Text Line Segmentation for Challenging Handwritten Document Images using Fully Convolutional Network

***Abstract*—This paper presents a method for text line segmentation of challenging historical manuscript images. These manuscript images contain narrow interline spaces with touching components, interpenetrating vowel signs and inconsistent font types and sizes. In addition, they contain curved, multi-skewed and multi-directed side note lines within a complex page layout. Therefore, bounding polygon labeling would be very difficult and time consuming. Instead we rely on line masks that connect the components on the same text line. Then these line masks are predicted using a Fully Convolutional Network (FCN). In the literature, FCN has been successfully used for text line segmentation of regular handwritten document images. The present paper shows that FCN is useful with challenging manuscript images as well. Using a new evaluation metric that is sensitive to over segmentation as well as under segmentation, testing results on a publicly available challenging handwritten dataset are comparable with the results of a previous work on the same dataset.**

I. Introduction

Historical handwritten documents are valuable since they connect past and present. Commonly they are converted into digital form for being easily available to scholars worldwide. However, digital historical documents pose real challenges for automatic writer identification, keyword searching, and indexing. Text line segmentation of document images is an essential pre-processing operation for these automatizing problems. The problems for text line segmentation of handwritten documents consist of touching, overlapping and crowded characters and vowel signs among consecutive text lines besides narrow interline spacing (Figure 1).

In addition to the problems of handwritten documents, challenging handwritten documents contain various writing styles with inconsistent font types and font sizes through multi-skewed, multi-directed and curved text lines (Figure 2).

Several text line extraction methods for handwritten documents have been proposed. Projection method was initially used for printed documents [1], [2] then modified for skewed [3], [4] and multi-skewed documents [5]. Smearing method [6] which fills the space between consecutive foreground pixels can be used on skewed documents [7] as well. Grouping method aggregates pixels or connected components in a bottom up strategy and is superior in case of skewed and curved text lines [8], [9]. Machine learning algorithms, a type of grouping method, handle text line segmentation as a pixel classification problem. Pixel classification can be done in a sliding window manner [10], [11] which is not desirable due to redundant and expensive computation of overlapping areas in the sliding windows. On the other hand, dense prediction does not suffer from redundant computation and has been successfully used for text line segmentation of handwritten documents [12], [13].

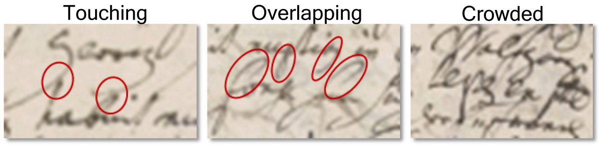

Fig. 1. Text line segmentation problems with regular handwritten documents

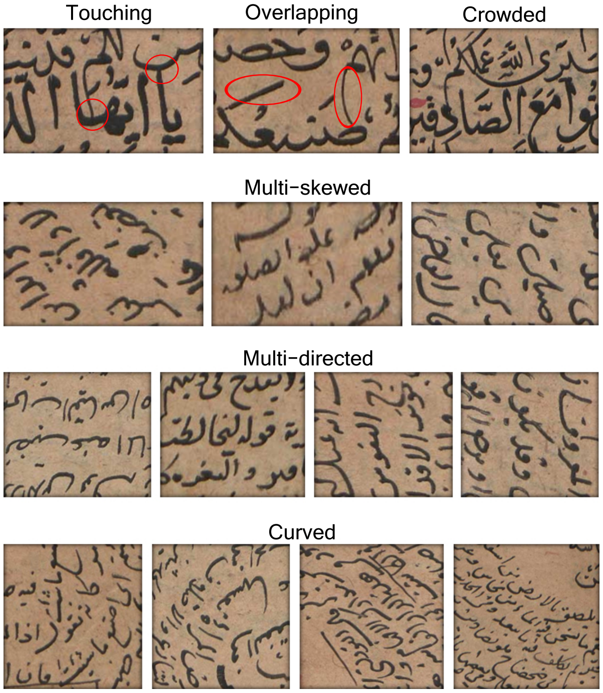

Fig. 2. Additional text line segmentation problems with challenging handwritten documents. Various writing styles are also noticeable.

