Binarization-free OCR for historical documents using LSTM networks

Conference Paper · August 2015

DOI: 10.1109/ICDAR.2015.7333935

Mohammad Yousefi

76 PUBLICATIONS **395** CITATIONS

SEE PROFILE

Thomas Breuel
Google Inc.

257 PUBLICATIONS **3,786** CITATIONS

SEE PROFILE

Ehsanollah Kabir
Tarbiat Modares University

88 PUBLICATIONS **884** CITATIONS

SEE PROFILE

Didier Stricker
Technische Universität Kaiserslautern

210 PUBLICATIONS **2,358** CITATIONS

SEE PROFILE

Binarization-free OCR for Historical Documents Using LSTM Networks

Mohammad Reza Yousefi*, Mohammad Reza Soheili*‡, Thomas M. Breuel†, Ehsanollah Kabir‡ and Didier Stricker*
*German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany
Email: {yousefi, soheili, didier.stricker}@dfki.de
†Technical University of Kaiserslautern, 67663 Kaiserslautern, Germany; Email: tmb@cs.uni-kl.de
‡Department of Electrical and Computer Engineering, Tarbiat Modares University, Iran
Email: kabir@modares.ac.ir

***Abstract*—A primary preprocessing block of almost any typical OCR system is binarization, through which it is intended to remove unwanted part of the input image, and only keep a binarized and cleaned-up version for further processing. The binarization step does not, however, always perform perfectly, and it can happen that binarization artifacts result in important information loss, by for instance breaking or deforming character shapes. In historical documents, due to a more dominant presence of noise and other sources of degradations, the performance of binarization methods usually deteriorates; as a result the performance of the recognition pipeline is hindered by such preprocessing phases. In this paper, we propose to skip the binarization step by directly training a 1D Long Short Term Memory (LSTM) network on gray-level text lines. We collect a large set of historical Fraktur documents, from publicly available online sources, and form train and test sets for performing experiments on both gray-level and binarized text lines. In order to observe the impact of resolution, the experiments are carried out on two identical sets of low and high resolutions. Overall, using gray-level text lines, the 1D LSTM network can reach 25% and 12.5% lower error rates on the low- and high-resolution sets, respectively, compared to the case of using binarization in the recognition pipeline.**

I. Introduction

Document binarization is playing an important role as the starting point in most of typical OCR pipelines, such that its performance has a key effect on the degree of success of the upcoming processing stages. A main advantage of document binarization is the removal of noise and any other source of redundancy in the input image that might hinder the final system recognition accuracy. An unfavorable outcome is, however, loss of important information during this process that can happen, among other things, in form of broken or deformed characters, which can potentially limit the performance of the whole pipeline.

In general, document binarization techniques can be coarsely divided into global and local approaches. In global methods, a single threshold value is determined based on the statistics of a given image, that is further used to threshold the document image. Global methods demonstrate robust performance when applied to documents with uniform background and clear separation with the foreground. However, such methods perform poorly in presence of document degradations such as non-uniform illumination and bleed-through. The alternative set of approaches makes use of thresholds that are calculated adaptively based on a local window centered around a pixel that is being thresholded. Local approaches are in particular of more interest in dealing with documents of mentioned degradations. A detailed review of binarization methods are beyond the scope of this paper; comprehensive reviews can be found at Refs. [1]–[5]

Even state-of-the-art binarization techniques can still result in some information loss by deforming or breaking characters. The situation worsens with low quality documents, such as historical documents that are inherently available with very degraded paper quality or print qualities. In such cases, binarization can commonly result in errors such as character holes being filled (in case of having larger ink drops), or broken characters (in case of having lesser amount of ink in a joint point of a character shape).

In this paper, we examine the feasibility of a character recognition pipeline that operates directly on gray-level text lines with the goal of avoiding the drawbacks and limitations of a binarization step. To that end, we design a number of experiments on historical Fraktur documents and provide comparisons between recognition results obtained from binarized and gray-level input images. The core element of our recognition pipeline is the Long Short Term Memory (LSTM) network, that has been recently shown to achieve very high-performance OCR both on printed and handwritten text across different languages and scripts [6]–[10].

