U-Net-bin: hacking the document image binarization contest

P.V. Bezmaternykh[1,2], D.A. Ilin [1], D.P. Nikolaev [1, 3]
[1] Smart Engines Service LLC, 117312, Moscow, Russia,
[2] Federal Research Center "Computer Science and Control" of RAS, 117312, Moscow, Russia,
[3] Institute for Information Transmission Problems of RAS, 127051, Moscow, Russia

Abstract

Image binarization is still a challenging task in a variety of applications. In particular, Document Image Binarization Contest (DIBCO) is organized regularly to track the state-of-the-art techniques for the historical document binarization. In this work we present a binarization method that was ranked first in the DIBCO`17 contest. It is a convolutional neural network (CNN) based method which uses U-Net architecture, originally designed for biomedical image segmentation. We describe our approach to training data preparation and contest ground truth examination and provide multiple insights on its construction (so called hacking). It led to more accurate historical document binarization problem statement with respect to the challenges one could face in the open access datasets. A docker container with the final network along with all the supplementary data we used in the training process has been published on Github.

Keywords: historical document processing, binarization, DIBCO, deep learning, U-Net architecture, training dataset augmentation, document analysis.

Citation: Bezmaternykh PV, Ilin DA, Nikolaev DP. U-Net-bin: hacking the document image binarization contest. Computer Optics 2019; 43(5): 825-832. DOI: 10.18287/2412-6179-2019-43-5-825-832.

Acknowledgements: The work was partially funded by Russian Foundation for Basic Research (projects 17-29-07092 and 17-29-07093).

Introduction

Image binarization is a procedure that classifies each pixel as a background or as a foreground element. It is commonly used in a wide range of domains, such as machine vision [1], forensics [2], personal identification [3], or document analysis. On the other hand, every domain requires its own set of different properties and peculiarities from the binarization procedure, which results in a variety of developed methods.

In contrast with more general image segmentation approaches, binarization is meaningful if we are not interested in distinguishing of conjunct groups of foreground pixels. Thus, document image binarization is, to some extent, appropriate for separating foreground text objects from document background [4]. This particular kind of binarization is mainly used in three ways: (i) as a preparation step for the following optical character recognition (OCR) procedure, (ii) to reduce the amount of memory required for storing documents in archival systems and online libraries, and (iii) to enhance the image for human perception [5]. The typical sample of document image binarization usage is shown in Fig. 1.

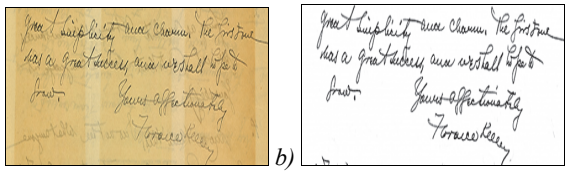


Fig. 1. Sample of document binarization application: (a) source image, (b) binarization result

In this work, we explore a specific application case: historical document image binarization. Historical documents and manuscripts tend to suffer from a wide range of distortions which make binarization an extremely challenging task. In 2009, in order to track state-of-the-art in this domain, the first international Document Image Binarization Contest (DIBCO) [6] was organized in the context of ICDAR conference. Now it takes place regularly and its rules are well established. The organizers define an evaluation methodology and provide it to the participants. They also prepare a benchmarking (testing) dataset which consists of color images with corresponding binary ground truth pixel maps (usually it contains about 10 or 20 images). The key feature of the contest is the unavailability of this dataset for the participants until the end of the competition. It results in inability for the contestants to tune up their algorithms and protects against potential overfitting. Every competition ends with a publication containing brief description and quality measurement for all the proposed methods.

In this work we explain in detail the binarization method submitted to the DIBCO'17 that won in both machine-printed and handwritten categories among 26 evaluated algorithms [7]. We chose a CNN based approach using U-Net architecture [8] because of its ability to process big image patches capturing their contexts. Having explored the provided ground truth from the previous contests and peculiarities of their construction, we describe our understanding of the precise problem statement of DIBCO. We also provide some useful insights on training data preparation and augmentation techniques.

The rest of the work is organized as follows: section I gives an overview of some related work; section II demonstrates the architecture of the neural network and describes training procedures; section III presents experimental results on the DIBCO datasets; section IV contains a discussion about the proposed approach to binarization problem solving and our particular solution applicability.

1. Related work

Among numerous existing binarization methods we firstly need to mention the two classical ones: Otsu [9] and Sauvola [10]. Despite them being rather aged, they are still often used. In particular, the DIBCO organizers use them in their contests as a baseline. The Otsu method belongs to the class of **global binarization** and it is probably the most widely used method of such class in practical applications. Global methods calculate a single pixel intensity threshold for the whole image. In general, these methods cannot be applied directly on images with non-uniform illumination. As a result, a huge variety of original method modifications appeared, such as recursive Otsu application [11], two-dimensional Otsu [12], [13], or document image normalization [14] before global thresholding. Furthermore, background estimation is an important step that helps to prepare an image for the further thresholding [15], [16]. The Sauvola method is a canonical example of **local binarization** methods. It is an extension to the famous Niblack's algorithm [17]. It calculates a threshold for every pixel in the image with respect to its local neighborhood. Most often it is determined by a square window of specific size centered around the processed pixel. To find the local threshold for it, both Niblack and Sauvola methods rely on the usage of two first central moments of pixel values in the window. This window size affects the resulting binarization quality and should be carefully chosen. Quality evaluation of several local methods can be found in [18] and [19]. A number of works are dedicated to automatical estimation of local method parameters. In a recent article [20], a multi-scale Sauvola's modification was presented. Earlier, a multi-window binarization approach was presented [21]. In general, locally adaptive methods produce better results for historical document images. Knowledge of document specific domain can be used for selecting the window size. Text stroke width estimation is a common technique that helps to deal with this problem [16]. In 2012, Howe proposed document binarization with automatic parameter tuning [22].

