ISSN 2664-9969

UDC 004.8
DOI: 10.15587/2706-5448.2024.298405

Oleg Yakovchuk,
Walery Rogoza

AN OVERVIEW OF STATISTICAL AND NEURAL-BASED LINE SEGMENTATION METHODS FOR OFFLINE HANDWRITING RECOGNITION TASK

The object of the research is the line segmentation task. To recognize the handwritten text from the documents in image format offline handwriting recognition technology is used. The text recognizer module accepts input as separate lines, so one of the important preprocessing steps is the detection and splitting of all handwritten text into distinct lines.

In this paper, the handwritten text line segmentation task, its requirements, problems, and challenges are examined. Two main approaches for this task that are used in modern recognition systems are reviewed. These approaches are statistical projection-based methods and neural-based methods. Multiple works and research papers for each type of approach are reviewed analyzing their strengths and weaknesses considering the described tasks, constraints, and input data peculiarities. Overall acquired results are formed in a single table for comparison.

Based on the latest works that utilize deep neural networks the new possibilities of using these methods in recognition systems are described that were unavailable with traditional statistical segmentation approaches.

The constructive conclusions are made based on the review, describing the main pros and cons of these two approaches for the line segmentation task. These results can be further used for the correct selection of suitable methods in handwriting recognition systems to improve their performance and quality, and for further research in this area.

Keywords: *handwriting text line segmentation, line splitting, text detection, recognition algorithms, deep neural networks.*

Received date: 16.12.2023
Accepted date: 08.02.2024
Published date: 12.02.2024

How to cite
Yakovchuk, O., Rogoza, W. (2024). An overview of statistical and neural-based line segmentation methods for offline handwriting recognition task. Technology Audit and Production Reserves, 1 (2 (75)), 14–19. doi: https://doi.org/10.15587/2706-5448.2024.298405

1. Introduction

In the modern era of the digitalized world, much of the information is preferably stored in a digital format. However, there are still a huge amount of documents that are stored as images of different papers, sheets, workbooks, official tasks, scratches, notebooks, blackboards, and even photos of advertisements that contain text. With modern technologies like handwriting recognition, people can digitalize all such documents by getting readable and editable digital text.

«Online» and «Offline» are different subclasses of the Handwriting Recognition task. Online recognition uses the dynamic representation of input text as traces of pen or finger movements. Offline recognition operates with a static representation of the input document like an image that can contain text. Offline handwriting recognition is part of the Optical Character Recognition (OCR) technology. In this paper, let's observe the segmentation methods for offline handwriting recognition task. However, most of the methods can also be applied to online handwriting by employing techniques for converting input data between modalities, for example [1].

The text segmentation stage represents a fundamental preprocessing step in the context of most handwriting recognition systems. The segmentation task can include character segmentation [2], word segmentation [3], and line segmentation serving as the primary step when working with complex documents provided as input. The overall performance of a handwritten recognition system strongly relies on the results of the text line detection process. In the case that text line detection does not give good results, this will affect the accuracy of the word segmentation as well as the text recognition procedure. Text line segmentation of handwritten documents is still one of the most complicated problems in developing a reliable OCR. The nature of handwriting makes the process of text line segmentation inherently challenging.

The goal of the line segmentation task is to partition handwritten text into segments, each containing an isolated and complete line with all its corresponding pixels. These line segments can be presented as bounding boxes, but there are cases where rectangles are not enough to separate two neighbor lines that could be intersected. So, polygons are another way to represent the segmented line regions. An example of the segmentation task workflow is shown in Fig. 1.

The text line segmentation task itself is highly connected with the line detection task. If the first already assumes that the input image is a document that certainly contains some text, the second one covers many more use cases, such as finding the handwritten text of any forms, types, and locations on the image by splitting it into lines. Thereby, the line detection task raises the problem of defining how the text line should be presented.