However, text line extraction of challenging documents has not been extensively studied. This paper provides a dataset[1] of challenging documents with multi-skewed, multi-directed and curved handwritten text lines. It then describes text line segmentation of this dataset using Fully Convolutional Network (FCN). We also propose a new evaluation metric that

[1] https://www.cs.bgu.ac.il/~vml/

is sensitive to both, over and under segmentation of lines in contrast to the available metrics. Using the new evaluation metric we show that FCN based method is comparable to Cohen et al.'s method [9].

In the rest of the paper we describe our method and the new evaluation metric in detail, and present the challenging dataset and report experimental results. After comparing results we conclude and discuss the future directions.

II. Method

Fully Convolutional Network has made great improvements in object segmentation field [14]. It is an end to end semantic segmentation framework that extracts the features and learns the classifier function simultaneously. FCN inputs the original images and their pixel level annotations for learning the hypothesis function that can predict whether a pixel belongs to a text line label or not. So the crucial question is how to annotate the text lines. Baseline labeling is not applicable to all the alphabets whereas bounding polygon is applicable but very cumbersome for crowded documents [15]. Instead of baseline or bounding polygon, we used line mask labeling that connects the characters in the same line (Figure 4).

A. *FCN architecture*

The FCN architecture (Figure 3) we used is based on the FCN proposed for semantic segmentation [14]. First five blocks, encoder part, follow the design of VGG 16-layer network [16] except the discarded final layer. The encoder consists of five convolutional blocks. Each convolutional block contains a number of convolutional layers followed by a max pooling layer. Through the encoder, input image is downsampled, and the filters can see coarser information with larger receptive field. Then the decoder part of FCN upsamples coarse outputs to dense pixels. Upsampling is done by transpose convolution by applying a convolution filter with a stride equal to $\frac{1}{f}$, for upsampling by a factor f.

FCN has two types, FCN8 and FCN32, according to the upsampling factor used in the last layer. FCN32 upsamples the last convolutional layer by $f = 32$ at one time. However, particularly FCN8 architecture was selected because it has been successful in page layout analysis of a similar dataset [17]. FCN8 adds final layer of encoder to the lower layers with finer information, then upsamples the combined layer back to the input size. Default input size of VGG is 224×224, which does not cover more than 2 main text lines and 3 side text lines. To include more context we changed the input size to 320×320 pixels. We also changed the output channel to 2 which is the number of classes, text line or background.

B. *Pre-processing*

We binarize the images by applying an adaptive binarization method for historical documents [18]. Binarized images were inverted before inputting them to the FCN. Then we manually annotated the line masks on the document images. A sequence of original, binarized and labeled document images is demonstrated in Figure 4. Finally a total of 50.000 and 6.000 random patches of size 320×320 were generated for training and validation sets of each fold respectively.

C. *Training and testing*

We applied 6 fold cross validation scheme for the experiments. Each fold was split into train, validation and test sets. In each fold, training continued for 80 epochs and the model with the least validation loss value was saved. The best model was then used to predict the corresponding test set. This training procedure ensures generalizability of the proposed model. The FCN was trained by a batch size of 16, using Stochastic Gradient Descent (SGD) with momentum equals to 0.9 and learning rate equals to 0.001. VGG was initialized with its publicly available pre-trained weights.

During the testing, a sliding window of size 320×320 was used for prediction, but only the inner window of size 100×100 was considered. Page was padded with black pixels at its right and bottom sides if its size is not an integer multiple of the sliding window size, in addition to padding it at 4 sides for considering only the central part of the sliding window.

D. *Post-processing*

Occasionally predicted line masks were disconnected. Thus, we needed to post-process the FCN output. Given a predicted line mask image, firstly the orientation of each connected component was computed. Then the image was split into N layers where each layer contains the connected components with same orientation. Later a directional morphological operation was applied on each layer. Resulting layers at the end were combined back using a pixel-wise OR operation.