These classical methods are often applied as subroutines in new binarization algorithms [23]. Another approach is to divide the input image into subregions and select a suitable binarization method for them from a predefined set (e.g., [24], [25]). In [26], [27], the combination of binarization methods is presented.

In recent years, a number of binarization methods based on supervised learning techniques has increased significantly [28 – 31]. They tend to use deep neural networks (mostly CNNs) of different architectures and best of them have already outperformed the classical methods. It means that usage of classical approaches nowadays is reasonable only for tasks with computational restrictions. Contrary, in DIBCO the time limit for a binarization procedure is not imposed, which allows to submit networks with a huge number of neurons, arbitrary depth and architecture. No wonder that among the top six solutions in DIBCO'17 only the deep architectures were presented. Since we had chosen the same approach, our main considered problems were: (i) proper network architecture selection, (ii) sensible training dataset preparation. Each of these problems are discussed below.

2. Approach

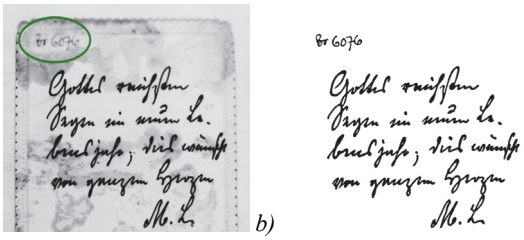
In this section, we describe our vision of historical document image binarization problem, our approach to training data preparation, justification of neural network architecture selection and its training details.

General overview

For the initial training dataset, we used 65 handwritten and 21 machine-printed document images provided by the competition organizers. These images contained not the entire documents but only the cropped regions of interest. All documents were gathered from different sources: archives, old books and their covers, and handwritten letters. Therefore, they did not represent the documents that are used in daily life (e.g., ID cards, bills, etc.). Only Latin-based fonts for both machine-printed and handwritten texts were used.

As a ground truth, binarized version of each image was provided. Although for many practical applications quality measurements can be done rather easily and effectively [32], for this contest an existence of pixel-wise ground truth is essential. To gain a deeper understanding of the DIBCO problems we paid attention to the way of pixel labeling for the most problematic cases.

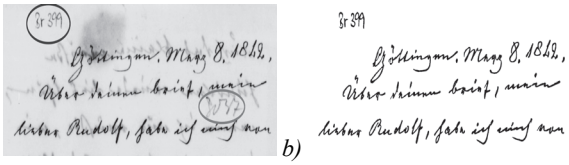
Let's consider few cases. In Fig. 2*a*, a faint handwriting at the top left corner must be classified as a foreground (Fig. 2*b*). It is located outside of the main text area and it differs greatly in brightness.



a) *b)*

Fig. 2. Sample with faint text fragment (outside): (a) Source, (b) Ground truth

In the case in Fig. 3*a*, there is a similar situation in the same corner, but the handwritten number between second and third rows must be classified as a background element despite it has virtually the same gray level (Fig. 3*b*).

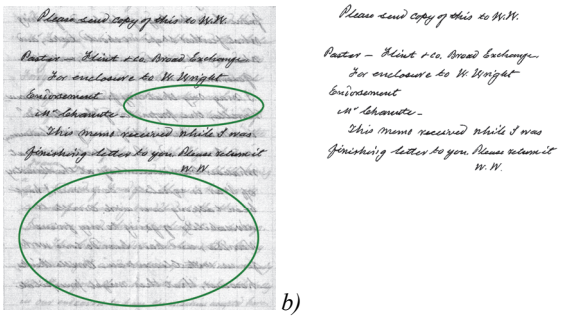


a) *b)*

Fig. 3. Sample with faint text fragment (inside): (a) Source, (b) Ground truth

In general, we assume that when the faint fragments are located next to the main text lines they should be classified as a background. It is especially important in the presence of text lines bleeding through the opposite side of document page and overlapping with the strong lines (Fig. 4*a*). In such case, every pixel should be

segmented very carefully. We also need to determine the situation when all the lines in the region are from the opposite side (Fig. 4*a*, in the bottom).