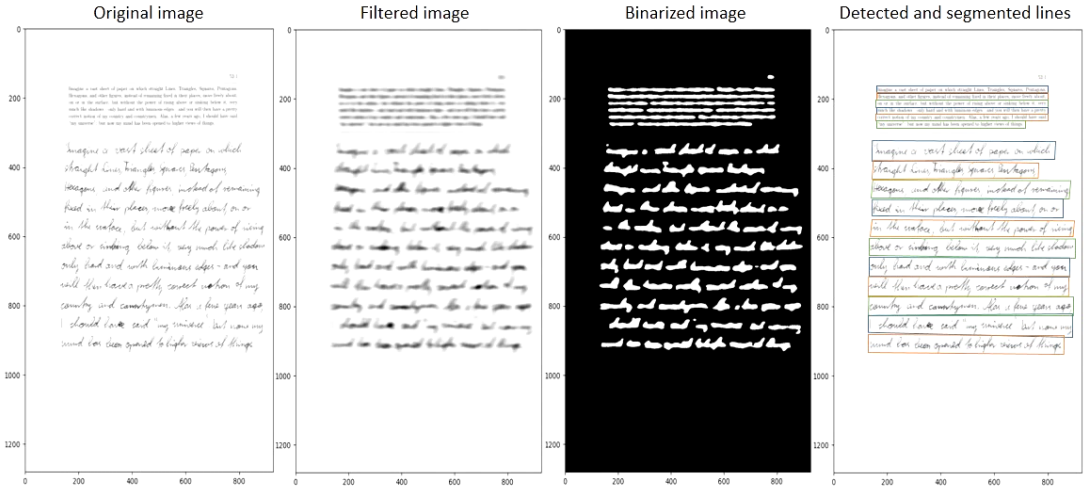

Fig. 1. The basic workflow of the text segmentation task. Filtering and binarization are one of possible preprocessing steps before the segmentation process. The result of handwriting segmentation can be presented as bounding boxes of the segmented text lines

In the literature, one can observe that text lines are defined either as their baseline [4], as their bounding box [5], as the set of pixels corresponding to their handwritten components [6], or as the area corresponding to the core of the text without parts of ascenders and descenders, also called X-Height [6].

There are plenty of line segmentation methods: projection-based, smearing, grouping, methods based on the Hough transform, and stochastic, together with methods based on Neural Networks (NN) that also resolve the text detection task. *The aim of this research* is to provide a review of existing approaches for the offline handwritten text line segmentation proposed by researchers, encompassing both projection-based and neural-based methods.

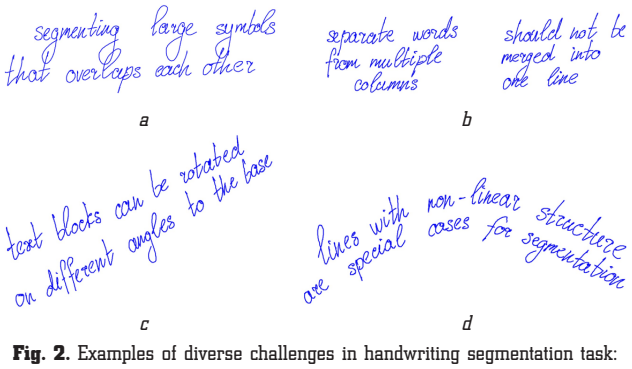

Fig. 2. Examples of diverse challenges in handwriting segmentation task: *a* – strokes overlapping between lines; *b* – columns structure; *c* – rotated lines; *d* – non-linear text structure

2. Materials and Methods

Splitting lines of text presents a challenging task, especially in the case of multi-block documents, unaligned texts, lines with varying rotation angles, etc. Some systems can require that in addition to horizontally aligned text, vertical text should also be correctly segmented and recognized.

Documents containing handwritten text can often contain closely spaced lines when the lower strokes of one line can intersect with the other line below. In such cases, the output line regions may intersect with one another but should not contain any strokes from other lines. Furthermore, variations in writing styles, inconsistent line spacing, and the presence of noise, smudges, and other artifacts in handwritten documents can also hinder the accuracy of the segmentation process. Some of the examples are shown in Fig. 2.

A particularly challenging case is the text written in a single stroke, without any hand-up of the pen. In such cases, the boundaries between characters and words are not clearly defined, making it difficult to accurately segment the text.

