Text line segmentation for handwritten document images

Berat Kurar Barakat[a,**], Rafi Cohen[a], Irina Rabaev[b], Klara Kedem[a], Jihad El-Sana[a]

[a]*Department of Computer Science, Ben-Gurion University of the Negev*
[b]*Software Engineering Department, Shamoon College of Engineering*

ABSTRACT

Handwritten document images contain heterogeneous text lines in terms of text line height, spacing between characters and words. This heterogeneity challenges text line segmentation algorithms. While learning based algorithms can deal with this challenge automatically, non-learning based algorithms require to be adjusted. This paper studies the possibility of an automatic but non-learning based text line segmentation algorithm for handwritten document images. Such algorithm also eliminates the efforts necessary for labeling training data. Proposed algorithm uses automatic scale selection to detect blob lines that hover the text lines. Detected blob lines guides an energy minimization function to extract the text lines. We provide results investigating effects of various factors and compare our method's performance on six publicly available datasets to show its robustness. Proposed approach achieves state of the art results on both, previous and recent publicly available datasets.

1. Introduction

Digital handwritten documents are not easily explorable in their raw form, but need to be transcribed further into machine readable text. Certainly, manual transcription of large number of documents is not feasible in a reasonable time. Hence, there is a significant need for reliable handwritten document image processing algorithms. Text line segmentation is an essential operation and prerequisite for many document image analysis tasks. Advancement in text line segmentation performance will boost the performance of other tasks, such as word segmentation and word recognition.

Text line segmentation consists of text line detection and text line extraction. Text line detection locates each text line by its baseline or x-height representation. Text line extraction in turn leads to polygonal or pixel level representation of text lines. Extraction level representation is more precise and useful for higher level document image analysis tasks. With the advances in deep learning numerous learning based methods have been proposed for text line segmentation of handwritten documents. Learning based methods (Renton et al., 2018; Gruuening et al., 2017; Oliveira et al., 2018; Kurar Barakat et al., 2018) can inherently handle the problems arising from complex layout of text lines and heterogeneity of documents. However, they require vast amount of annotation effort and are tailored for text line detection besides. On the other hand, non-learning based methods (Bukhari et al., 2009; Cohen et al., 2013; Saabni et al., 2014) work like a charm but require to be readjusted for every document image in an ad-hoc manner.

We present a learning free text line segmentation method that does not necessitate annotation effort and automatically adapts to the heterogeneity of handwriting. Our algorithm is based on scale space theory together with automatic scale selection. Automatic scale selection addresses the problem of how to select local appropriate scales for further analysis without need for external parameter tuning. We propose employ this idea for automatic scale selection of heterogeneous text line features. The proposed method first enhances text lines using automatic scale selection together with multi-scale second derivative of anisotropic Gaussian filters. Then, the enhanced lines are binarized to form the blob lines which hover the text lines. Finally, energy minimization via graph cut assigns each connected component to a blob line. A preliminary work about this method was presented at the International Conference on Image Analysis and Recognition (Cohen et al., 2014). This paper investigates it further and promotes its efficiency in comparison to recent learning-based methods.

**Corresponding author:
e-mail: berat@post.bgu.ac.il (Berat Kurar Barakat)

2. Related work

Text line extraction approaches fall in three broad categories: top-down, bottom-up, and hybrid. Top-down approaches partition document images into text lines based on global features. Bottom-up approaches group pixels or connected components based on local features to form text lines. Hybrid approaches combine techniques from top-down and bottom-up categories.

2.1. Top-down approaches

Top-down approaches are mainly based on projection profile, Hough transform, smearing or seam carving. Projection profile sums pixel values among the horizontal axis for each y axis value of a binary image to determine locations of text lines. Projection profile is commonly used for simple document images (Antonacopoulos and Karatzas, 2004) but can be improved for gray scale images (Kesiman et al., 2016) and slightly skewed lines (Ouwayed and Belaïd, 2012).

Hough transform calculates the parameters of linear structures produced by the text components in the document image. Shapiro et al. (1993) used Hough transform to find the global orientation of a document image and apply projection profile along this orientation. Hough transform can also be applied to the centroids of connected components and directly align them as text lines (Gatos et al., 2014). Hough transform is robust with skewed straight lines, but entails high computational cost.