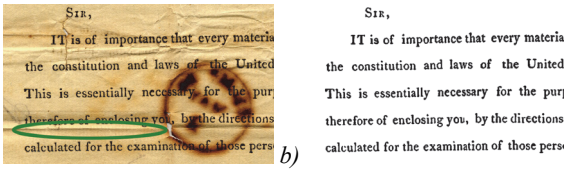


a) *b)*

Fig. 4. Sample with huge "empty" regions: (a) Source, (b) Ground truth

Simple methods, even locally adaptive ones, could unlikely solve such cases successfully, but neural networks with large receptive field size could overcome this issue. Another way to deal with it is to use the observation that bleeding text tends to have a backward slant, which can be captured by almost any kind of CNN. Between these two approaches we chose the second one, because large receptive field usage could easily result in the network overfitting.

Another challenge for local methods is presented in Fig. 5. The seal should be fully classified as background (Fig. 5*a*) despite it has some human-readable text inside. For any simple binarization method it is a serious problem, because they were mainly designed to deal with binarization problem in general, but in this contest, it is obvious that there are text lines as foreground and everything else as background.

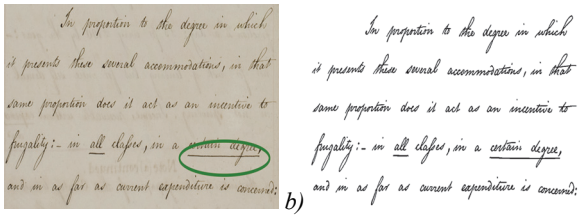


a) *b)*

Fig. 5. Sample with seal and folding: (a) Source, (b) Ground truth

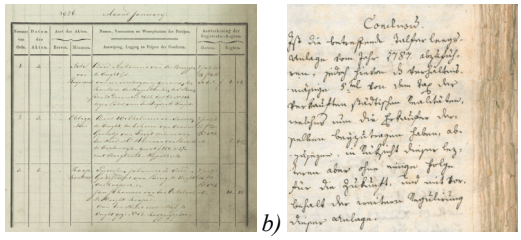
An important exception is shown in Fig. 6*a*. We can notice that rather long thin underlining must be preserved during the binarization process, but it was a rare case in the DIBCO datasets. At the same time, this kind of lines must be separated from paper foldings (Fig. 5*a*), which were, conversely, rather widespread.

To view the problem from a broader perspective, we also looked for the original datasets where these contest images had been extracted from. In parallel, we tried to find other archive collections in public domain. The READ project (URL: https://read.transkribus.eu) was extremely useful during this process. It resulted in several thousands of suitable images (many of them were also used in another ICDAR competitions). We noticed that tables were widespread in the archives, and their layout was virtually indistinguishable from underlinings in the case in Fig. 6*a*. Such a table sample is presented in Fig. 7*a*. From that point of view, layout matters and binarization method should preserve all the table primitives along with the foreground content as in Fig. 6*a*. Another problem is a set of manuscript or book page edges (Fig. 7*b*). These edges tend to have complicated structure but they definitely should be classified as background like the mentioned paper foldings (Fig. 5*a*). Despite the fact that such problems hadn't been presented in the previous contests we were not insured that they would be missing in the upcoming benchmarking dataset. The DIBCO organizers do not provide any samples from the upcoming competition and you must be ready to a wide variation of input data. So, we have selected several images with complex layout and page edges for further usage.



a) *b)*

Fig. 6. Sample with underlining: (a) Source, (b) Ground truth



a) *b)*

Fig. 7. Samples found in open-access datasets: (a) Table layout, (b) Page edges

Network architecture

As was mentioned above, we consider neural network based solution. This network should produce the output of exactly the same size as an input image. We picked well-known U-Net architecture which could overcome the challenges described earlier and had already been successfully applied for various image segmentation problems [8], [33], [34]. The main advantage of U-shaped architecture is its ability to capture the context in general, like local adaptive binarization methods do, on the contracting path and provide pixel-wise accuracy of classification on the symmetric expanding path which is essential for the DIBCO contest.

The network can be trained end-to-end without specifying any information about the image. We used all the 86 images from the previous contests as an initial dataset. Every image was reduced to grayscale before the training process. We divided these images into small patches of 128×128 pixels. The patch size was selected experimentally (we tried all the powers of two from 16×16 to 512×512). The samples of these patches are shown in Fig. 8. As a result, we generated approximately 70000 patches. 56000 of them were used for the network training and other 14000 were used for the validation. We used cross-validation for quality measurement because of the dataset variability. For every validation step we split initial dataset into two groups using 80/20 rule (69 images for training, 17 images for validation), so patches from the same images never got into both train and validation

subsets simultaneously. No augmentation methods were applied at this stage. For every patch the ideal binary mask from the provided ground truth was assigned.

The learning process was implemented using Keras [35] library. We used Adam optimizer [36] and binary cross-entropy as a loss function. The final evaluation of pixel binarization result was measured using the standard Intersection over Union (IoU) metric.

Fig. 8. The patch samples

Even the first experiments showed that this approach resulted in a solution which drastically outperformed the baseline methods, Otsu and Sauvola (89.5 % instead of about 78 % using FM metric).

Further training

In our work, network parameter tuning, learning process customization, and data augmentation model selection were done manually. After every experiment we evaluated the relevance of the trained network on images we have found earlier. First iterations failed as intended. The example with incorrectly classified document edges pixels is shown in Fig. 9*b*. In order to overcome these wrong results, we chose 5 images with edges and tables for adding them into the training dataset. We applied our trained network to receive the initial binarization result and then manually corrected all erroneously labeled pixels. This process is demonstrated in Fig. 9.