The line splitting task is also applicable to other types of documents such as lists, formulas, and tables, each with its own specific requirements. For example, documents with formulas should be split into separate lines with individual formulas in each line, considering that a single formula can consist of multiple levels with frac elements [7].

Addressing the described challenges requires the development of sophisticated algorithms that can accurately identify and segment the text while accounting for variations in handwriting styles and document layouts.

3. Results and Discussion

3.1. Projection-based approach. This approach is based on a common text structure that contains the static vertical distance between text lines and space between words in a row. Such a method to separate lines and words is intuitive and is used by humans when one is reading any structured text.

Horizontal projection method. For the line segmentation task, the horizontal projection methods are used. To obtain the vertical projection profile it's needed to sum pixel values along the horizontal axis for each y coordinate. From such a profile curve, the vertical gaps between the text lines can be determined. The profile curve is analyzed to find its extremums [8].

To divide the lines into individual regions, an intensity threshold is chosen based on maximum projection concentrations: the more pixels are projected to the same axis point, the more probability to have a text line on this horizontal level [9]. This threshold has to be proportional to the average line length in the document. In the next step, a false line exclusion algorithm is applied. The lines with a height below a predetermined threshold are removed. The latter threshold value is proportional to the average height of the text lines in the whole document. Sometimes more than two thresholds are used to filter and postprocess the results of projection methods. All these thresholds can be determined heuristically or calculated based on document sizes. An example of successful usage of this method is shown in the Fig. 3.

ISSN 2664-9969

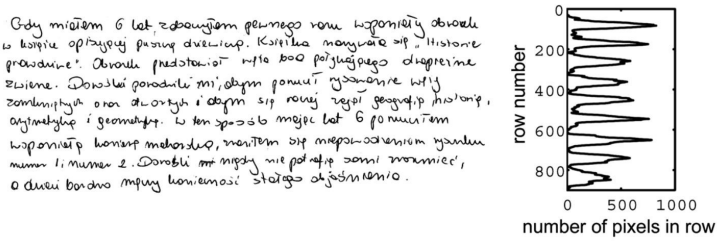

Fig. 3. Horizontal projections method for getting splitted line segments (from [10])

This segmentation method is relatively fast, it also deals with variations in text sizes, structures, and distances between lines. But one of the problems is related to short text lines with a couple of words inside. Such lines give low peaks and then are discarded by the parameter threshold. So, it was proposed an updated projection-based algorithm based on the dynamic threshold that varies depending on the local maximums of the projection graph.

Dynamic threshold for projection method. A global-to-local strategy is used in this projection method improvement. The values of the projection profile are not normalized, so the main idea is to extract the globally significant peaks of the projection graph. The threshold won't be constant in all ranges of arguments. A specific window is used to iterate through the projection graph to filter graph ranges calculating its own threshold value. Notably, that threshold dynamic calculation increases the algorithm's overall complexity [10].

So, this variable threshold permits the determination of low-size lines and some overlapping peaks of the graph. But other problems with this approach remain. Very narrow text lines still may be omitted. Another serious problem is lines overlapping which is a common situation for handwriting texts. Such lines are indistinguishable for simple projection-based methods. A possible solution is to apply the partial projection method combined with other segmentation techniques.

3.2. Neural-based approach. For the text line detection problem, the works that use deep neural networks appeared not so far ago. One of the first proposals was the combination of a Multi-Dimensional Long-short Term Memory (MDLSTM) neural network combined with convolutional layers to predict a bounding box around the line, contributed in [5]. These methods give good results but are limited to horizontal lines, also such networks are pretty heavy to train. Also, by using this approach, the text detection task could be considered, it allows finding and segmenting handwritten or printed text in natural images with a complex background, different surfaces, and non-linear text forms.