Smearing fills the white space between the consecutive black pixels along the same direction if their distance is within a predefined threshold (Wong et al., 1982). Smearing is sensitive to overlapping text strokes, therefore Shi and Govindaraju (2004) smeared background pixels running through the overlapping strokes to build line separators. On the other hand, Alaei et al. (2011) adapted smearing to skewed lines by applying it in a strip wise fashion. Main difficulty of smearing is determining the optimum threshold and was dealt by Swaileh et al. (2015) using steerable directional filters.

Seam carving computes the path of minimum energy cost from one end of the image to another. Saabni and El-Sana (2011) employed medial seams for text line extraction of binary document images using signed distance transform as the energy map. Later on, they improved the method for gray scale documents using geodesic distance transform as energy map (Saabni et al., 2014).

2.2. Bottom-up approaches

Top-down approaches process the document images at global level, which is problematic when the document does not have a Manhattan layout. Therefore, bottom-up approaches process document images at local level and does not assume straight lines. They group the elements into text lines. Elements can be pixels, super pixels or connected components. The counter part of this is isolation of local elements which is complicated for touching components across consecutive lines. Bottom-up approaches are mainly based on clustering or classification.

Clustering algorithms group elements according to their features in an unsupervised manner (Yin and Liu, 2009). They can be applied to binarized document images (Roy et al., 2012; Diem et al., 2013) as well as to gray scale document images (Garz et al., 2012, 2013). These work clustered super pixels into words and locally join them to form the text lines. Recently Gruuening et al. (2017) clustered super pixels into text lines in a greedy manner therefore applicable to different datasets without parameter tuning. Clustering algorithms are suitable for heterogeneous document collections however the number of clusters has to be selected sensitively.

Classification algorithms classify the elements according to their features in a supervised manner. They are robust to noisy and transformed images, but require a large amount of annotated data for training. Early methods used non-convolutional classifiers with hand crafted features (Baechler et al., 2013; Mehri et al., 2013; Chen et al., 2014). Recent methods are inspired by convolutional neural networks, which have proven to be efficient. Moysset et al. (2015) used recurrent neural network to segment paragraphs but not full pages. Pastor-Pellicer et al. (2016) used convolutional neural network first to classify page pixels as paragraph, and then to classify paragraph pixels as text line or non text line. Text line extraction from full page is studied as a problem of predicting the bounding box around the text lines (Moysset et al., 2016, 2017). Sequential nature of these methods limit them to be used by sliding windows on horizontal text lines. Pixel classification in a sliding window fashion is not desirable desirable due to redundant and expensive computation of overlapping areas in the sliding windows. As a remedy, dense prediction has been successfully used for text line segmentation of handwritten documents (Renton et al., 2018; Kurar Barakat et al., 2018).

2.3. Hybrid approaches

Hybrid approaches combine the strengths of top-down and bottom-up approaches while eliminating their weaknesses. Fischer et al. (2010) calculated the starting point and the skew of text lines with projection profile then use them to search a piecewise linear seperating path. Kumar et al. (2011) globally estimated coarse text lines and locally corrects misclassified elements to solve the problem of touching text lines. Clausner et al. (2012) focused on solving skewed and touching text lines using a combination of clustering connected components and projection profile analysis. Smearing approaches (Bukhari et al., 2009) lack a mechanism for determining the appropriate scale of the filter particularly for degraded gray-scale historical documents.

3. Background

This section presents notations and definitions of (1) the scale space representation with automatic scale selection, (2) component tree and (3) energy minimization via graph cuts.

3.1. Scale space representation with automatic scale selection

Processing unknown image structures can be approached by representing image structures at different scales, so called scale space representation (Witkin, 1987). However, scale space representation does not address the problem of how to select local appropriate scales. Hence, we use automatic scale selection

(Lindeberg, 1998) that adapts to the local scales of image structures. Scale space representation together with automatic scale selection apply to a large class of differential image descriptors. We adapt both of them, for detecting blobs from text lines and term it as blob line detection with automatic scale selection. We formulate a scale space representation using Laplacian of anisotropic Gaussians, since a text line has different scales along the two coordinate directions.

3.1.1. Scale space of anisotropic Gaussians

Given a continuous image $I : \mathbb{R}^2 \to \mathbb{R}$, its scale space of anisotropic Gaussians $L : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ is defined by convolving the image

$$L(x, y; \sigma_x, \sigma_y) = I(x, y) * g(x, y; \sigma_x, \sigma_y) \tag{1}$$

where $g : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ denotes the anisotropic Gaussian kernel

$$g(x, y; \sigma_x, \sigma_y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\left(\frac{x^2}{2\sigma_x{}^2} + \frac{y^2}{2\sigma_y{}^2}\right)} \tag{2}$$

In this representation $*$ is the convolution operator, σ_x is the scale parameter in horizontal direction, and σ_y is the scale parameter in vertical direction.