LSTM networks provide segmentation-free recognition as globally trained recognizers that take raw pixel data as input [6], [9], [11]. Belonging to the family of Recurrent Neural Networks, the LSTM architecture was proposed to overcome many of the limitations and problems of earlier recurrent architectures [12], [13], such as issues regarding the vanishing gradient in backpropagation training. LSTM is a highly non-linear recurrent network with multiplicative gates and additive feedback, that enable the network to exploit the context information in the data. A bidirectional variant of LSTM architecture was proposed by Graves et al. [13] to access context in both forward and backward directions, that are then connected to a single output layer. To avoid the requirement of segmented training data, Graves et al. [14] used a forward backward algorithm, known as Connectionist Temporal Classification (CTC) [15], to align transcripts with the output of the neural network.

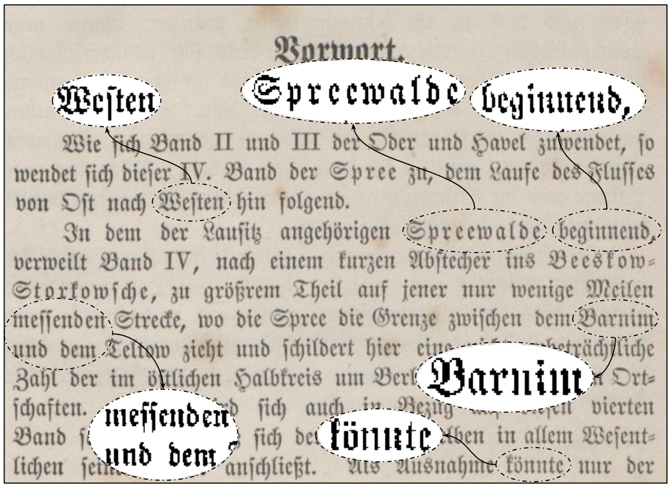
Breuel et al. [6] could achieve state-of-the-art performance by proposing a normalization step followed by a direct application of 1D LSTM networks to printed English and historical German Fraktur OCR. The normalization step is essential in the pipeline as 1D LSTM is not translationally invariant along the vertical axis; a similar approach is employed in this work, too, with modifications to adapt the normalization process for gray-level images.

The goal of this paper is to show that by avoiding the information loss obtained within a binarization-free pipeline on the one hand, and effectively exploiting the learning strengths of LSTM networks on the other hand, we can further reduce the recognition error compared even to the best of the standard binarization-based pipelines, such as [6]. We present experiments on historical German Fraktur documents, considered to be of the more difficult use cases of binarization techniques, due to having a combination of the above-mentioned causes for such techniques' failures.

II. Dataset

Historical documents usually face multiple sources of degradation. Starting from the paper quality, aging of a document paper causes in different types of color fades and changes; non-uniform yellowish background is a typical example of such effects. Another source of difficulty in historical documents is the print technology at the time of publication; inconsistent ink drops, for instance, results in character shapes printed in heavier or lighter strokes. In such cases, it can happen that binarization techniques fill the character holes, or break them into multiple separate shapes. Besides, in manual typesetting, very common in old documents, it could happen for a set of characters to be composed too close to each other in order to fit in a desired layout; in such cases, binarization techniques can easily underperform by either forming a single merged blob of shapes or simply eliminating characters, or major parts of them, by mistaking them for background noise.

In order to have a realistic and reasonable simulation of such cases for our experiments, we collected scanned pages of the four historic volumes of *Wanderungen durch die Mark Brandenburg* by Theodor Fontane, published between 1862 and 1882. The books were published in Fraktur, a common historical German script, containing many touching character and ligatures. The images and the ground-truth are publicly available at Deutsches Textarchiv website [16]. All together, we collected 1762 pages, from which we extracted around 58000 text lines and their corresponding ground-truth text. The training set includes 55000 randomly selected text lines, and the remaining 3000 text lines were used for test phase (from 100 different pages from the four volumes). Page segmentation and text line extraction stages were carried out using OCRopus open source system [17]. In order to observe the impact of input image resolution in our experiments, we collected the document images in two available resolutions of 100 and 200 dpi, respectively.



Vorwort.

Wie ſich Band II und III der Oder und Havel zuwendet, ſo wendet ſich dieſer IV. Band der Spree zu, dem Laufe des Fluſſes von Oſt nach Weſten hin folgend.