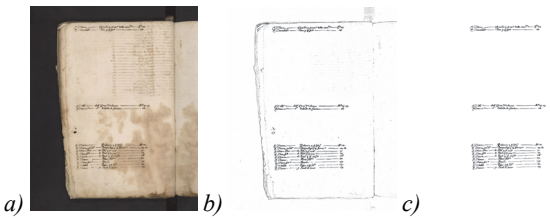


Fig. 9. Preparing sample of ground truth:
(a) Source, (b) Initial result, (c) Ground truth

At that stage we introduced on-the-fly data augmentation strategies to the training process. Data augmentation is essential to provide the network robustness against different kinds of degradations or deformations. Our patch size was small enough to fit in memory and allowed to utilize batch training along with all augmentation strategies. After each iteration we retrieved 2000 worst patches with the highest deviation from the ground truth and images which they had been extracted from. Then we classified the errors by type. For the most common type we prepared an augmentation strategy to generate images with such a problem. Having confirmed that the network really provides bad output on these images we added this strategy to the set of augmentations. Finally, this set consisted of: (i) image shifting, (ii) contrast stretching, (iii) gaussian, salt and pepper noises, (iv) scale variation. The augmented samples are shown in Fig. 10. Due to unavailability of the target dataset we used cross-validation approach again. The 80 / 20 rule was preserved here and patches were grouped by the original big images as on the initial stage. The impact of these augmentation techniques on the cross-validation result is presented in Table 1. We also had to find balanced trade-offs between used augmentation techniques because some results were contradicting. This led to the second column in Table 1, which represent how likely the augmentation would be applied to the patch.

We also tested image mirroring augmentation technique but it resulted in quality degradation, because fragments of slanted text lines bleeding from the opposite page side started to mess up with the regular ones. Gaussian blurring also didn't help us in this problem. The random elastic deformations allowed us to produce better results on handwritten images, but on printed ones results got worse and, after all, we refused to use them. From Table 1 we can observe that using augmentation techniques helped us to increase validation quality from starting 89.53 % to final 99.18 %.

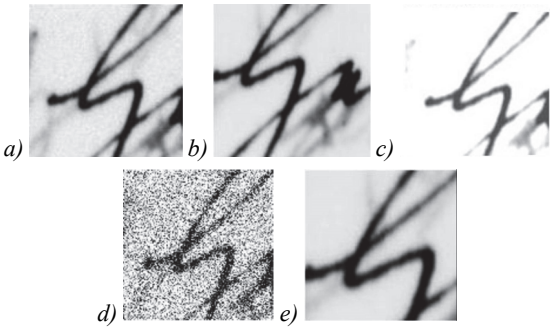


Fig. 10. Augmented patch samples:
(a) Original, (b) Shifted, (c) Contrasted, (d) Noised, (e) Scaled

Table 1. Used types of data augmentation and their impact on cross-validation evaluation

Type	Chance (%)	Quality (%)
No augmentation	-	89.53
+ shift	100	92.58
+ noise	10	93.37
+ contrast	20	95.64
+ scale	20	98.41
+ lines	5	**99.18**

3. Results

During the DIBCO'17 competition our method was independently evaluated by contest organizers and compared to the other 25 binarization techniques. For this purpose, they had prepared two new datasets from 10 machine-printed and 10 handwritten document regions. None of these images were available to the participants before their publication. The final results and all the measurements were presented in [7].

A lot of methods based on convolutional neural networks were submitted and they occupied the top six ranking positions. Such architectures as deep supervised network (DSN), fully convolutional networks (FCN), reccurent neural networks (RNN) with LSTM layers were used in this contest. Some of them used ensembles of several networks which operated over multiple image scales or integrated results from networks with different structures.

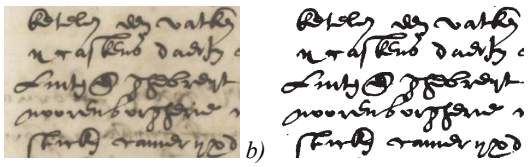
The brief version of that table with evaluation results of submitted methods is presented in Table 2. We can observe that our solution achieved best performance across every

provided metric. It also has score margin from the second place (309 against 455). In this contest, there weren't any images with problems related to the document edges, page foldings, or layout elements, which we tried to overcome. The samples of original images along with binarization results are shown in Fig. 11, Fig. 12.

In Table 3 we show the measurements of previously trained network for the H-DIBCO'18 dataset. We have to notice that it outperformed all participants of the H-DIBCO'18 on the target dataset [37]. Moreover, the organizers also have published results of proposed methods obtained for DIBCO'17 dataset in [37] in Table II. The situation here is the same: no new method was good enough to improve results of the 2017 year.