Let $C = \{c_1, c_2, ..., c_M\}$ is the set of connected components in the predicted line mask image. C is further divided into N intersecting subsets $B_1, B_2, ..., B_N \subseteq C$ such that:

$$B_i = \{c_i : \alpha(c_i)^2 |v_j^T \cdot \theta(c_i)| < \epsilon\} \quad (1)$$

$$i = 1, 2, \dots M, j = 1, 2, \dots N$$

$$v_j = (\cos(j\frac{\pi}{N}), \sin(j\frac{\pi}{N})) \quad (2)$$

$$\alpha(c) = \frac{R_{maj}}{R_{maj} + R_{min}} \quad (3)$$

where $v_j \in [0, \pi]$ is a particular orientation and $\epsilon \in [0, 1]$ is the threshold for selecting the connected components perpendicular to this particular orientation. R_{maj} and R_{min} are the major and minor axes of the fitted ellipse to the connected component c respectively. $\alpha(c) \in [0.5, 1]$ indicates how sure are we about the orientation of the component c. $\theta(c)$ is the unit vector that represents the orientation of the fitted ellipse to the connected component c. Ellipse fitting was done using the algorithm described in [19].

Eventually for each subset B_i a morphological operation with a narrow kernel in the orientation of this subset was applied. Figure 5 shows the result of post-processing on a sample predicted line mask image.

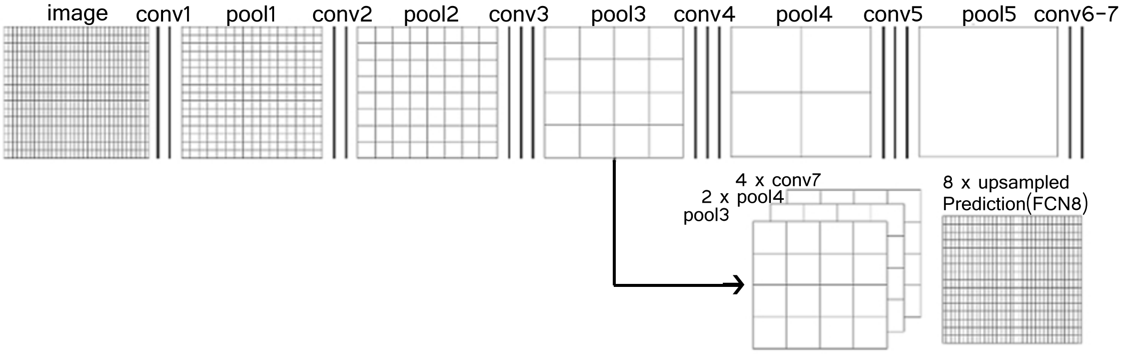

Fig. 3. The FCN architecture. Pooling and prediction layers are shown as grids that show relative coarseness. Convolutional layers are shown as vertical lines. FCN8 4 times upsamples the final layer, 2 times upsamples the pool4 layer and combine them with pool3 layer finally to upsample to input size.

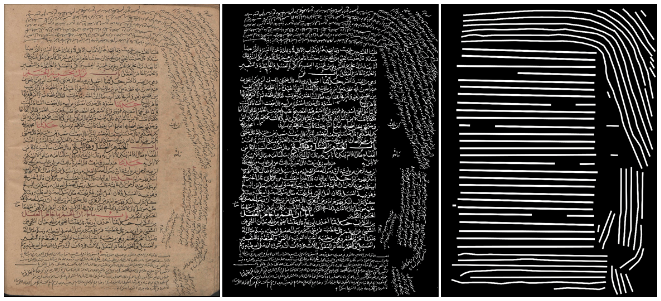

Fig. 4. A sequence of original, binarized and labeled document images. Random patches for training are generated from the binarized and labeled images.