In dem der Lauſitz angehörigen Spreewalde beginnend, verweilt Band IV, nach einem kurzen Abſtecher ins Beeskow-Storkowſche, zu größrem Theil auf jener nur wenige Meilen meſſenden Strecke, wo die Spree die Grenze zwiſchen dem Barnim und dem Teltow zieht und ſchildert hier eine [illegible] beträchtliche Zahl der im öſtlichen Halbkreis um Berl[illegible] Ortſchaften. [illegible] ſich auch in Bezug [illegible] vierten Band [illegible] ſich de[illegible]ben in allem Weſentlichen [illegible] anſchließt. Als Ausnahme könnte nur der

Fig. 1. Sample of a scanned historical document from our dataset, presenting a number of most common degradations of a historical document such as spots and variable ink spread within a character shape. The selected areas show equivalent parts of the document taken form the binarized version, in order to illustrate binarization artifacts such as filled/removed holes, broken and deformed characters (selected areas are zoomed-in for better visibility).

Figure 1 illustrates an example of part of a page in our dataset. As shown in this sample image, our dataset contains examples of most of the typical difficulties of historical documents such as spots, non-uniform background, and ink volume variations within the shape of characters and ligatures. It also shows examples of regions taken from the binarized form of the same page, to demonstrate some of the most common binarization artifacts (binarization method in this example is Percentile Filter [18], from OCRopus system; details on the choice of binarization method are provided at Section IV).

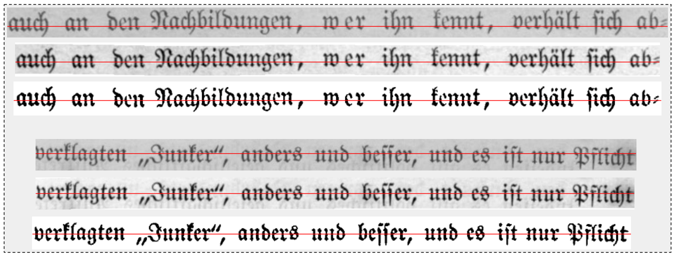
III. Binarization-free Character Recognition

Similar to previous applications of LSTM in printed and handwritten OCR [6], [8], our recognition pipeline is composed of two main components: a text-line normalization step followed by the 1D LSTM-based recognizer.

A. Text-line Normalization

Given that 1D LSTM is not translationally invariant along the vertical axis, the normalization step is necessary to limit the variations to only the horizontal axis. Through the text line normalization step, the absolute position and scale along the vertical axis is normalized to a given height, which fits the 1D LSTM requiring all the input images to have the same height. A number of different methods have been implemented for this normalization step in the OCRopus [17] system; we have chosen the center-normalizer method for this work because it makes few assumptions about the underlying script and has been shown in previous works to perform reliably in both printed and handwritten OCR of different scripts [6], [8].

The center-normalizer inverts the input image and smoothes the resulting image with a large Gaussian filter (sigma equal to half of the image height). The center of the text line is then taken to be the location of the maximum of the smoothed image along the vertical direction; this center is placed in the vertical center of the output image. A scale is computed at each horizontal point by computing the mean absolute deviation of the smoothed image from the centerline; the image is rescaled locally to make this mean absolute deviation constant. In all the previous works, the normalization procedure was only used with binarized text lines. However,



auch an den Nachbildungen, wer ihn kennt, verhält ſich ab-

auch an den Nachbildungen, wer ihn kennt, verhält ſich ab-

auch an den Nachbildungen, wer ihn kennt, verhält ſich ab-

verklagten „Junker“, anders und beſſer, und es iſt nur Pflicht

verklagten „Junker“, anders und beſſer, und es iſt nur Pflicht

verklagten „Junker“, anders und beſſer, und es iſt nur Pflicht

Fig. 2. Two text lines from our dataset. From top to bottom, for each text line, are the original image, and the normalized versions of gray-level and binarized forms. The red line should help to visualize the relative positions of characters in a text line before and after the normalization step. Our normalization can perform equally good on both binarized and gray-level text lines in limiting the translations along the vertical axis. It should be mentioned that the visual difference between the gray-level intensity of original and normalized gray-level images are only due to the intensity value normalization and scaling that is carried out in the normalization process, and no gray-level information is discarded.

the very same procedure but with slight parameter changes can be applied to gray-level text lines. The exact normalization procedure is implemented and available in open source form as part of the OCRopus system [17].

Figure 2 shows the result of normalization step applied to samples of gray-level and binarized text lines from our dataset.