3.1.2. Scale space of Laplacian of anisotropic Gaussians

Scale space of Laplacian of anisotropic Gaussians is computed by differentiating the scale space of anisotropic Gaussians with respect to x and y two times

$$L_{x^2y^2}(x, y; \sigma_x, \sigma_y) = \partial_{x^2y^2} L(x, y; \sigma_x, \sigma_y) \tag{3}$$

or equivalently by convolving the image with Laplacian of anisotropic Gaussians

$$L_{x^2y^2}(x, y; \sigma_x, \sigma_y) = I(x, y) * [g_{x^2y^2}(x, y; \sigma_x, \sigma_y)] \tag{4}$$

Given the anisotropic Gaussian in Eq. 2, its Laplacian is

$$g_{x^2y^2} = g_{x^2} + g_{y^2} \tag{5}$$

where

$$g_{x^2} = \left(\frac{x^2 - \sigma_x{}^2}{\sigma_x{}^4}\right) \cdot g(x, y; \sigma_x, \sigma_y) \tag{6}$$

and

$$g_{y^2} = \left(\frac{y^2 - \sigma_y{}^2}{\sigma_y{}^4}\right) \cdot g(x, y; \sigma_x, \sigma_y) \tag{7}$$

3.1.3. Automatic scale selection

In scale space representation the amplitude of the Laplacian in Eq. 3 decreases with scale. Based on this phenomena, automatic scale selection states that local extreme over scales of γ-normalized Laplacian

$$L_{x^2y^2}^{\gamma-norm} = (\sigma_x\sigma_y)^{\gamma/2} L_{x^2y^2}(x, y; \sigma_x, \sigma_y) \tag{8}$$

corresponds to the significant structures.

3.2. Component tree

Component tree (Naegel and Wendling, 2010) organizes the connected components of level sets in a tree structure. Let C_t be the set of connected components obtained by thresholding with threshold t. The nodes in a component tree correspond to the components in C_t for varying values of the threshold t. The root of the tree is the member of $C_{t_{\min}}$, where $t_{\min}$ is chosen such that $|C_{t_{\min}}|$=1. Level ℓ in the tree correspond to $C_{t_{\min}+\ell d}$, where d is a parameter that determines the step size for the tree. There is an edge between $C_i \in C_t$ and $C_j \in C_{t+1}$ if and only if $C_j \subseteq C_i$. The maximal threshold $t_{\max}$ used in the tree construction is simply the maximal value in the map.

4. Proposed approach

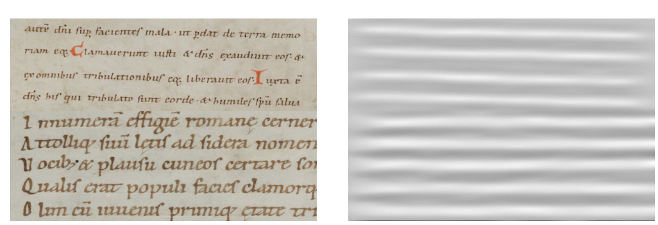
The proposed approach utilizes scale space representation with automatic scale selection to detect blob lines that hover through the text lines. Detected blob lines are then binarized. Finally, energy minimization via graph cuts extracts the text lines with the help of binarized blob lines.

4.1. Blob line detection with automatic scale selection

We informally define a blob to be a connected region that is significantly brighter than its neighbourhood. Text line detection aims to derive the blob lines that hover through the text lines. These blob lines can be derived by convolution of text lines with Laplacian of anisotropic Gaussian from Eq. 5 elongated along the horizontal direction. A good scale for text lines is related to the text line height which is variable in handwritten documents. For this reason we use automatic scale selection that is based on normalized Laplacian in Eq. 8.

We constructed a scale space representation by convolving document image with the Laplacian of anisotropic Gaussian from Eq. 5 within a range of scales corresponding to the height range of the characters in the binarized document. Range computation is also automatic and necessary for narrowing the search space. Hence, we set the range of σ_x to $[\mu/2, (\mu + \sigma)/2]$, where μ and σ are the average and standard deviation of the heights of the connected components in the document; and for every value of σ_x, $\sigma_y = 3 \times \sigma_x$.