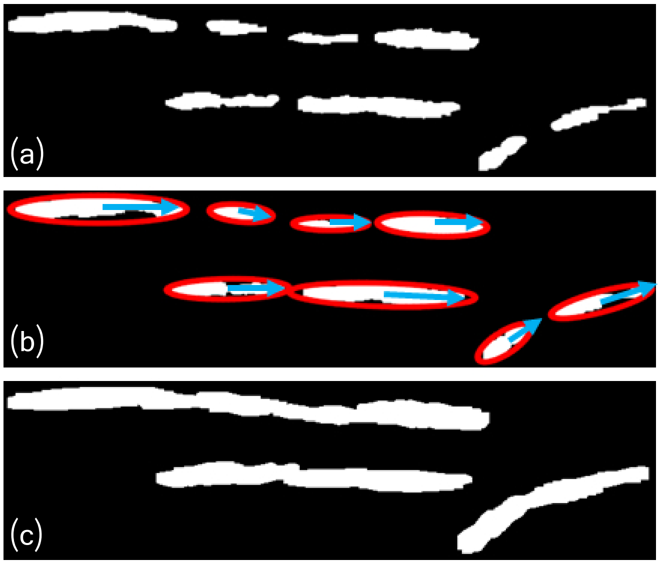

Fig. 5. Post processing phases: (a) Predicted line mask may have disconnected components. (b) For each component an ellipse (red) is fitted and its orientation vector $\theta(c)$ (blue) is computed. (c) Morphological dilation is applied to each component with a narrow kernel in the direction of its fitted ellipse.

E. Connectivity Component Based Line Extraction Accuracy Metric

Available evaluation methods for text line segmentation either use a pixel-wise matching mostly normalized by line length or maximum overlap according to a certain threshold between the extracted and annotated lines. These methods have their short-comings. Thus, we present a different evaluation method that provides a better picture of the results.

The theoretical basis is as follows. A line extraction algorithm succeeds in extracting a complete text line if it has succeeded in finding all the connected components of this line. That is if the algorithm labels all the connected components of a line with the same label, then it has successfully extracted this line without any errors. This is in contrast to having multiple labels, over segmentation, or extracting part of the connected components, under segmentation, along the same text line.

To describe the new metric, we define the term *connectivity component*. A connectivity component is the connection between two consecutive components with the same label. The number of connectivity components in a line is equal to the number of connectivity components between every two consequent connected components and in addition to it a beginning of line connectivity component. The extra connectivity component handles cases where a line contains one connected component only. *C*omplete extraction of a line with several connectivity components is extracting all its connectivity components and assigning them the same label.

To quantify the new metric we define recall and precision for calculating F-measure. Recall is the number of connectivity components extracted by the algorithm in a line, out of all connectivity components found in the corresponding line in ground truth. Precision is the number of correct connectivity components extracted by the algorithm in a line out of all connectivity components extracted by the algorithm. Note that some connectivity components extracted by the algorithm are not found in the ground truth, and some connectivity components are found in the ground truth but not extracted by the algorithm. First type of error is quantified in the precision

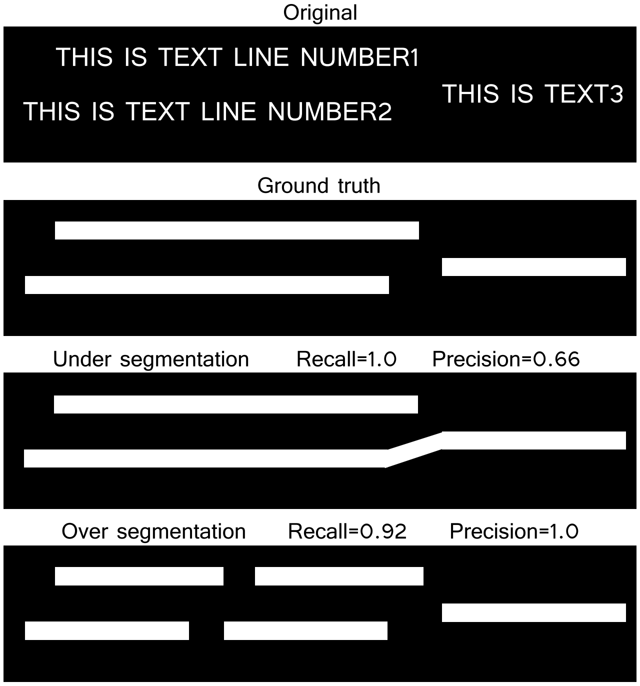

Fig. 6. Connectivity component based metric penalizes under segmentation by its precision definition and over segmentation by its recall definition.

part of the metric, while the latter type of error is quantified in the recall part of the metric.