Fully Convolutional Network method. The main idea is based on using the X-height representation for a text line, so every pixel of the document image has to be labeled as belonging to the text line or not. Therefore, the text detection problem can be viewed as a semantic segmentation problem where the Fully Convolutional Network is one of the most suitable solutions. An FCN is a convolutional neural network (CNN) whose dense layers have been removed, making it able to process images from variable sizes. Considering dense layers cannot keep the spatial information in the output, a fully convolutional network works as an encoder and decoder, where the encoder corresponds to the CNN without dense layers, and the decoder is an additional part that is used to build an output with the same resolution as the input [11].

The network architecture is shown in the Fig. 4. It uses a 7-layer architecture: the 2 first layers correspond to standard convolutions, with a dilation with value 1. Then two layers with dilation 2 and two layers with dilation 4. Those dilation rates are used to replace pooling layers, in order to keep the same receptive fields as after a 2×2 pooling layer. Finally, a last convolution layer is added for prediction, with dilation 1 and filter size 1. The idea behind those dilations is the fact that text line detection does not require a large context to be efficient.

One more method based on FCN was proposed in [12]. Based on the NN model output the multi-oriented text line candidates are extracted from the output regions by taking the local information (MSER components) into account. False text line candidates are eliminated by the character centroid information. The character centroid information is provided by a separate smaller fully convolutional network (named Character-Centroid FCN).

Authors of the [11] claim the results of FCN are better than methods based on steerable filters. Still, this method has limitations, one of which is much more inference time for method execution compared to statistical approaches. Also, it needs large datasets and much time for model training.

Convolutional model with Differentiable Binarization. The binarization post-processing is an essential step for the segmentation-based detection methods, which converts probability maps produced by segmentation models into lines bounding boxes and regions of text.

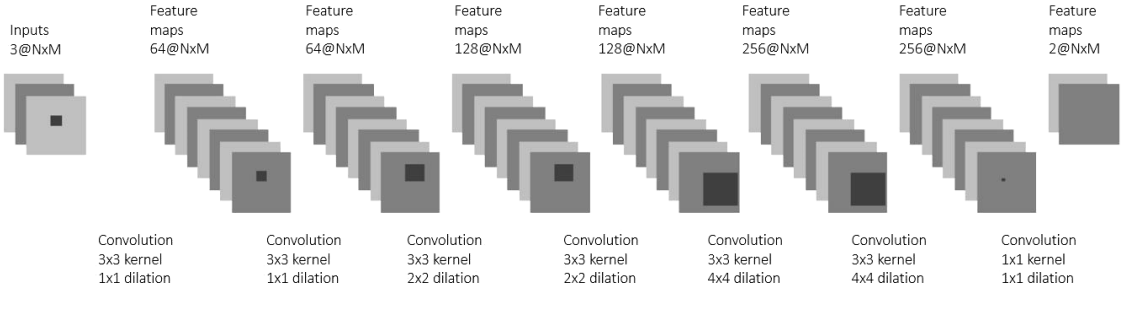

Fig. 4. Fully Convolutional Network model architecture used for text line detection and segmentation (from [11])

The next study [13] suggests a combination of a simple network for semantic segmentation and the new module named Differentiable Binarization which performs the binarization process inside a segmentation network. It allows acquiring the binarization map together with the text segmentation map, therefore it significantly simplifies the post-processing for the next recognition steps and speeds up the whole recognition process.

The model architecture and overall workflow of the proposed method are shown in Fig. 5. Firstly, the input image is fed into a feature-pyramid backbone. The «pred» layer consists of a 3×3 convolutional operator and two deconvolutional operators with stride 2. The «1/2», «1/4», ... and «1/32» indicate the scale ratio compared to the input image. Secondly, the pyramid features are up-sampled to the same scale and cascaded to produce feature *F*. Then, feature *F* is used to predict both the probability map (*P*) and the threshold map (*T*). After that, the approximate binary map ($B\hat{}$) is calculated by *P* and *F*. In the training period, the supervision is applied to the probability map, the threshold map, and the approximate binary map, where the probability map and the approximate binary map share the same supervision. In the inference period, the bounding boxes can be obtained easily from the approximate binary map or the probability map by a box formulation module [13].