Then, for each pixel we chose the strongest response along the γ-normalized Laplacians given by Eq. 8 with $\gamma = 2$. Fig. 1 illustrates the result of blob line detection on a sample heterogeneous document.



(a) (b)

Fig. 1. (a) A heterogeneous document image formed by appending two document images with various text line heights. (b) Detected blob lines using automatic scale selection. Notice that various text line heights are successfully handled automatically.

4.2. *Blob line binarization*

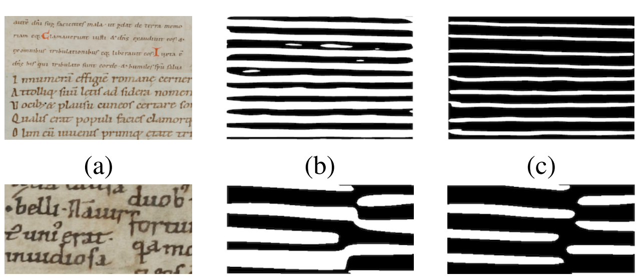
The blob lines detected by automatic scale selection are represented by a gray scale image (Fig. 1(b)). We investigate two binarization methods.

4.2.1. *Component tree*

We binarize this gray scale image using component tree algorithm (Naegel and Wendling, 2010) in order to remove potential false ligatures among the binarized blob lines (Fig. 2(e)). Component tree is built by assigning every node to a connected component of the gray scale image. Each connected component is labeled as valid or invalid based on the max of its ligature scores. To measure the ligature scores, we fit least square linear splines to the points of each connected component. For each spline on a connected component, ligature score is the average 1-norm between the linear fit and the connected component points in that spline. A connected component is considered as valid, if its maximum ligature score is less than the 80% percent of maximum filter scale. We traverse component tree in a breadth first search manner with $d = 1$. At each node, if the connected component is valid, it is taken as a binary blob line, and the search along this branch is complete. Otherwise, component is refined by recursively processing the children of the node.

4.2.2. *Hysteresis*

The gray scale image that represents the blob lines has a range from large negative values to large positive values. Hence we first standardize every pixel in a window size of $N \times N$ where we set $N = 2 \times (\max \sigma_x + 1)$. Standardized blob lines image is binarized using Hysteresis method that allows to ignore the noise outside of blob lines (Fig. 2(c) and (f)).

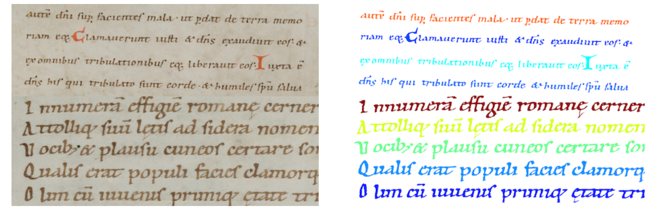


(a) (b) (c)
(d) (e) (f)

Fig. 2. Binarization results of sample input documents in the first column. Second column shows the binary blob lines from using component tree. Component tree can not eliminate all the spurious blob lines. Third column shows the binary blob lines from using hysteresis. Notice that the spurious blob lines among and between the text lines are removed.

4.3. *Text line extraction with energy minimization using graph cuts*

This paper targets text line segmentation. Text line segmentation consists of text line detection and text line extraction. Binary blob lines that are detected in Section 4.2 correspond to the text line detection phase. Extraction level representation requires further to assign a text line label to each pixel in the document image (Fig. 3(b)).



(a) (b)

Fig. 3. (a) A heterogeneous document image formed by appending two document images with various text line heights. (b) Extracted text lines represented by pixel labels.

We use energy minimization (Delong et al., 2012) for assigning connected components to text lines with the help of detected blob lines. It urges to assign components to the label of the closest blob line while straining to assign closer components to the same label and not to assign any component to spurious blob lines. Let $\mathcal{L}$ be the set of binary blob lines, and C be the set of connected components in the binary document image. Energy minimization finds a labeling f that assigns each component $c \in C$ to a label $l_c \in \mathcal{L}$, where energy function $E(f)$ has the minimum.

$$E(f) = \sum_{c \in C} D(c, \ell_c) + \sum_{\{c,c'\} \in \mathcal{N}} d(c, c') \cdot \delta(\ell_c \neq \ell_{c'}) + \sum_{\ell \in \mathcal{L}} h_\ell \quad (9)$$

Energy function has three terms:

