Text Line Segmentation of Handwritten Documents by Area Mapping

Abhijeet Boragule*, GueeSang Lee*

Abstract

Text line segmentation is a preprocessing step in OCR, which can significantly influence the accuracy of document analysis applications. This paper proposes a novel methodology for the text line segmentation of handwritten documents. First, the average width of the connected components is used to form a 1-D Gaussian kernel and a smoothing operation is then applied to the input binary image. The adaptive binarization of the smoothed image forms the final text lines. In this work, the segmentation method involves two stages: firstly, the large connected components are labelled as a unique text line using text line area mapping. Secondly, the final refinement of the segmentation is performed using the Euclidean distance between the text line and small connected components. The group of uniquely labelled text candidates achieves promising segmentation results. The proposed approach works well on Korean and English language handwritten documents captured using a camera.

Keywords: Text-line Segmentation|Gaussian Blurring|Euclidean Distance|Handwritten Documents.

I. INTRODUCTION

Numerous methods have been proposed for the process of handwritten text line segmentation, which plays an important role in OCR and document analysis applications. Analyzing handwritten documents is challenging, because of the variability in handwriting, skew, and orientation and the overlapping of the text characters. Various methods have been proposed for the segmentation of handwritten text lines in different languages. The text line segmentation can be represented by the text area line, string, cluster, string or baselines [1].

Handwritten characters show variations in writing styles and multilingual properties. Several previously proposed methods are discussed as follows: In Telugu printed document text line segmentation; Koppla et al [2] presented a fringe map based technique for text line segmentation; in the fringe map, each pixel of the binary image is associated with a fringe number that denotes the distance to the nearest black pixel. The fringe map value is used to segment the text lines. The PNF is located in the fringe distance map. The Joining PNF method makes a region between the lines. The fringe map method can also deal with Kannada documents. In fringe map method, several heuristics rules are used to deal with noise which is not perfect for all linguistics documents. This method can eliminate small text candidate because of the noise in fringe map.

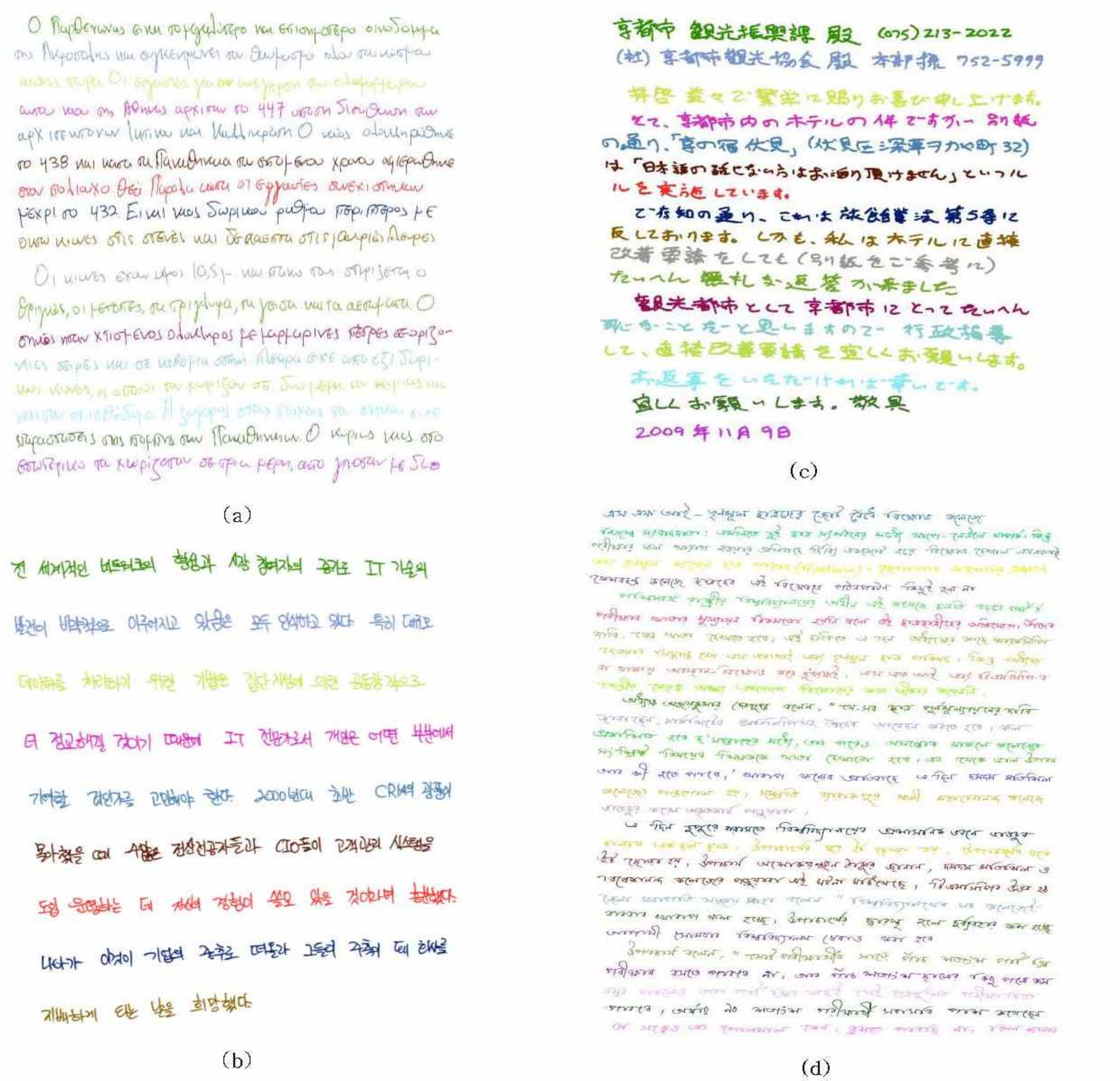
Vassilis Papavassiliou et al. [3] presented a combination of special structural elements, in which the morphological approach represents the text line segmentation. This method is based on heuristic rules.