A basic post-processing pipeline is used to get the text segmentation results and its binarized segments. First, the probability map produced from the segmentation network is converted to a binary image by applying a step function with a constant threshold. Then, some heuristic techniques like pixel clustering are used to group pixels into text regions. So, a threshold map is predicted adaptively, where the thresholds can be diverse in different regions. Then, an approximate function for binarization (Differentiable Binarization) is used, which binarizes the segmentation map using the threshold map. In this manner, the segmentation network is jointly optimized with the binarization process, leading to better segmentation results.

The testing results show that this method deals well with curved text, intersections between lines, rotated and multi-oriented text (vertical text also), and with multi-language texts [13]. Also, the authors claim that one limitation is that the method can't deal with cases «text inside text», which means that a text instance is inside another text instance, but this is a common limitation for segmentation-based scene text detectors.

3.3. Results evaluation and comparison. Overall experimental results from the reviewed research works are collected in Table 1.

The method described in [9] was tested on 300 images containing 7201 lines. The experiment results in 97.01 % accuracy in correct line segmentation. The line is calculated as incorrectly segmented if even a single component, from one line is associated with another line, if a single line is split into 2 or more lines, or if any number of lines are merged together. Such high accuracy results are explained by the dataset simplicity. It does not contain any hard cases for segmentation, such as line intersections, skewed or rotated lines, etc.

The proposed algorithm from [10] was evaluated on specific unconstrained handwritten Polish documents that contain both ordinary images and more difficult images in terms of text line segmentation. Such cases contain lines of different lengths which is common for texts including many paragraphs or dialogs. On that dataset, this method shows good results with 91.9 % accuracy. The limitation of the proposed method comes from the drawbacks of projection profiling. Given the information from a profile is calculated only in the horizontal direction, therefore the algorithm can't deal well with different slanting and curved text lines.

In [11] proposed a learning-based method for the line segmentation task. Two datasets were used – the cBAD dataset and one more private dataset. The Xheight labeling is provided for each line as ground truth. From the cBad dataset, 176 images were used for training, 40 for validation, and 539 for testing.

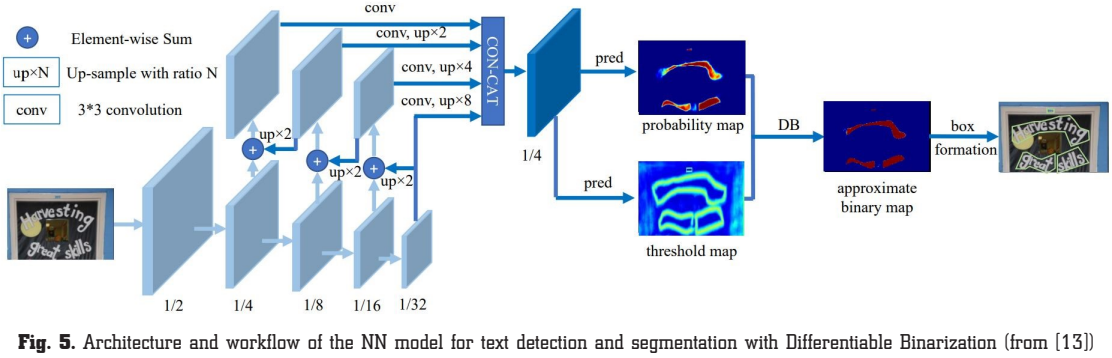

Fig. 5. Architecture and workflow of the NN model for text detection and segmentation with Differentiable Binarization (from [13])

Table 1

Experimental results

Method	Experiment data	Number of lines	Experiment results
Horizontal projection profiling [9]	CEDARFOX dataset (300 images)	7201	97.01 % accuracy
Variable threshold for horizontal projection method [10]	Handwritten Polish documents with mixed length lines (60 images)	1514	91.9 % accuracy
Fully convolutional network [11]	cBAD dataset (539 images)	~16000	93 % mIoU
FCN with additional model for characters centroids [12]	MSRA-TD500 (200 images)	–	74 % *F-score*
Convolutional model with Differentiable Binarization [13]	MSRA-TD500 (200 images)	–	84.9 % *F-score*

ISSN 2664-9969

For the Intersection over Union (mIoU) measure the 93 % result was acquired. This method coped well with low-quality historical documents, also it resolved plenty of problems from the statistical methods, such as segmentation of skewed and rotated lines. Much of the limitations of this method are connected to the large dataset needed for the model training.