1. Data cost is the cost of assigning component c to label l_c. For every $c \in C$, $D(c, \ell_c)$ is defined as the Euclidean distance between the centroid of c and the blob line l_c. Closer the component to a blob line, higher the chance the component to be assigned to the label of this blob line.
2. Smoothness cost is the cost of assigning adjacent components to different labels. Let N be the nearest component pairs. For every $\{c, c'\} \in \mathcal{N}$, $d(c, c') = \exp(-\alpha \cdot d_e(c, c'))$ where $d_e(c, c')$ is the Euclidean distance between the centroids of the components c and c'. $\alpha = (2 \langle d_e(c, c') \rangle)^{-1}$ where $\langle \cdot \rangle$ denotes expectation over all pairs of adjacent elements (Rother et al., 2004). $\delta(\ell_c \neq \ell_{c'})$ is 1 if the condition inside the parentheses holds and 0 otherwise.
3. Label cost: For every blob line $\ell \in \mathcal{L}$, h_ℓ is defined as $\exp(\beta \cdot r_\ell)$ where r_ℓ is the normalized number of foreground pixels overlapping with blob line ℓ and β is a constant we set to 2. The label cost reduces the chance to assign a component to the label of a spurious blob line (Fig. 2(b) and (e)).

5. Evaluation

The proposed method is evaluated on an extensive variety of datasets that cover both historical as well as modern texts of different languages and resolutions. In this section we present these datasets and metrics used to measure the performance on each dataset. For each dataset we use only its test set because our method is not learning based. The proposed method's output is defined as pixel labels but it is manipulated according to the ground truth definition of each dataset.

5.1. Parzival and Saint Gall datasets

Parzival dataset (Baechler et al., 2013) includes 47 pages of a German manuscript from the 13th century. Saint Gall dataset (Diem et al., 2013) contains 60 pages of a Latin manuscript from the 9th century. The ground truth of this dataset is defined as pixel labels hence no manipulation done on the proposed method's output. We measure the performance by means of the Pixel Hit Rate (PHR) and the F-Measure (FM) as described in (Baechler et al., 2013).

5.2. ICDAR 2009 and ICDAR 2013 datasets

ICDAR 2009 dataset (Gatos et al., 2011) contains 200 pages written in English, French, German and Greek. ICDAR 2013 dataset (Stamatopoulos et al., 2013) contains 150 pages written in English, Greek and Bangla. Ground truth is defined as pixel labels, hence no manipulation done on the proposed method's output. We measure the performance by means of the Detection Rate (DR), Recognition Accuracy (RA) and the F-Measure (FM) as described in (Stamatopoulos et al., 2013).

5.3. DIVA-HisDB dataset

DIVA-HisDB dataset (Simistira et al., 2017) contains 30 pages from 3 medieval manuscripts. Ground truth is defined as bounding polygons. Therefore, we get the tight polygons around the pixel labels of our output and measure the performance by means of the Intersection over Union (IU) as described in (Simistira et al., 2017).

5.4. cBAD dataset

cBAD dataset (Diem et al., 2017) contains 539 document images from 7 different archives. This dataset's ground truth is defined as baselines whereas our method extracts pixel labels. To find baselines from pixel labels, first we get tight polygons around the pixel labels of text lines. Then, for each bounding polygon we get its lower contour points and iteratively fit a regression line among these points by excluding the outliers (Fig. 4). Using the extracted baselines, performance is measured by means of Precision (P), Recall (R) and F-measure (FM) as described in (Diem et al., 2017).

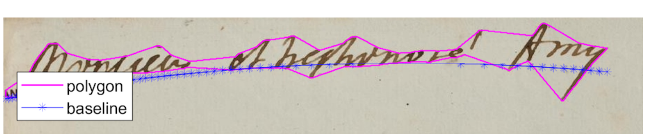


Fig. 4. Baseline extraction from bounding polygon.

6. Experiments

This section investigates the behaviour of the proposed method using various factors. We present the effects of number of knots, type of binarization, component evolution maps for text line height range estimation, and post processing by merging blob lines. These experiments were done on the book CB55 of DIVA-HisDB dataset. Default setting is with 10 knots, component tree binarization, statistical text line height range estimation and without merging broken blob lines. Using the default setting, line IU is 100 and pixel IU is 97.02 (Table 1). In each experiment, we keep all factors same as default but change the factor we want to show its effect and report the result (Table 2).