B. LSTM Networks

We use 1D LSTM as the recognition module in our experiments. 1D Bidirectional LSTM networks, in conjunction with the normalization step described above, have been previously shown to achieve very low error rates on printed and handwritten OCR, and we use the a similar architecture as described in [6], [10], that includes a Connectionist Temporal Classification layer (CTC) as the single output layer [15]. The CTC layer is performing ground-truth alignment using a forward-backward alignment, avoiding the requirements for presegmented input data.

For our experiments in this paper, we use OCRopus system [17] and a slightly modified version of open-source RNNLIB library [19], both of which providing implementations of LSTM networks (the modifications include simpler I/O and Unicode support). For experiments with a single hidden layer, we use OCRopus python-based implementation, that on the positive side, converges faster with its modified decoding mechanism at the output layer, but it runs more slowly as the network size increases. For experimenting with larger network architectures, we use RNNLIB as it provides support for multiple layers and runs faster (C++ implementation). Using larger network architectures is making sense given that we try to not only learn the transcription task, but also the network is supposed to generalize over the gray-scale background.

IV. Experiments and Results

To observe the performance of our binarization-free OCR pipeline, we perform a number of experiments on historical Fraktur documents. As described earlier, the degradations inherent in historical documents can highlight the effect of binarization artifacts in the recognition process. To this end, we design experiments to train our pipeline on gray-level and

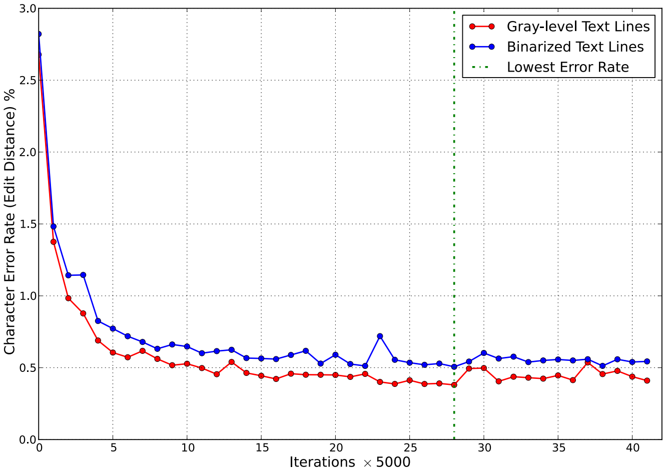


Fig. 3. Test error curve on the low-resolution test set. The results are the average of five times initiating the training procedure with different random seeds. The error rate on the test set is reported every 5000 iterations; training is continued for around four epochs. The lowest average error rate of 0.38% is achieved by using the gray-level text lines, which demonstrates about 24% improvement compared to the lowest error rate obtained by using binarized text lines (with lowest error rate of 0.50%).

binarized images, and compare the results. As error rates, we use edit distance (the ratio of insertions, deletions, and substitutions relative to the length of the ground-truth), and the error rate is measured at a character level. We perform experiments on two identical set of document images in high and low resolution, so that we can evaluate the of outcome of using gray-level images in recognition pipeline in different resolutions (we do not use a lexicon in the experiments reported in this work).

For the binarization, we use Percentile Filter [18] that has been shown to outperform other standard methods, such as Sauvola binarization [20]. As pointed out by Ref. [18], Sauvola binarization might result in a slightly higher Fmeasure compared to Percentile filter, but Percentile Filter performs far better in OCR-based evaluations. This was also confirmed in our preliminary experiments with both binarization schemes and we eventually chose Percentile Filter for the experiments reported in this paper (the details are skipped over for the sake of brevity); implementations of both binarization techniques are available as part of the OCRopus system [17].

For the low resolution set, the text lines are normalized to the height of 25 pixels (chosen to be close to the average height of text lines in the training set). We performed a set of initial experiments with different 1D LSTM architectures, and found 100 hidden states with a learning rate of $1e-4$ to be the optimal parameters for the low resolution set (momentum was set to 0.9 in all experiments).

The error curve on the test set for the low-resolution images, is shown in Figure 3. The results are the average of five times running the experiments with different random seeds for initialization of weights and order of training samples. Using gray-level text lines, our LSTM network was able to achieve an average test error rate of 0.38%, which compared to the lowest average error rate of 0.50% obtained on binarized images, demonstrates 24% improvement (the absolute best error rate on the gray-level images was 0.27% from 547 errors, compared to 0.41% on the binarized images from 813 errors, out of total 196394 characters). Tesseract system [21], running in line-wise mode with a German-Fraktur language model, achieved an error rate of 1.7% on this test set.