The Watershed algorithm is used to cover the text line area and the connected components. The Euclidean distance tracing method uses less heuristics rules. The morphological operation can select the components from another line because of the wrong selection of structuring element.

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by MEST(2015-024474) and by the technological innovation R&D Program of SMBA(S2173771).

*Member, Dept. of ECE, Chonnam National University, Korea.

Manuscript : 2015.07.01
Revised : 2015.09.23
Confirmation of Publication : 2015.09.25
Corresponding Author : Gueesang Lee,
e-mail : gslee@jnu.ac.kr



(a)

(b)

(c)

(d)

Fig. 1. Text line segmentation results of arbitrarily selected (a) English (b) Korean (c) Japanese and (d) Indian Bangla handwritten documents

:lausner C. et al [4] proposed a combination of 'o methods based on Projection profile analysis iles and connected components analysis. The ›nnected components are grouped and the large ›mponents are split under the projection profiles.

Raid Saabni et al. [5] proposed the seams carving ethod, in which the distance transform of the nary document and its energy map of all different ›ams in the document are classified as a text line. hen, dynamic programming is used to compute the inimum energy from the left to right paths. Vasant Manohar et al [6] presented the graph clustering method, in which an undirected graph is constructed. In this method, the nodes correspond to the connected components. A minimum value is used to partition the nodes in the graph and each cluster is considered as a unique line. The method was applied to large document images and showed significant improvements. This method is generally based on ensemble documents.

Syed B. et al.[7] proposed an isotropic Gaussian filter bank for text line extraction. A multi oriented anisotropic Gaussian filter is smoothed over the input image and the text lines are extracted from the smoothed region. This method can be applied to handwritten text line segmentation, but overlapping and skewness can affect its accuracy. Generally, a significant preprocessing step is needed to locate the text area.

The consecutive text candidates can be traced in the text area. Yi Li et al [8] smoothed the binary image in the grayscale with a rectangular window. The enhanced text line level set algorithm is used to classify the text lines into unique ones. However, the previous algorithms are still not perfect for selection of small components (dot, commas, and inverted commas). Generally, commas and small connected components are located to nearest text line. Our algorithm use minimum Euclidean distance from the main text line to label a small component. The fast and effective labeling strategy performs better segmentation operation.