In [12] the FCN method is improved by using the additional model for calculating the character's centroids for better detection accuracy. Also, more complex datasets were used for training and testing, such as ICDAR2013, ICDAR2015, MSRA-TD500. On the last dataset, the next results were acquired: *precision*=83 %, *recall*=67 %, *F-score*=74 %, and average time cost for one image 2.1 sec. Compared to older work [14] that is based on a similar approach without the additional mode, the *F-score*=65 % is really lower, but the *time cost*=0.8 sec is almost three times faster. So, with a significant performance decrease, higher detection and segmentation accuracy could be achieved with this improvement.

The research in [13] utilized the same datasets as previous studies, along with additional ones. It considered multiple models with different sizes and parameters, a lightweight backbone ResNet-18, and a full model ResNet-50. An *F-score*=84.9 % was achieved on the MSRA-TD500 dataset with the full model. Together with other results, this method outperforms the state-of-the-art methods on five standard scene text benchmarks, in terms of speed and accuracy. Even with a lightweight backbone, this method can achieve competitive performance with real-time inference speed. In the future, the authors are interested in extending their method for end-to-end text spotting.

The obtained results make it possible to assess the quality of the methods considered for solving the task of handwritten text line segmentation, compare statistical and neural approaches overall, and analyze the limitations of each method. An analysis of the selected methods is provided, describing the main idea, obtained experimental results, and the strengths and weaknesses of each, based on the defined requirements and constraints of the segmentation task. Let's also address the problem of handwritten line detection, for which neural methods are mostly applied.

The research findings can help determine the most suitable algorithm for solving the handwriting segmentation task, considering the specified problem and peculiarities of the handwritten text. The main limitations in this study are raised by the diverse nature of the input data, which can be both handwritten and printed text, varies across different tasks, and is uniquely defined in each reviewed work. Despite conducting the research under martial law in Ukraine, this did not affect the obtained results.

Further research on line segmentation methods could focus on a deeper investigation of the handwritten text detection task itself, which is crucial when working with noisy data, heterogeneous backgrounds, or in conditions without information about the presence of handwritten text in the input data.

4. Conclusions

This paper provides a survey of the line segmentation task, its requirements, challenges, and a comprehensive review of the main approaches that are used in text line segmentation and line detection tasks in modern offline handwriting recognition systems.

The statistical methods that are mostly based on projection profiling are commonly used together with different combinations of morphological filters and postprocessing algorithms. This approach has good segmentation results with different languages on structured documents, but it has limitations for skewed and rotated texts, the same as for unstructured texts with noise backgrounds.

The neural-based methods utilize the convolutional layers in deep networks to detect text lines on any images, so most limitations of this approach are defined by the training datasets. Such methods perform the text line detection task together with the segmentation itself, so they highly extend the possibilities of recognition systems where they are used and resolve many of the problems that were critical for the projection-based segmentation methods. The latest works based on this approach propose a detection system that performs differentiable binarization in a segmentation network that allows getting a good trade-off between detection accuracy and algorithm efficiency. It allows to use these methods in real-time applications achieving high detection results across different surfaces, in the presence of background noise, and accommodating various text orientations and forms.

The results of this study hold potential utility for scientific purposes in future research within this field and can be applied practically by utilizing the considered methods in recognition systems to improve its performance and accuracy.

Conflict of interest

The authors declare that they have no conflict of interest in relation to this study, including financial, personal, authorship, or any other, that could affect the study and its results presented in this article.

Financing

The study was conducted without financial support.

Data availability

The manuscript has no associated data.

Use of artificial intelligence

The authors confirm they did not use artificial intelligence technologies when creating the presented work.