Let $G = \{g_1, g_2, g_3, \dots g_m\}$ is the set of connected components of a line in the ground truth, $E_i \in \{E_1, E_2, E_3, \dots E_n\}$ is the set of extracted lines such that $E_i \cap G \neq \emptyset$, then for this line in the ground truth, recall (R) and precision (P) is:

$$R = \sum_i \frac{|E_i \cap G| - 1}{|G| - 1} \tag{4}$$

$$P = \frac{\sum\limits_i |E_i \cap G| - 1}{\sum\limits_i |E_i| - 1} \tag{5}$$

The recall definition penalizes over segmentation of a line where an extraction algorithm assigns multiple labels to the components of a single line. In contrast, the precision definition penalizes under segmentation where an extraction algorithm incorrectly assigns a line label to the components that are not in the ground truth of this line (Figure 6).

III. Dataset

Although several benchmark datasets [20]–[22] of handwritten document images are available, a challenging document dataset is absent. We collected a set of challenging document images from the Islamic Heritage Project (IHP), Harvard. This dataset is publicly available[1].

The challenging dataset contains 30 pages from two different manuscripts. It is written in Arabic language and contains 2732 text lines where a considerable amount of them are multi-directed, multi-skewed or curved. Ground truth where text lines were labeled manually by line masks, is also available in the dataset.

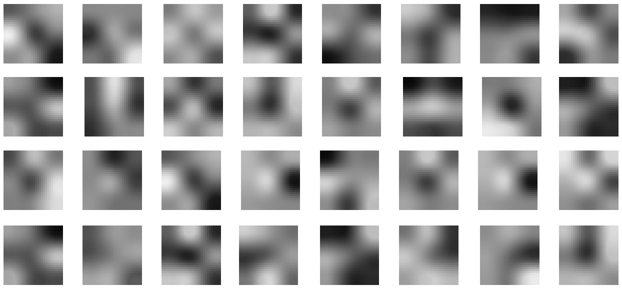

Fig. 7. Visualization of the filters in the first convolutional layer.

TABLE I
Comparison with the method of Cohen et al.

Method	Recall	Precision	F-measure
Proposed	**0.82**	**0.78**	**0.80**
Cohen et al. [9]	0.74	0.60	0.66

IV. Results

We tested the proposed system on the new challenging handwritten document dataset. In each fold we trained FCN on 50.000 patches randomly cropped from 20 pages, validated on 6.000 patches randomly cropped from 5 pages and tested on 5 whole pages using a sliding window. Predicted line mask images were then post-processed with $N = 10$ and $\epsilon = 0.2$. Extracted text lines were evaluated using the new metric to calculate the F-measure.

Entire training took around 9 days. Visualization of the first convolutional layer filters shows that network have learned and filters have converged (Figure 7). The model achieved 89% training accuracy and 88% validation accuracy on average. Two characteristics of the dataset lead the model lacking to overfit to the training set. First it contains two manuscripts with 6 and 24 pages. The manuscript with 6 pages caused most of the errors. Second, although dataset contains considerable amount of multi-skewed, multi-directed and curved lines, they spatially cover smaller area due to smaller font size. This lead to less number of random patches with skewed or curved lines in relative to the number of random patches with regular lines.

Table I shows the performance of our method compared with the method of Cohen et al. [9]. Their approach achieved outstanding results on ICDAR2009 and ICDAR2013 datasets. We run their publicly available code[2] on the challenging handwritten dataset.