6.1. Effect of number of knots

When we use component tree to binarize the blob lines, the best threshold for each blob line is determined by the help of a false ligature score. This score is measured by fitting linear splines. The aim of using splines is their ability to fit on curved blob lines as well as straight blob lines. By default, method uses 20 knots to fit the linear splines. Here we investigate different number of knots and show their effect on the method's performance (Table 1). Varying the number of knots does not impact the performance substantially. This can be explained by the fact that almost all text lines in the evaluation datasets are horizontal. However, linear spline fitting makes the proposed method suitable for curved text lines (Kurar Barakat et al. (2019)).

Table 1. Effect of number of knots on the performance.

	5 knots	10 knots	15 knots	20 knots
Line IU	99.33	100	100	100
Pixel IU	96.58	97.02	97.01	97.02

6.2. Effect of binarization

Binarizing gray scale blob lines using component tree can eliminate false ligatures between the blob lines. However, it can not detect the spurious lines that do not exceed the false ligature score because they are straight (Fig. 2(b)). Thus, we decided to explore effect of using Hysteresis thresholding which can eliminate the noise in surroundings of blob lines. Comparing with the results of default setting, it appears that problem of spurious blob lines is not as common as false ligatures (Table 2).

6.3. Effect of component evolution maps

A good estimation of text line height range is important to get adequate blob lines from the scale space representation. Therefore, we investigate effect of using Component Evolution Maps (CEM) (Biller et al., 2012) to estimate the range of text line height. CEM can reveal this information rigorously from non-binarized document images. Table 2 shows the results using CEM which are worse than the results using statistical estimation (Table 1). Because the interlinear glosses that exist in the gray scale document image mislead the CEM algorithm.

6.4. Effect of merging broken blob lines

When there is a big gap between the words of a line, the blob line can be disconnected. Thus, for each blob line, the direction vector that connects its end points is defined as the direction of the blob line. Two blob lines are merged if (a) the direction of the vector connecting the two components falls between the direction of the two components, and (b) their vertical distance is less than the average text line height. In the case a connected component c overlaps two blob lines, we relabel c by assigning each pixel in c to the closest blob line. Merging achieves an insignificant improvement (Table 2).

Table 2. Effect of various factors on the results.

	Hystresis	CEM	Merging
Line IU	96.02	69.27	100
Pixel IU	94.04	80.38	97.04

7. Results

We present results on six datasets. All the models use component tree binarization, statistical estimation for text line height range and post processing by merging the broken blob lines. Parzival and Saint Gall datasets contain historical handwritten documents whereas ICDAR 2009 and ICDAR 2013 datasets contain modern handwritings. To our best, proposed approach achieves the state of the art results (Tables 3–4) on these datasets, using the same model, without any parameter setting.

Table 3. Comparison of our method with state of the art results on Parzival and Saint Gall datasets.

Dataset	Method	PHR	FM
Parzival	Baechler et al. (2013)	96.30	96.40
	Our method	**98.31**	**97.88**
Saint Gall	Diem et al. (2013)	98.94	99.03
	Our method	**99.08**	**99.22**

Table 4. Comparison of our method with state of the art results on ICDAR2009 and ICDAR2013 datasets.

Dataset	Method	RA	FM
ICDAR2009	Gatos et al. (2011)	99.50	99.53
	Our method	**99.70**	**99.69**
ICDAR2013	Stamatopoulos et al. (2013)	98.64	98.66
	Our method	**98.86**	**98.90**

DIVA-HisDB is a historical document dataset with many number of consequently touching text lines and heterogeneous in terms of text line height. Results on DIVA-HisDB dataset outperforms the best results in the competition (Table 5). The proposed algorithm can segment touching text lines with a line IU of 100. The degradation in pixel IU comes from conversion process of pixel labels into bounding polygons.

cBAD is a historical document dataset with a larger evaluation set and contains documents with varying layouts, originating from different time periods and locations. It is heterogeneous in terms of text line height and length. Our results on cBAD dataset underperforms the state of the art results due to binarization and conversion of pixel labels into baselines (Table 6). In fact cBAD dataset is not proper for binarization and designed to measure the performance using baselines that are independent of binary components. Besides, state of the art method on cBAD dataset is deep learning based and requires vast amount of labeled data for training.

8. Conclusion

This paper proposes an automatic and learning free text line segmentation algorithm that does not require to be adjusted for various handwritten document images. The algorithm is based on scale space theory and automatic scale selection which brings the advantage of working with variable text line heights. Range of text line heights in a document is estimated by statistical information of connected components. Automatic scale selection detects the blob lines that hover the text lines. Next, energy minimization extracts text lines with the help of detected blob lines. Proposed algorithm is robust as we presented results on a large range of datasets, including historical and modern documents. It is not a learning based algorithm but requires documents to be binarized which in turn causes pixel wise errors. Nevertheless this approach achieves several state of the art results. Future direction can investigate using a fully convolutional network to detect the blob lines.