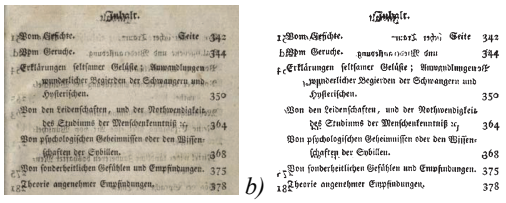
Table 2. Evaluation results (brief version taken from [7])

Brief	Score	FM	Fps	PSNR	DRD
1) U-Net (Proposed method)	**309**	**91.01**	**92.86**	**18.28**	**3.40**
2) FCN (VGGNet)	455	89.67	91.03	17.58	4.35
3) Ensemble (3 DSN)	481	89.42	91.52	17.61	3.56
4) Ensemble (5 FCN, no postprocessing)	529	86.05	90.25	17.53	4.52
5) Ensemble (FCN, with postprocessing)	566	83.76	90.35	17.07	4.33
6) FCN	608	88.37	89.59	17.10	4.94
7) Howe based	635	89.17	89.88	17.85	5.66
Otsu	-	77.73	77.89	13.85	15.5
Sauvola	-	77.11	84.10	14.25	8.85



a) b)

Fig. 11. Binarization result obtained using our solution (handwritten sample): (a) Source, (b) Our result



a) b)

Fig. 12. Binarization result obtained using our solution (printed sample): (a) Source, (b) Our result

Table 3. Benchmark results for H-DIBCO'18

Method	FM	Fps	PSNR	DRD
Our	**89.36**	**92.78**	**19.43**	**3.90**
Winner-2018	88.34	90.24	19.11	4.92

Needless to say, that binarization with such a network is really time-consuming procedure, so the simplification of the final network is highly desirable.

4. Discussion

The proposed solution (trained neural network) evidently was focused on the specific binarization problem of historical document images with Latin-based typeface. An independent evaluation shows that with this predefined set of restrictions the obtained quality is remarkable. But we clearly understand that universal (non-specific) solutions are much more interesting in general. We tried to understand the limitations of our solution. For these purposes we also measured its quality on open parts of Nabuko and LiveMemory datasets taken from the DIB project (URL: https://dib.cin.ufpe.br) using the same DIBCO methodology (these evaluation results are presented in Table 4).

Table 4. Benchmark results for open parts of Nabuko and LiveMemory datasets

Dataset	FM	Fps	PSNR	DRD
Nabuko	90.98	89.71	19.69	2.54
LiveMemory	83.21	78.13	16.87	3.55

The images in Nabuko dataset looks slightly different from DIBCO ones but the obtained results are rather similar. To understand what these numbers mean let's consider the original source image alongside with the result closest to the averaged one for this dataset. In Fig. 13 this source image is presented and a region with a handwriting is highlighted.

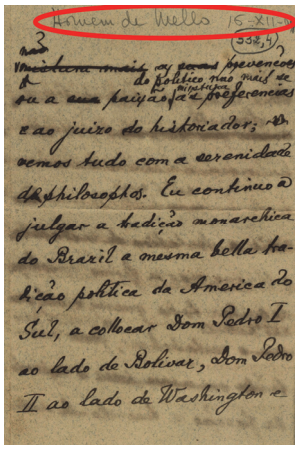
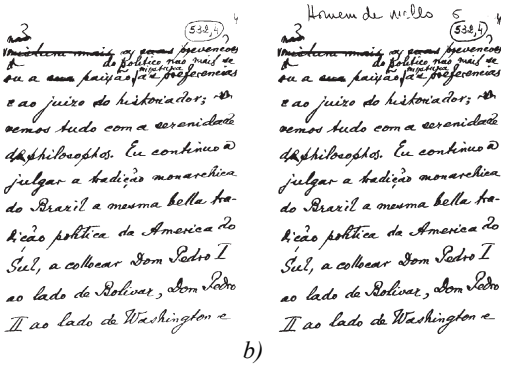


Fig. 13. Sample from Nabuko dataset

In Fig. 14 the corresponding ground truth and our result are shown. The DIBCO measurements for this result are equal to 90.88, 89.41, 17.05, 3.58. The highlighted region contains only background pixels in ground truth and our solution have classified them as a foreground, which seems reasonable from our point of view.



a) b)

Fig. 14. Sample from Nabuko dataset: a) Ground truth, b) Our result

LiveMemory images, in opposite, are far different and the results, in general, are much worse. Sometimes our

binarization gives wrong answer on the regions where simple methods would success easily. For example, it is unable to deal with plots which don't occur in historical documents (Fig. 15).