Our method outperforms the method of Cohen et al. in terms of both recall and precision. Both methods have lower precision values than recall values. This demonstrates that most of their errors are due to wrongly connected lines in their output. Therefore both method tend to under segment more than over segment. We have noticed that in the output of our method, wrongly connected lines mostly crop up at the

[2] http://www.cs.bgu.ac.il/~rafico/LineExtraction.zip

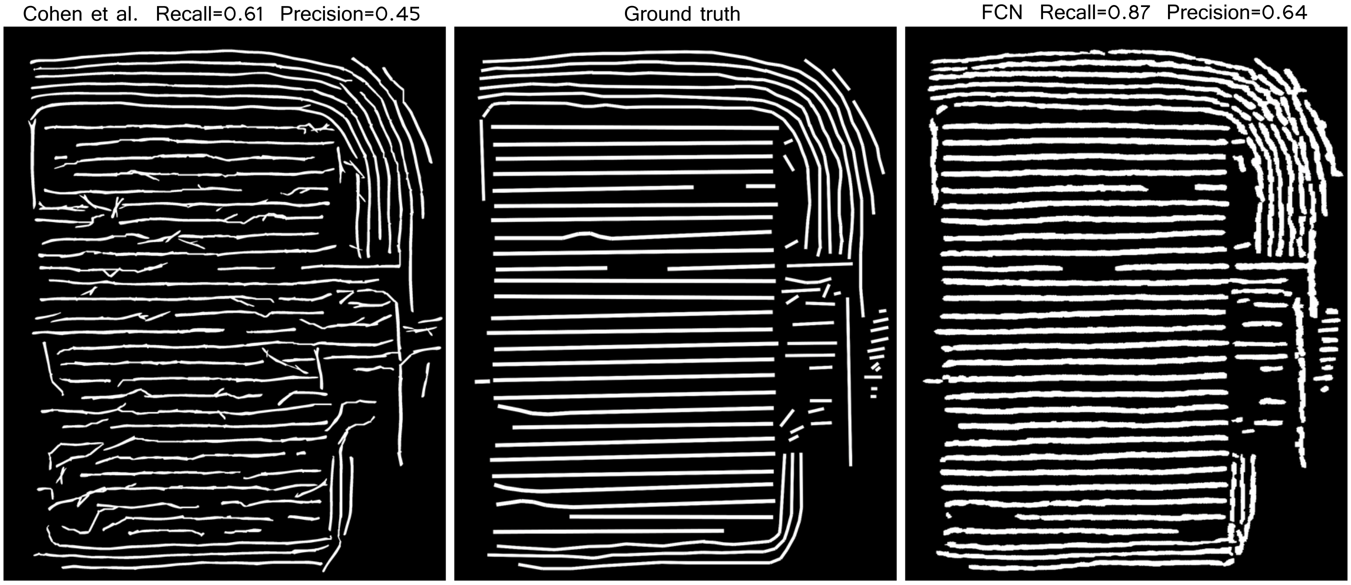
Cohen et al. Recall=0.61 Precision=0.45
Ground truth
FCN Recall=0.87 Precision=0.64

Fig. 8. Sample image of ground truth and corresponding outputs of Cohen et al. [9] and FCN. Lower precision values show that both method tend to under segment. Most errors of FCN method occur at curved areas whereas most errors of method of Cohen et al. occur at the main text areas.

curved areas in contrast to the output of Cohen et al where the wrongly connected lines are mostly crop up at the main text areas. The former was a result of small number of training patches with curved lines. Curved lines can be long but their curved part covers relatively a small spatial area which is one or two corner parts of a page. The latter was a result of small number of main text lines in relative to the number of side text lines in a page, where the average height of text lines converges to the height of side text lines. Therefore method of Cohen et al., which runs according to the average height of text lines, has most errors in main text areas. Figure 8 shows some qualitative results for the latter and the former types of errors on the challenging dataset.

V. Conclusion

This paper introduces challenging handwritten documents, presents a dataset of challenging handwritten documents and its text line segmentation using FCN. Line mask labeling is less cumbersome for challenging handwritten documents and is a proper way for FCN training. We have also defined a new evaluation metric with the concept of connectivity component. This metric is sensitive to both over and under segmentation. New metric is used to validate the proposed method on the challenging handwritten dataset. As a future work, performance on curved text lines can be improved by augmenting patches with curved lines.

References

[1] J. Ha, R. M. Haralick, and I. T. Phillips, "Document page decomposition by the bounding-box project," in *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, vol. 2. IEEE, 1995, pp. 1119–1122.