Acknowledgments

This research was supported by the Frankel Center for Computer Science in Ben-Gurion University of the Negev.

References

Alaei, A., Pal, U., Nagabhushan, P., 2011. A new scheme for unconstrained handwritten text-line segmentation. Pattern Recognition 44, 917–928.

Antonacopoulos, A., Karatzas, D., 2004. Document image analysis for world war ii personal records, in: Document Image Analysis for Libraries, 2004. Proceedings. First International Workshop on, IEEE. pp. 336–341.

Baechler, M., Liwicki, M., Ingold, R., 2013. Text line extraction using dmlp classifiers for historical manuscripts, in: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on, IEEE. pp. 1029–1033.

Biller, O., Kedem, K., Dinstein, I., El-Sana, J., 2012. Evolution maps for connected components in text documents, in: Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR, pp. 405–410. doi:10.1109/ICFHR.2012.201.

Bukhari, S.S., Shafait, F., Breuel, T.M., 2009. Script-independent handwritten textlines segmentation using active contours, in: 2009 10th International Conference on Document Analysis and Recognition, IEEE. pp. 446–450.

Chen, K., Wei, H., Liwicki, M., Hennebert, J., Ingold, R., 2014. Robust text line segmentation for historical manuscript images using color and texture, in: Pattern Recognition (ICPR), 2014 22nd International Conference on, IEEE. pp. 2978–2983.

Clausner, C., Antonacopoulos, A., Pletschacher, S., 2012. A robust hybrid approach for text line segmentation in historical documents, in: Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE. pp. 335–338.

Cohen, R., Asi, A., Kedem, K., El-Sana, J., Dinstein, I., 2013. Robust text and drawing segmentation algorithm for historical documents, in: Proceedings of the 2nd International Workshop on Historical Document Imaging and Processing, ACM. pp. 110–117.

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

Table 5. Comparison of our method with state of the art results on DIVA-HisDB dataset.

	CB55		CSG18		CSG863		Overall	
	Pixel IU	Line IU	Pixel IU	Line IU	Pixel IU	Line IU	Pixel IU	Line IU
Simistira et al. (2017)	96.67	98.36	96.93	96.91	**97.54**	99.27	97.05	97.86
Our method	**97.04**	**100**	**98.14**	**98.38**	97.31	**100**	**97.50**	**99.46**

Table 6. Comparison of our method with state of the art results on cBAD dataset.

Dataset	Method	P	R	FM
cBAD	Diem et al. (2017)	**97.30**	**97.00**	**97.10**
	Our method	82.09	91.13	86.38

Delong, A., Osokin, A., Isack, H.N., Boykov, Y., 2012. Fast approximate energy minimization with label costs. International journal of computer vision 96, 1–27.

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

Diem, M., Kleber, F., Sablatnig, R., 2013. Text line detection for heterogeneous documents, in: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on, IEEE. pp. 743–747.

Fischer, A., Indermühle, E., Bunke, H., Viehhauser, G., Stolz, M., 2010. Ground truth creation for handwriting recognition in historical documents, in: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, ACM. pp. 3–10.

Garz, A., Fischer, A., Bunke, H., Ingold, R., 2013. A binarization-free clustering approach to segment curved text lines in historical manuscripts, in: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on, IEEE. pp. 1290–1294.

Garz, A., Fischer, A., Sablatnig, R., Bunke, H., 2012. Binarization-free text line segmentation for historical documents based on interest point clustering, in: Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on, IEEE. pp. 95–99.

Gatos, B., Louloudis, G., Stamatopoulos, N., 2014. Segmentation of historical handwritten documents into text zones and text lines, in: Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on, IEEE. pp. 464–469.

Gatos, B., Stamatopoulos, N., Louloudis, G., 2011. Icdar2009 handwriting segmentation contest. International Journal on Document Analysis and Recognition (IJDAR) 14, 25–33.

Gruening, T., Leifert, G., Strauss, T., Labahn, R., 2017. A robust and binarization-free approach for text line detection in historical documents, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), IEEE. pp. 236–241.