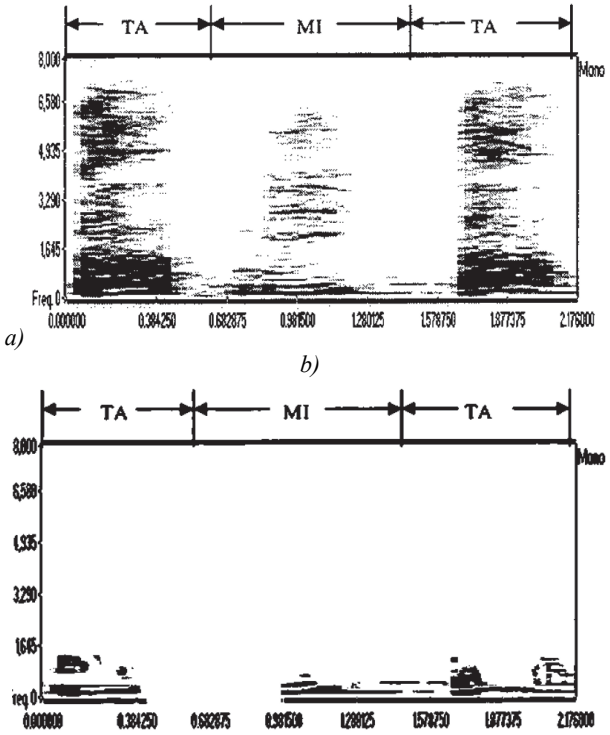


Fig. 15. Sample from LiveMemory dataset: (a) Ground truth, (b) Our result

Analyzing results obtained for this dataset, we assumed that our solution was sensitive to the logical symbol size (these symbols are the main object of interest in the problem domain). The U-Net has a convolutional architecture and this size is very meaningful for it. This assumption was confirmed. The input image fragment for the pretrained network must be resampled in accordance to the expected logical symbol size. On the DIBCO datasets this size was equal to 30 and 60 pixels. For the LiveMemory dataset this size is equal to 15 and simple doubling of the source image leads to far better results. We present such binarized images (with and without upsampling) in Fig. 16. Given this, the method computational complexity is equal to $O(n_s)$, where n_s is an area of the input image after rescaling.

We also checked the solution on set of the images containing hieroglyphs to confirm that the presence of Latin-based typeface was not obligatory for successful binarization. The original image with the obtained result are shown in Fig. 17.

Despite the fact that our solution was not intended to be used outside of the initial domain, the same proposed approach (not the pre-trained network itself) can be applied to specify the binarization problem statement and prepare a solution parameterized with the relevant training data. Also, we need to indicate that for any binarization contest and proper problem statement a presence of consistent ground truth is essential.

Conclusion

Lately, deep convolutional network based solutions outperformed the state-of-the-art methods virtually in every document image analysis problem. In this work we explored the peculiarities of the DIBCO series and focused on neat binarization problem statement. We justified U-Net architecture usage for these purposes and provided some insights for training data preparation. It seems that it was first application of such network submitted to the DIBCO competition. It achieved the best results in this contest in 2017 which stayed unbeatable on H-DIBCO'2018.

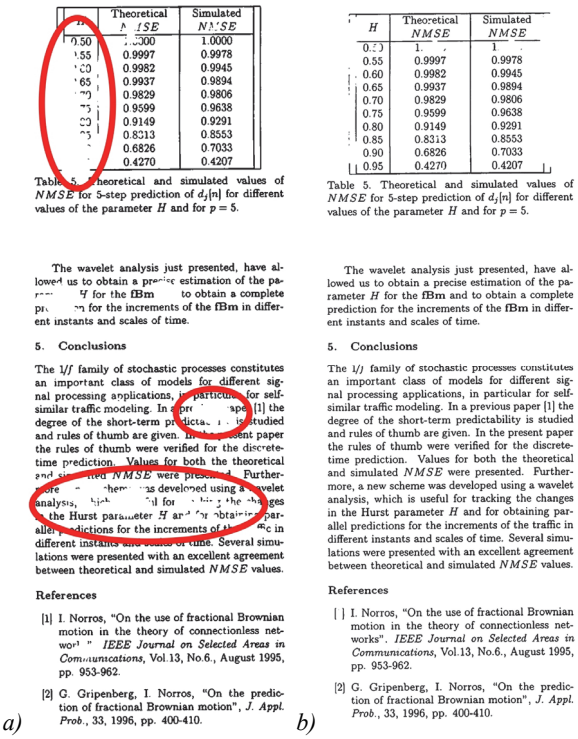


Fig. 16. Binarization result before and after scaling: (a) Without scaling, (b) With scaling

Moreover, such an architecture, as it was mentioned by its authors, can be applied for a huge variety of domains in image segmentation and binarization area. This was recently confirmed in the work [34] where one U-shaped network was used for several historical image analysis tasks simultaneously with an excellent quality. To produce a much more stable solution, a combination of different image datasets must be used during the training process as in recent work [38]. Binarization methods should produce sensible results not only at document scans but also at video streams. Recently, a new mobile captured identity document dataset was published [39] which is suitable for this purpose and brings new set of challenges for the binarization problem.

Our implementation consciously doesn't use any pre- or post-processing steps or any ensembling technique, despite the fact that it could lead to further quality improvement. From our point of view, this solution can be considered as a useful baseline for the further researches related to the enhancements in training data preparation, augmentation techniques usage, and neural

network simplification in the binarization area. We assume that the knowledge how to properly combine the accurate task statement, the domain specific features, and machine learning together is essential and it helped us to outperform other similar network solutions.