Figure 4 shows examples of recognition errors occurred on binarized input, and avoided by using gray-level text lines.

The experiments were repeated for the same dataset but in high-resolution; the goal was to observe the potential improvements by using the gray-level images, also in cases where the binarization technique performs considerably better due to the higher resolution. In this set of experiments, the input text lines were normalized to the height of 48 pixels, and the optimal 1D LSTM network architecture was found, in preliminary experiments, to be two layers of hidden states with 100 and 200 nodes, respectively, and a learning rate of $1e{-}4$. As shown in the test error curve in Figure 5, using gray-level text lines reduces the error for about 12.5% (error rate of 0.14% using the gray-level text lines compared to error rate of 0.16% on binarized text lines). On this test set, Tesseract system [21], achieved an error rate of 1.2% (line-wise mode with German-Fraktur language model).

As expected, the improvement is not so dramatic as the one achieved on the low-resolution set, as the binarization artifacts are reduced to a large extent with higher resolution input images. However, the results are still notable, as we have been able to eliminate one preprocessing step and yet reach the same or slightly improved performance by using the gray-level text lines. Another remarkable finding is the demonstration of learning capacities of LSTM networks, that they are not only capable of achieving a highly accurate transcription performance, but can also internally take care of a large amount of background noise and intensity variations in gray-level text lines, when compared to their binarized forms.

It should be emphasized that we do not claim a full OCR pipeline independent of any binarization step; binarization is still very helpful in performing several preprocessing steps such as layout analysis and page segmentation. Rather, we carefully claim improvements in the recognition accuracy of 1D LSTM network once we use gray-level text lines. And that the improvements are more noticeable in low quality documents, where basically the binarization techniques perform poorly. In such cases, the learning module can do a better job by just operating on the original pixels. In high quality documents, using the gray-level text lines is still making sense, as we can still achieve similar or slightly better recognition accuracy without binarization step in the recognition phase.

V. Conclusion

In this paper, we propose a binarization-free text line recognition system based on 1D LSTM network. The main motivation of skipping the binarization step, and training the network using gray-level text lines, is to avoid the binarization artifacts such as broken or deformed characters, that are among the most common sources of errors in the recognition phase. In the proposed pipeline, the absolute position and scale of the gray-level text lines are first normalized along the vertical axis, such that the variations are limited only to the horizontal axis. A 1D LSTM network is then directly applied to the

