361.
[7] R. Hedjam, M. Cheriet, Historical document image restoration using multispectral imaging system, Pattern Recognit. 46 (8) (2013) 2297–2312.
[8] R.F. Moghaddam, M. Cheriet, Low quality document image modeling and enhancement, Int. J. Doc. Anal. Recogn. (IJDAR) 11 (4) (2009) 183–201.
[9] M.R. Yagoubi, A. Serir, A. Beghdadi, A new automatic framework for document image enhancement process based on anisotropic diffusion, in: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, IEEE, 2015, pp. 1126–1130.
[10] B. Sun, S. Li, X.-P. Zhang, J. Sun, Blind bleed-through removal for scanned historical document image with conditional random fields, IEEE Trans. Image Process. 25 (12) (2016) 5702–5712.
[11] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[12] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.
[13] J. Calvo-Zaragoza, G. Vigliensoni, I. Fujinaga, Pixel-wise binarization of musical documents with convolutional neural networks, in: International Conference on Machine Vision Applications (MVA), IEEE, 2017, pp. 362–365.
[14] Q.N. Vo, S.H. Kim, H.J. Yang, G. Lee, Binarization of degraded document images based on hierarchical deep supervised network, Pattern Recognit. 74 (2018) 568–586.
[15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.
[16] C. Tensmeyer, T. Martinez, Document image binarization with fully convolutional neural networks, in: Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, 1, IEEE, 2017, pp. 99–104.
[17] S. Xie, Z. Tu, Holistically-nested edge detection, Int. J. Comput. Vis. 125 (1–3) (2017) 3–18.
[18] O. Ronneberger, P. Fischer, T. Brox, U-Net: convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.
[19] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst. Man Cybern. 9 (1) (1979) 62–66.
[20] X. Peng, H. Cao, P. Natarajan, Using convolutional encoder-decoder for document image binarization, in: Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, 1, IEEE, 2017, pp. 708–713.
[21] J. Calvo-Zaragoza, A.-J. Gallego, A selectional auto-encoder approach for document image binarization, Pattern Recognit. 86 (2019) 37–47.
[22] J. Sauvola, M. Pietikäinen, Adaptive document image binarization, Pattern Recognit. 33 (2) (2000) 225–236.
[23] W. Niblack, An introduction to digital image processing, 34, Prentice-Hall Englewood Cliffs, 1986.
[24] Y.-T. Pai, Y.-F. Chang, S.-J. Ruan, Adaptive thresholding algorithm: efficient computation technique based on intelligent block detection for degraded document images, Pattern Recognit. 43 (9) (2010) 3177–3187.
[25] R.F. Moghaddam, M. Cheriet, A multi-scale framework for adaptive binarization of degraded document images, Pattern Recognit. 43 (6) (2010) 2186–2198.
[26] R.F. Moghaddam, M. Cheriet, AdOtsu: an adaptive and parameterless generalization of Otsu's method for document image binarization, Pattern Recognit. 45 (6) (2012) 2419–2431.
[27] Z. Shi, S. Setlur, V. Govindaraju, Image enhancement for degraded binary document images, in: Document Analysis and Recognition (ICDAR), 2011 International Conference on, IEEE, 2011, pp. 895–899.
[28] B. Gatos, I. Pratikakis, S.J. Perantonis, Adaptive degraded document image binarization, Pattern Recognit. 39 (3) (2006) 317–327.
[29] G.D. Vo, C. Park, Robust regression for image binarization under heavy noise and nonuniform background, Pattern Recognit. 81 (2018) 224–239.
[30] B. Su, S. Lu, C.L. Tan, Robust document image binarization technique for degraded document images, IEEE Trans. Image Process. 22 (4) (2013) 1408–1417.
[31] H.Z. Nafchi, R.F. Moghaddam, M. Cheriet, Phase-based binarization of ancient document images: model and applications, IEEE Trans. Image Process. 23 (7) (2014) 2916–2930.
[32] R.F. Moghaddam, M. Cheriet, Beyond pixels and regions: a non-local patch means (NLPM) method for content-level restoration, enhancement, and reconstruction of degraded document images, Pattern Recognit. 44 (2) (2011) 363–374.
[33] D. Song, W. Liu, T. Zhou, D. Tao, D.A. Meyer, Efficient robust conditional random fields, IEEE Trans. Image Process. 10 (24) (2015) 3124–3136.
[34] K. Zagoris, I. Pratikakis, Bio-inspired modeling for the enhancement of historical handwritten documents, in: Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, 1, IEEE, 2017, pp. 287–292.
[35] Q. Chen, Q.-s. Sun, P.A. Heng, D.-s. Xia, A double-threshold image binarization method based on edge detector, Pattern Recognit. 41 (4) (2008) 1254–1267.
[36] M.A. Ramírez-Ortegón, E. Tapia, L.L. Ramírez-Ramírez, R. Rojas, E. Cuevas, Transition pixel: a concept for binarization based on edge detection and gray-intensity histograms, Pattern Recognit. 43 (4) (2010) 1233–1243.
[37] F. Jia, C. Shi, K. He, C. Wang, B. Xiao, Degraded document image binarization using structural symmetry of strokes, Pattern Recognit. 74 (2018) 225–240.
[38] N.R. Howe, Document binarization with automatic parameter tuning, Int. J. Doc. Anal. Recogn. (IJDAR) 16 (3) (2013) 247–258.
[39] I. Pratikakis, K. Zagoris, G. Barlas, B. Gatos, ICDAR2017 competition on document image binarization (DIBCO 2017), in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2017, pp. 1395–1403.
[40] F. Westphal, N. Lavesson, H. Grahn, Document image binarization using recurrent neural networks, in: 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), IEEE, 2018, pp. 263–268.
[41] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[42] A.L. Maas, A.Y. Hannun, A.Y. Ng, Rectifier nonlinearities improve neural network acoustic models.
[43] B. Gatos, K. Ntirogiannis, I. Pratikakis, ICDAR 2009 document image binarization contest (DIBCO 2009), in: Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on, IEEE, 2009, pp. 1375–1382.
[44] I. Pratikakis, B. Gatos, K. Ntirogiannis, H-DIBCO 2010-handwritten document image binarization competition, in: Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on, IEEE, 2010, pp. 727–732.
[45] I. Pratikakis, B. Gatos, K. Ntirogiannis, ICFHR 2012 competition on handwritten document image binarization (H-DIBCO 2012), in: Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on, IEEE, 2012, pp. 817–822.
[46] F. Deng, Z. Wu, Z. Lu, M.S. Brown, Binarizationshop: a user-assisted software suite for converting old documents to black-and-white, in: Proceedings of the 10th annual joint conference on Digital libraries, ACM, 2010, pp. 255–258.
[47] H.Z. Nafchi, S.M. Ayatollahi, R.F. Moghaddam, M. Cheriet, An efficient ground truthing tool for binarization of historical manuscripts, in: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on, IEEE, 2013, pp. 807–811.
[48] R. Hedjam, H.Z. Nafchi, R.F. Moghaddam, M. Kalacska, M. Cheriet, ICDAR 2015 contest on multispectral text extraction (MS-TEx 2015), in: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, IEEE, 2015, pp. 1181–1185.
[49] I. Pratikakis, B. Gatos, K. Ntirogiannis, ICDAR 2011 document image binarization contest (DIBCO 2011), in: Document Analysis and Recognition (ICDAR), 2011 International Conference on, IEEE, 2011, pp. 1506–1510.
[50] I. Pratikakis, B. Gatos, K. Ntirogiannis, ICDAR 2013 document image binarization contest (DIBCO 2013), in: Document Analysis and Recognition (ICDAR), 2013 International Conference on, IEEE, 2013.
[51] K. Ntirogiannis, B. Gatos, I. Pratikakis, ICFHR2014 competition on handwritten document image binarization (H-DIBCO 2014), in: Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on, IEEE, 2014, pp. 809–813.