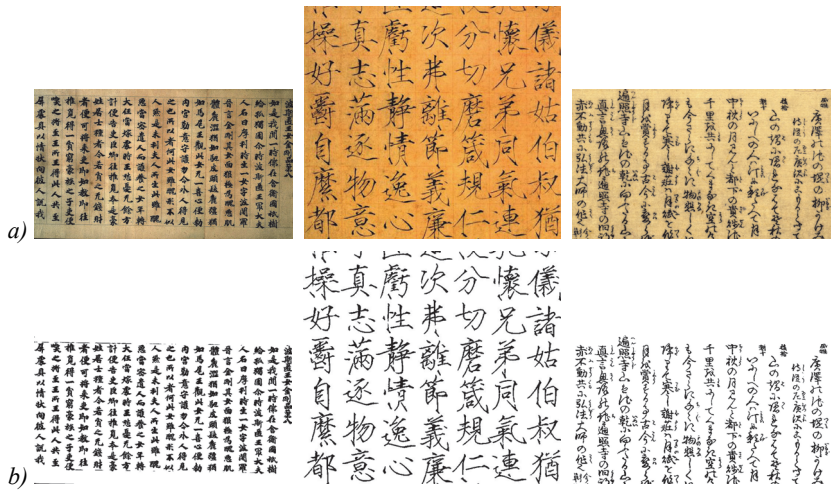


Fig. 17. Samples of documents with hieroglyphs with our binarization results: (a) Source image, b) Result

References

[1] Kruchinin AYu. Industrial DataMatrix barcode recognition for an arbitrary camera angle and rotation [In Russian]. Computer Optics 2014; 38(4): 865-870.

[2] Fedorenko VA, Sidak EV, Giverts PV. Binarization of images of striated toolmarks for estimation of the number of matching striations traces [In Russian]. Journal of Information Technologies and Computational Systems 2016; 3: 82-88.

[3] Gudkov V, Klyuev D. Skeletonization of binary images and finding of singular points for fingerprint recognition. Bulletin of the South Ural State University. Ser Computer Technologies, Automatic Control & Radioelectronics 2015; 15(3): 11-17. DOI: 10.14529/ctcr150302.

[4] Nikolaev DP. Segmentation-based binarization method for color document images. Proceedings of the 6th German-Russian Workshop "Pattern Recognition and Image Understanding" (OGRW-6) 2003: 190-193.

[5] Nagy G. Disruptive developments in document recognition. Patt Recogn Lett 2016; 79: 106-112. DOI: 10.1016/j.patrec.2015.11.024.

[6] Gatos B, Ntirogiannis K, Pratikakis I. ICDAR 2009 document image binarization contest (DIBCO 2009). 2009 10th International Conference on Document Analysis and Recognition 2009: 1375-1382. DOI: 10.1109/icdar.2009.246.

[7] Pratikakis I, Zagoris K, Barlas G, Gatos B. ICDAR2017 Competition on document image binarization (DIBCO 2017). 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) 2017; 1: 1395-1403. DOI: 10.1109/icdar.2017.228.

[8] Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. 2015. Source: ⟨https://arxiv.org/abs/1505.04597⟩.

[9] Otsu N. A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 1979; 9(1): 62-66. DOI: 10.1109/tsmc.1979.4310076.

[10] Sauvola J, Pietikäinen M. Adaptive document image binarization. Pattern Recognition 2000; 33(2): 225-236. DOI: 10.1016/s0031-3203(99)00055-2.

[11] Cheriet M, Said JN, Suen CY. A recursive thresholding technique for image segmentation. IEEE Trans Image Process 1998; 7(6): 918-921. DOI: 10.1109/83.679444.

[12] Jianzhuang L, Wenqing L, Yupeng T. Automatic thresholding of gray-level pictures using two-dimension Otsu method. International Conference on Circuits and Systems 1991. DOI: 10.1109/ciccas.1991.184351.

[13] Ershov EI, Postnikov VV, Terekhin AP, Nikolaev DP. Exact fast algorithm for optimal linear separation of 2D distribution. European Conference on Modelling and Simulation 2015: 469-474.

[14] Shi Z, Setlur S, Govindaraju V. Digital image enhancement using normalization techniques and their application to palm leaf manuscripts. 2005. Source: ⟨https://cedar.buffalo.edu/~zshi/Papers/kbcs04_261.pdf⟩.

[15] Gatos B, Pratikakis I, Perantonis SJ. Adaptive degraded document image binarization. Pattern Recognition 2006; 39(3): 317-327. DOI: 10.1016/j.patcog.2005.09.010.

[1] Lu S, Su B, Tan CL. Document image binarization using background estimation and stroke edges. Int J Doc Anal Recognit 2010; 13(4): 303-314. DOI: 10.1007/s10032-010-0130-8.

[16] Niblack W. An introduction to digital image processing. Upper Saddle River, NJ: Prentice-Hall Inc; 1990.

[17] Trier OD, Taxt T. Evaluation of binarization methods for document images. IEEE Trans Pattern Anal Mach Intell 1995; 17(3): 312-315. DOI: 10.1109/34.368197.

[18] Khurshid K, Siddiqi I, Faure C, Vincent N. Com parison of Niblack inspired binarization methods for ancient documents. Document Recognition and Retrieval XVI 2009. DOI: 10.1117/12.805827.

[19] Lazzara G, Géraud T. Efficient multiscale Sauvola's binarization. Int J Doc Anal Recognit 2014; 17(2): 105-123. DOI: 10.1007/s10032-013-0209-0.