References

1. Sumi, T., Kenji Iwana, B., Hayashi, H., Uchida, S. (2019). Modality Conversion of Handwritten Patterns by Cross Variational Autoencoders. *Computer Vision and Pattern Recognition*. doi: https://doi.org/10.48550/arXiv.1906.06142
2. Volkova, V., Deriuga, I., Osadchyi, V., Radyvonenko, O. (2018). Improvement of Character Segmentation Using Recurrent Neural Networks and Dynamic Programming. *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, 218–222. doi: https://doi.org/10.1109/dsmp.2018.8478457
3. Omayio, E. O., Sreedevi, I., Panda, J. (2022). Word Segmentation by Component Tracing and Association (CTA) Technique. *Journal of Engineering Research*. doi: https://doi.org/10.36909/jer.15207
4. Gruning, T., Labahn, R., Diem, M., Kleber, F., Fiel, S. (2018). READ-BAD: A New Dataset and Evaluation Scheme for Baseline Detection in Archival Documents. 2018 *13th IAPR International Workshop on Document Analysis Systems (DAS)*. Vienna, 351–356. doi: https://doi.org/10.1109/das.2018.38

5. Moysset, B., Kermorvant, C., Wolf, C., Louradour, J. (2015). Paragraph text segmentation into lines with Recurrent Neural Networks. *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, 456–460. doi: https://doi.org/10.1109/icdar.2015.7333803
6. Vo, Q. N., Lee, G. (2016). Dense prediction for text line segmentation in handwritten document images. *2016 IEEE International Conference on Image Processing (ICIP)*, 3264–3268. doi: https://doi.org/10.1109/icip.2016.7532963
7. Yakovchuk, O., Cherneha, A., Zhelezniakov, D., Zaytsev, V. (2020). Methods for Lines and Matrices Segmentation in RNN-based Online Handwriting Mathematical Expression Recognition Systems. *2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP)*. doi: https://doi.org/10.1109/dsmp47368.2020.9204273
8. Razak, Z., Zulkiflee, K., Idris, M., Tamil, E., Noor, M., Salleh, R. et al. (2007). Off-line handwriting text line segmentation: A review. *International Journal of Computer Science and Network Security, 8 (7)*, 12–20.
9. Arivazhagan, M., Srinivasan, H., Srihari, S. (2007). A statistical approach to line segmentation in handwritten documents. *Document Recognition and Retrieval XIV*. doi: https://doi.org/10.1117/12.704538
10. Ptak, R., Żygadło, B., Unold, O. (2017). Projection–Based Text Line Segmentation with a Variable Threshold. *International Journal of Applied Mathematics and Computer Science, 27 (1)*, 195–206. doi: https://doi.org/10.1515/amcs-2017-0014
11. Renton, G., Chatelain, C., Adam, S., Kermorvant, C., Paquet, T. (2017). Handwritten Text Line Segmentation Using Fully Convolutional Network. *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 5–9. doi: https://doi.org/10.1109/icdar.2017.321
12. Zhang, Z., Zhang, C., Shen, W., Yao, C., Liu, W., Bai, X. (2016). Multi-oriented Text Detection with Fully Convolutional Networks. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4159–4167. doi: https://doi.org/10.1109/cvpr.2016.451
13. Liao, M., Wan, Z., Yao, C., Chen, K., Bai, X. (2020). Real-Time Scene Text Detection with Differentiable Binarization. *Proceedings of the AAAI Conference on Artificial Intelligence, 34 (7)*, 11474–11481. doi: https://doi.org/10.1609/aaai.v34i07.6812
14. Xu, Y., Yin, X., Huang, K., Hao, H. W. (2013). Robust Text Detection in Natural Scene Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36 (5)*, 970–983. doi: https://doi.org/10.1109/tpami.2013.182

✉***Oleg Yakovchuk,*** *Assistant, Postgraduate Student, Department of System Design, National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv, Ukraine, ORCID: https://orcid.org/0000-0002-9842-9790, e-mail: olegyakovchuk@gmail.com*

Walery Rogoza, *Doctor of Technical Sciences, Professor, Department of System Design, National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv, Ukraine, ORCID: https://orcid.org/0000-0003-2327-156X*

✉*Corresponding author*