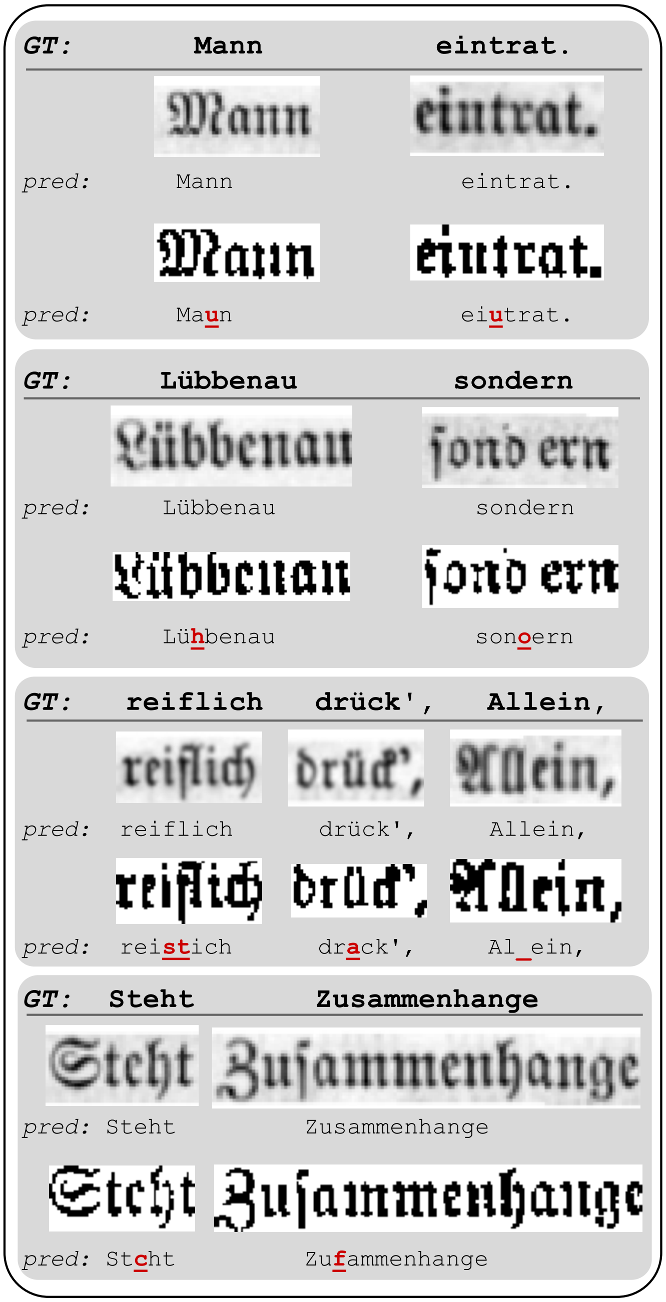


Fig. 4. Examples of the recognition errors, in the low-resolution set, occurring mainly due to binarization artifacts; each box contains sample images in gray-level and binarized form, and the outputs of the recognition pipeline (pred), as well as the ground-truth (GT). The first box, from top, shows examples of characters that are broken in the binarization process and lead to recognition errors, such as a broken "n" confused with "u", happening very frequently in our experiments with binarized text lines. Second box, shows examples of deformed characters after binarization that look very similar to others characters (such as a deformed "ü" followed by "b" looking like an "h" or remaining parts of "d" becoming similar to "o"). The third box includes an example of a Fraktur ligature "fl" being confused with "st" due to binarization errors. The fourth box, shows an example of a broken hole in character "e", that results in a erroneous recognition, due to similarities to the character "c". Similar to the confusion shown in the third box, character "s" is recognized as "f" in the binarized case (we do not have a special symbol for long "s" in the ground-truth). It should be noted that all the errors shown in this example set are avoided by using the gray-level text lines.

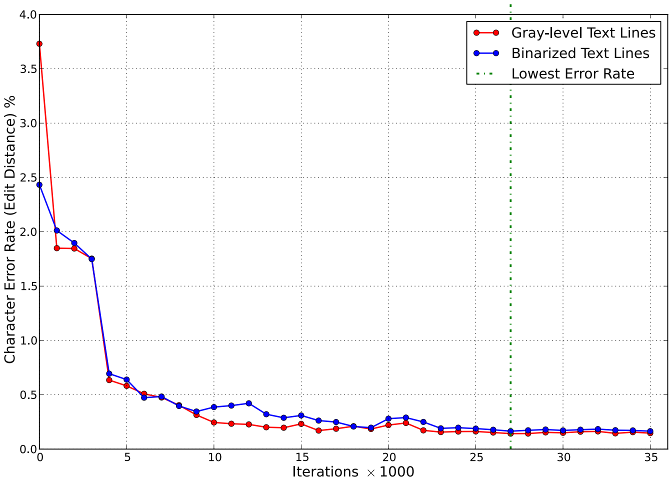


Fig. 5. Test error curve on the high-resolution test set. The lowest average error rate of 0.14% is achieved by using the gray-level text lines, which demonstrates about 12.5% improvement compared to the lowest error rate obtained by using binarized text lines (with lowest error rate of 0.16%). The improvement in error rate obtained by using gray-level text lines is smaller in this case, as the binarization errors and artifacts are reduced when using high-resolution input images.

normalized text lines. Our experiments on two sets of historical documents in low- and high-resolutions demonstrated improved performance over the case of using binarized images. The performance improvements are justified considering that there is essentially less information loss when using the gray-level text lines and character shapes are almost fully preserved. This is, however, possible with the learning capacities of LSTM networks that not only learn the transcription task, but can also successfully generalize over the existing variations in gray-level text lines in terms of nonuniform illumination and background and other sources of degradations, that are particularly present in historical documents.

References

[1] N. Chaki, S. H. Shaikh, and K. Saeed, “A Comprehensive Survey on Image Binarization Techniques,” *Exploring Image Binarization Techniques, Studies in Computational Intelligence*, vol. 560, pp. 5–16, 2014. [Online]. Available: http://link.springer.com/10.1007/978-81-322-1907-1

[2] P. Stathis, E. Kavallieratou, and N. Papamarkos, “An evaluation survey of binarization algorithms on historical documents,” in *2008 19th International Conference on Pattern Recognition*, 2008, pp. 2–5.