Kesiman, M.W.A., Burie, J.C., Ogier, J.M., 2016. A new scheme for text line and character segmentation from gray scale images of palm leaf manuscript, in: 15th International Conference on Frontiers in Handwriting Recognition 2016, At Shenzhen, China, pp. 325–330.

Kumar, J., Kang, L., Doermann, D., Abd-Almageed, W., 2011. Segmentation of handwritten textlines in presence of touching components, in: Document Analysis and Recognition (ICDAR), 2011 International Conference on, IEEE. pp. 109–113.

Kurar Barakat, B., Cohen, R., El-Sana, J., Rabaev, I., 2019. Vml-moc: Segmenting a multiply oriented and curved handwritten text line dataset, in: 3rd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR), IEEE. pp. 151–155.

Kurar Barakat, B., Droby, A., Kassis, M., El-Sana, J., 2018. Text line segmentation for challenging handwritten document images using fully convolutional network, in: 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE. pp. 374–379.

Lindeberg, T., 1998. Feature detection with automatic scale selection. International journal of computer vision 30, 79–116.

Mehri, M., Héroux, P., Gomez-Kramer, P., Boucher, A., Mullot, R., 2013. A pixel labeling approach for historical digitized books, in: 12th International Conference on Document Analysis and Recognition, IEEE. pp. 817–821.

Moysset, B., Kermorvant, C., Wolf, C., 2017. Full-page text recognition: Learning where to start and when to stop, in: Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, IEEE. pp. 871–876.

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

Moysset, B., Louradour, J., Kermorvant, C., Wolf, C., 2016. Learning text-line localization with shared and local regression neural networks, in: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE. pp. 1–6.

Naegel, B., Wendling, L., 2010. A document binarization method based on connected operators. Pattern Recognition Letters 31, 1251–1259.

Oliveira, S.A., Seguin, B., Kaplan, F., 2018. dhsegment: A generic deep-learning approach for document segmentation, in: 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE. pp. 7–12.

Ouwayed, N., Belaïd, A., 2012. A general approach for multi-oriented text line extraction of handwritten documents. International Journal on Document Analysis and Recognition (IJDAR) 15, 297–314.

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

Renton, G., Soullard, Y., Chatelain, C., Adam, S., Kermorvant, C., Paquet, T., 2018. Fully convolutional network with dilated convolutions for handwritten text line segmentation. International Journal on Document Analysis and Recognition (IJDAR) 21, 177–186.

Rother, C., Kolmogorov, V., Blake, A., 2004. Grabcut: Interactive foreground extraction using iterated graph cuts, in: ACM transactions on graphics (TOG), ACM. pp. 309–314.

Roy, P.P., Pal, U., Lladós, J., 2012. Text line extraction in graphical documents using background and foreground information. International Journal on Document Analysis and Recognition (IJDAR) 15, 227–241.

Saabni, R., Asi, A., El-Sana, J., 2014. Text line extraction for historical document images. Pattern Recognition Letters 35, 23–33.

Saabni, R., El-Sana, J., 2011. Language-independent text lines extraction using seam carving, in: Document Analysis and Recognition (ICDAR), 2011 International Conference on, IEEE. pp. 563–568.

Shapiro, V., Gluhchev, G., Sgurev, V., 1993. Handwritten document image segmentation and analysis. Pattern Recognition Letters 14, 71–78.

Shi, Z., Govindaraju, V., 2004. Line separation for complex document images using fuzzy runlength, in: null, IEEE. p. 306.

Simistira, F., Bouillon, M., Seuret, M., Würsch, M., Alberti, M., Ingold, R., Liwicki, M., 2017. Icdar2017 competition on layout analysis for challenging medieval manuscripts, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), IEEE. pp. 1361–1370.

Stamatopoulos, N., Gatos, B., Louloudis, G., Pal, U., Alaei, A., 2013. Icdar 2013 handwriting segmentation contest, in: 2013 12th International Conference on Document Analysis and Recognition, IEEE. pp. 1402–1406.

Swaileh, W., Mohand, K.A., Paquet, T., 2015. Multi-script iterative steerable directional filtering for handwritten text line extraction, in: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, IEEE. pp. 1241–1245.

Witkin, A.P., 1987. Scale-space filtering, in: Readings in Computer Vision. Elsevier, pp. 329–332.

Wong, K.Y., Casey, R.G., Wahl, F.M., 1982. Document analysis system. IBM journal of research and development 26, 647–656.

Yin, F., Liu, C.L., 2009. Handwritten chinese text line segmentation by clustering with distance metric learning. Pattern Recognition 42, 3146–3157.