[52] I. Pratikakis, K. Zagoris, G. Barlas, B. Gatos, ICFHR2016 handwritten document image binarization contest (H-DIBCO 2016), in: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE, 2016, pp. 619–623.
[53] B. Su, S. Lu, C.L. Tan, Binarization of historical document images using the local maximum and minimum, in: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, ACM, 2010, pp. 159–166.
[54] G. Huang, Z. Liu, L. van der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017, pp. 2261–2269.

Sheng He gained a cum laude Ph.D. degree in artificial intelligence from the University of Groningen, the Netherlands, in 2017. From 2017 to 2018, he was a postdoctoral fellow at the University of Groningen. In 2018, he joined Harvard Medical School as a research fellow. He received the Chinese government award for outstanding self-financed students abroad (2016) from the Chinese Scholarship Council. His research interests include handwritten document analysis, deep learning, and medical image analysis.

Lambert Schomaker is Professor in Artificial Intelligence at the University of Groningen and was the director of its AI institute ALICE from 2001 to 2018. His main interests are pattern recognition and machine learning problems, with applications in handwriting recognition problems. He has contributed to over 200 peer-reviewed publications in journals and books (h=17/ISI, h=42/Google Citations). His work is cited in 23 patents. In recent years, his focus has been on continuous-learning systems and bootstrapping problems, where learning starts using very few examples. Prof. Schomaker is a senior member of IEEE, a member of the IAPR, and a member of Dutch research program committees in e-Science (at the NWO), Computational Humanities (at the KNAW), Computational science, and energy (at Shell/NWO/FOM). He received two IBM Faculty Awards (in 2011 and 2012) for the Monk word-retrieval system in historical manuscript collections using high-performance computing.