The 1-D blurring with binary document and the Euclidean tracing of connected components of text line area achieves better segmentation results with multilingual documents.

The remainder of this work is organized as follows. In section 2, the proposed method for text line segmentation is explained. In section 3, the method of evaluating the performance and the experimental results are described. In section 4, the conclusion is presented.

II. PROPOSED METHOD

In this paper, we proposed a novel methodology for handwritten text line segmentation. The proposed segmentation process is divided into two stages. In the first stage of segmentation, each text candidate area is mapped with the text line image. In the second stage, for the final refinement of segmentation, each text candidate is selected based on the closest Euclidean distance. The labeling strategy is used in this work. The uniquely labeled text candidate is used for final segmentation. A description of the proposed approach is given in the following section.

2.1 Text Line Formation

Initially, the average height of the connected components is estimated from the entire binary document. In this step, our goal is to produce the text lines. To do this, a 1-D Gaussian kernel [1 Average width*10] is created. The 1-D Gaussian kernel is treated with the input binary image. The reason for applying Gaussian blurring is to blur the text candidates so the adaptive binarization text line can be easily formed later.

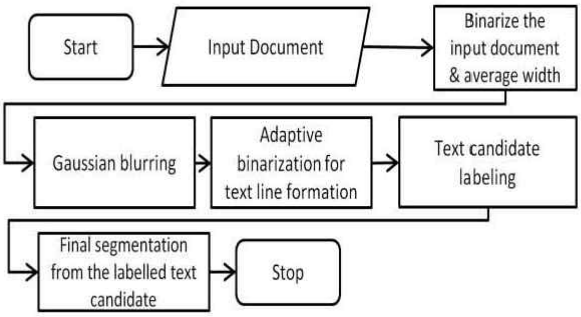

Start
Input Document
Binarize the input document & average width
Gaussian blurring
Adaptive binarization for text line formation
Text candidate labeling
Final segmentation from the labelled text candidate
Stop

Fig. 2. Basic flow chart of the proposed method

The width of the Gaussian kernel is 10 times the average width. This parameter is obtained from tests conducted on the dataset to obtain the best performance. In handwritten documents, some text candidates are overlapped and some are skewed. Considering this problem, Gaussian blurring and adaptive binarization are sufficient to select only the text lines and avoid overlapping.

In figure 3, the binary image is overlapped with the blurred image for better observation. As shown in the yellow box, the overlapped text candidate region is not blurred, because of the lower number of pixels in the overlapped area. For the final text line, adaptive binarization is applied to the blurred image.

The noise in the final binary text line is removed to get the exact total number of lines. In figure 4, figure 4(b) is the blurred image (Grayscale) after adaptive binarization of the text lines to form figure 4(c). In figure 4(c), all of the text lines are used in further processing for the purpose of labelling a text candidate and segmenting the text candidate.

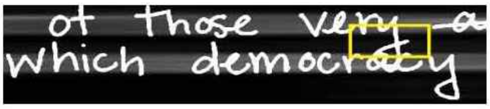
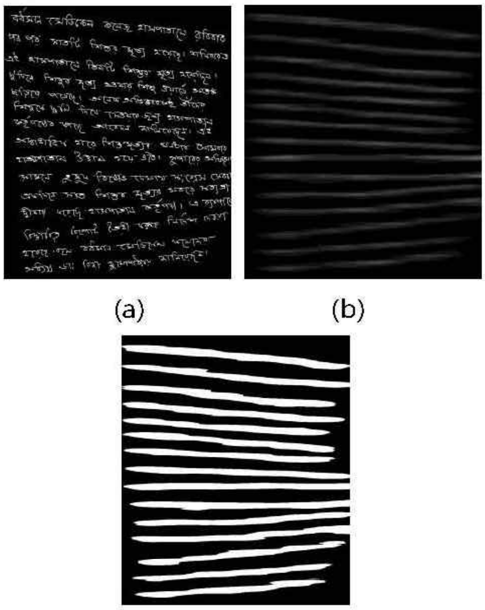


Fig. 3. Overlapping of binary image and blurred image

The aim of applying adaptive binarization to the binary blurred image is to select the high confidence pixels in order to form binary lines.