[2] R. Manmatha and N. Srimal, "Scale space technique for word segmentation in handwritten documents," *Lecture notes in computer science*, pp. 22–33, 1999.

[3] M. Arivazhagan, H. Srinivasan, and S. Srihari, "A statistical approach to handwritten line segmentation," *Document Recognition and Retrieval XIV, Proceedings of SPIE, San Jose, CA*, pp. 6500T–1, 2007.

[4] I. Bar-Yosef, N. Hagbi, K. Kedem, and I. Dinstein, "Line segmentation for degraded handwritten historical documents," in *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*. IEEE, 2009, pp. 1161–1165.

[5] N. Ouwayed and A. Belaïd, "A general approach for multi-oriented text line extraction of handwritten documents," *International Journal on Document Analysis and Recognition*, pp. 1–18, 2012.

[6] K. Y. Wong, R. G. Casey, and F. M. Wahl, "Document analysis system," *IBM journal of research and development*, vol. 26, no. 6, pp. 647–656, 1982.

[7] A. Alaei, U. Pal, and P. Nagabhushan, "A new scheme for unconstrained handwritten text-line segmentation," *Pattern Recognition*, vol. 44, no. 4, pp. 917–928, 2011.

[8] S. S. Bukhari, F. Shafait, and T. M. Breuel, "Script-independent handwritten textlines segmentation using active contours," in *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*. IEEE, 2009, pp. 446–450.

[9] R. Cohen, I. Dinstein, J. El-Sana, and K. Kedem, "Using scale-space anisotropic smoothing for text line extraction in historical documents," in *International Conference Image Analysis and Recognition*. Springer, 2014, pp. 349–358.

[10] B. Moysset, C. Kermorvant, C. Wolf, and J. Louradour, "Paragraph text segmentation into lines with recurrent neural networks," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 456–460.

[11] J. Pastor-Pellicer, M. Z. Afzal, M. Liwicki, and M. J. Castro-Bleda, "Complete system for text line extraction using convolutional neural networks and watershed transform," in *Document Analysis Systems (DAS), 2016 12th IAPR Workshop on*. IEEE, 2016, pp. 30–35.

[12] Q. N. Vo and G. Lee, "Dense prediction for text line segmentation in handwritten document images," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3264–3268.

[13] G. Renton, C. Chatelain, S. Adam, C. Kermorvant, and T. Paquet, "Handwritten text line segmentation using fully convolutional network," in *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on*, vol. 5. IEEE, 2017, pp. 5–9.

[14] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[15] T. Grüning, R. Labahn, M. Diem, F. Kleber, and S. Fiel, “Read-bad: A new dataset and evaluation scheme for baseline detection in archival documents,” *arXiv preprint arXiv:1705.03311*, 2017.

[16] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[17] B. Barakat and J. El-Sana, “Binarization free layout analysis for arabic historical documents using fully convolutional networks,” in *Arabic Script Analysis and Recognition (ASAR), 2nd International Workshop*. IEEE, 2018, pp. 26–30.

[18] I. Bar-Yosef, I. Beckman, K. Kedem, and I. Dinstein, “Binarization, character extraction, and writer identification of historical hebrew calligraphy documents,” *International Journal of Document Analysis and Recognition (IJDAR)*, vol. 9, no. 2-4, pp. 89–99, 2007.

[19] A. W. Fitzgibbon, R. B. Fisher *et al.*, “A buyer’s guide to conic fitting,” *DAI Research paper*, 1996.

[20] B. Gatos, N. Stamatopoulos, and G. Louloudis, “Icdar2009 handwriting segmentation contest,” *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 14, no. 1, pp. 25–33, 2011.

[21] N. Stamatopoulos, B. Gatos, G. Louloudis, U. Pal, and A. Alaei, “Icdar 2013 handwriting segmentation contest,” in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, 2013, pp. 1402–1406.

[22] M. Diem, F. Kleber, S. Fiel, T. Grüning, and B. Gatos, “cbad: Icdar2017 competition on baseline detection,” in *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on*, vol. 1. IEEE, 2017, pp. 1355–1360.