[3] M. R. Gupta, N. P. Jacobson, and E. K. Garcia, “OCR binarization and image pre-processing for searching historical documents,” *Pattern Recognition*, vol. 40, no. 2, pp. 389–397, Feb. 2007. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0031320306002202

[4] R. Chamchong and C. C. Fung, “Optimal selection of binarization techniques for the processing of ancient palm leaf manuscripts,” in *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 2010, pp. 3796–3800. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5642008

[5] S. Wu and A. Amin, “Automatic thresholding of gray-level using multistage approach,” in *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*, 2003, pp. 493 – 497 vol.1.

[6] T. M. Breuel, A. Ul-Hasan, M. A. Al-Azawi, and F. Shafait, “High-Performance OCR for Printed English and Fraktur Using LSTM Networks,” in *2013 12th International Conference on Document Analysis and Recognition*. Ieee, Aug. 2013, pp. 683–687. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6628705

[7] A. Ul-Hasan, S. B. Ahmed, F. Rashid, F. Shafait, and T. M. Breuel, “Offline printed urdu nastaleeq script recognition with bidirectional LSTM networks,” in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2013, pp. 1061–1065.

[8] M. R. Yousefi, M. R. Soheili, T. M. Breuel, and D. Stricker, “A comparison of 1D and 2D LSTM architectures for the recognition of handwritten Arabic,” in *2015 Document Recognition and Retrieval XXII, in press.*

[9] A. Graves and J. Schmidhuber, “Offline handwriting recognition with multidimensional recurrent neural networks,” in *Neural Information Processing Systems*, 2009, pp. 1–8. [Online]. Available: http://papers.nips.cc/paper/3449-offline-handwriting-recognition-with-multidimensional-recurrent-neural-networks

[10] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, “A novel connectionist system for unconstrained handwriting recognition.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–68, May 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19299860

[11] G. a. Abandah, F. T. Jamour, and E. a. Qaralleh, “Recognizing handwritten Arabic words using grapheme segmentation and recurrent neural networks,” *International Journal on Document Analysis and Recognition (IJDAR)*, Mar. 2014. [Online]. Available: http://link.springer.com/10.1007/s10032-014-0218-7

[12] S. Hochreiter and J. Schmidhuber, “Long short-term memory.” *Neural Computation*, vol. 9, no. 8, pp. 1735–80, Nov. 1997. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/9377276

[13] A. Graves, “Supervised Sequence Labelling with Recurrent Neural Networks,” Ph.D. dissertation, Berlin, Heidelberg, 2009.

[14] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*, ser. ICML ’06. New York, NY, USA: ACM, 2006, pp. 369–376. [Online]. Available: http://doi.acm.org/10.1145/1143844.1143891

[15] A. Graves, “Connectionist Temporal Classification,” in *Supervised Sequence Labelling with Recurrent Neural Networks, Studies in Computational Intelligence*, 2012, vol. 385, pp. 61–93. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-24797-2_7

[16] “Deutsches Textarchiv - Grundlage für ein Referenzkorpus der neuhochdeutschen Sprache.,” 2015. [Online]. Available: http://www.deutschestextarchiv.de

[17] “OCRopus - Open Source Document Analysis and OCR system.,” 2015. [Online]. Available: github.com/tmbdev/ocropy

[18] M. Afzal, M. Krämer, S. Bukhari, M. Yousefi, F. Shafait, and T. Breuel, “Robust Binarization of Stereo and Monocular Document Images Using Percentile Filter,” in *Camera-Based Document Analysis and Recognition SE - 11*, ser. Lecture Notes in Computer Science, M. Iwamura and F. Shafait, Eds. Springer International Publishing, 2014, pp. 139–149. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-05167-3_11

[19] “RNNLIB: A recurrent neural network library for sequence learning problems..” [Online]. Available: https://github.com/meierue/RNNLIB

[20] J. Sauvola and M. Pietikäinen, “Adaptive document image binarization,” *Pattern Recognition*, vol. 33, pp. 225–236, 2000.

[21] R. Smith, “An overview of the tesseract OCR engine,” *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 2, pp. 629–633, 2007.