[20] Kim I-J. Multi-window binarization of camera image for document recognition. Ninth International Workshop on Frontiers in Handwriting Recognition 2004: 323-327. DOI: 10.1109/IWFHR.2004.70.

[21] Howe NR. Document binarization with automatic parameter tuning. Int J Doc Anal Recognit 2012; 16(3): 247-258. DOI: 10.1007/s10032-012-0192-x.

[22] Wen J, Li S, Sun J. A new binarization method for non-uniform illuminated document images. Pattern Recognition 2013; 46(6): 1670-1690. DOI: 10.1016/j.patcog.2012.11.027.

[23] Chen Y, Leedham G. Decompose algorithm for thresholding degraded historical document images. IEE

Proc – Vision, Image, Signal Process 2005; 152(6): 702. DOI: 10.1049/ip-vis:20045054.
[24] , Lin W-H, Chang F. A binarization method with learning-built rules for document images produced by cameras. Pattern Recognition 2010; 43(4): 1518-1530. DOI: 10.1016/j.patcog.2009.10.016.
[25] Gatos B, Pratikakis I, Perantonis SJ. Improved document image binarization by using a combination of multiple binarization techniques and adapted edge information. 19th International Conference on Pattern Recognition 2008. DOI: 10.1109/icpr.2008.4761534.
[26] Badekas E, Papamarkos N. Optimal combination of document binarization techniques using a self-organizing map neural network. Eng Appl Artif Intell 2007; 20(1): 11-24. DOI: 10.1016/j.engappai.2006.04.003.
[27] Wu Y, Natarajan P, Rawls S, AbdAlmageed W. Learning document image binarization from data. IEEE International Conference on Image Processing (ICIP) 2016. DOI: 10.1109/icip.2016.7533063.
[28] Westphal F, Lavesson N, Grahn H. Document image binarization using recurrent neural networks. 13th IAPR International Workshop on Document Analysis Systems (DAS) 2018. DOI: 10.1109/das.2018.71.
[29] Tensmeyer C, Martinez T. Document image binarization with fully convolutional neural networks. ICDAR 2017.
[30] Xiong W, Xu J, Xiong Z, Wang J, Liu M. Degraded historical document image binarization using local features and support vector machine (SVM). Optik 2018; 164: 218-223. DOI: 10.1016/j.ijleo.2018.02.072.
[31] Nikolaev DP, Saraev AA. Quality criteria for the problem of automated adjustment of binarization algorithms [In Russian]. Proceeding of the Institute for Systems Analysis of the Russian Academy of Science 2013; 63(3): 85-94.
[32] Krokhina D, Shkanaev AY, Polevoy DV, Panchenko AV, Nailevish SR, Sholomov DL. Analysis of straw row in the image to control the trajectory of the agricultural combine harvester (Erratum). Tenth International Conference on Machine Vision (ICMV 2017) 2018: 90. DOI: 10.1117/12.2310143.
[33] Chollet F, et al. Keras: The Python deep learning library. 2015. Source: ⟨https://keras.io⟩.
[34] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014. Source: ⟨https://arxiv.org/abs/1412.6980⟩.
[35] Pratikakis I, Zagori K, Kaddas P, Gatos B. ICFHR 2018 Competition on Handwritten Document Image Binarization (H-DIBCO 2018). 16th International Conference on Frontiers in Handwriting Recognition (ICFHR) 2018. DOI: 10.1109/icfhr-2018.2018.00091.
[36] Oliveira SA, Seguin B, Kaplan F. dhSegment: A generic deep-learning approach for document segmentation. 2018 16th Int Conf Front Handwrit Recognit 2018: 7-12.
[37] Calvo-Zaragoza J, Gallego A-J. A selectional auto-encoder approach for document image binarization. Pattern Recognition 2019; 86: 37-47. DOI: 10.1016/j.patcog.2018.08.011.
[38] Arlazarov VV, Bulatov K, Chernov TS, Arlazarov VL. MIDV-500: A dataset for identity documents analysis and recognitionon mobile devices in video stream. 2018. Source: ⟨https://arxiv.org/abs/1807.05786⟩.

Author's information

Pavel Vladimirovich Bezmaternykh, a software engineer at Institute for Systems Analysis of RAS, graduated from Moscow Institute of Steel and Alloys in 2009. Research interests are image processing, document image analysis. E-mail: *bezmpavel@smartengines.com* .

Dmitrii Alexeevich Ilin, graduated from Moscow Institute of Physics and Technology (State University) in 2014, majoring in Applied Mathematics and Computer Science. He has 7 years of experience working as a data scientist, developing high quality optical character recognition systems, algorithms and data flow processes for customers' attributes predictions, etc. Currently he works as a machine learning engineer. Research interests are computer vision, neural networks and data analysis. E-mail: *dmitry.ilin@phystech.edu* .

Dmitry Petrovich Nikolaev, Ph.D. in Physics and Mathematics, a head of the laboratory at the Institute for Information Transmission Problems of RAS. Graduated from Moscow State University in 2000. Research interests are machine vision, algorithms for fast image processing, pattern recognition. E-mail: *dimonstr@iitp.ru* .

Received June 20, 2019. The final version – August 01, 2019.