Those pixels which have low intensity in the blurred image (Grayscale) are not binarized. In figure 3, the yellow box shows that the overlapped region is not well blurred, and so will not be able to be binarized in the adaptive binarization process. After adaptive binarization with the blurred image, the connected component algorithm is used to get the exact number of lines.


(a) (b)

(c)

Fig. 4. (a) Input binary image (b) Gaussian Blurring (c) Adaptive binarization

2.2 Area Mapping & Euclidean distance

In this algorithm, a labelling strategy is used to label the text candidate. After the formation of the text lines, each of them is divided vertically. The algorithm selects each text line from figure 4(c) and performs the labelling process. This labelling strategy is very convenient for the final segmentation.

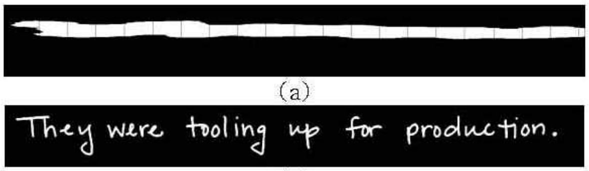
In the beginning, a text line is selected, as shown in figure 4(c). After that, the selected text line is divided vertically, as shown in figure 5 (a). The vertically divided connected components and the locations of their centroids are sorted from left to right based on their x coordinates. The vertically divided components help to treat skewed lines and recover small dots. The most serious problem is to segment the small dots, commas and inverted commas because of their effect on the accuracy of the algorithm. Algorithm 1 shows the pseudo code of the system.

$$EuclideanDistance = \sqrt{(x1-x2)^2 + (y1-y2)^2} \quad -\text{eq. (1)}$$

In the beginning, the labels and weights are set to 0. The procedure of labelling the connected components is performed in two steps. The first step is based on area mapping. In our approach, the term area mapping means selecting each text candidate (Connected component) and the coordinates of its bounding box and mapping it with another candidate in the text line image area, as shown in figure 5(c). If pixels are present inside of the bounding box area, then the total number of pixels is computed and used as the weight. If the weight is greater than 0, then the connected component is labelled with the respective line number.

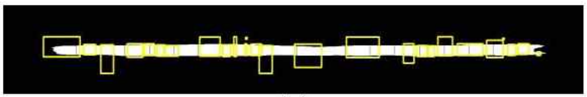
Algorithm 1 Text candidate labelling algorithm

```
Set all BoudingBox.label=0;
For linenumber to textlines;
Divide Components Vertically;
For   i=0 to BoundingBoxes   do
Weightcount=0;
Weightcount =Map Bounding box area with selected
  Text line;
 If(lineWeight<Weightcount)
      lineweight= Weightcount;
      BoudingBox.label= i +1;
    End if
  If(BoudingBox.label=0 )
     MinEdist=10000;
      For centroid in verticallydividedComponents   do
          TempCenter1= BoundingBoxCenter;
          TempCenter2= Centroid;
   Normdist=EuclideanDistance(Centroid,BoundingBo
   Center);
          If (MinEdist< Normdist)
              MinEdist= Normdist;
               BoudingBox.label= i;
          End if
          BoundingBoxCenter =TempCenter1;
         Centroid =TempCenter2;
  End For
 End if
 End For
 End For
```



(a)

(b)



(c)

Fig. 5. (a) Vertically divided text line (b) Binary Image (c) Connected components mapping with text line image

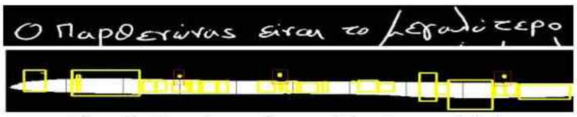


Fig. 6. Tracing of small text candidates

The weight is stored in a temporary variable and checked in the next iteration. In the second stage of the algorithm, if the text candidate is not in the range of the text line, then the Euclidean distance is checked with the vertically divided connected components. At this stage, if the connected component is not labelled because the condition fails in the first step, then the Euclidean distance from each connected component centroid to each vertically divided centroid is computed using eq. (1) and the connected component is labelled with the selected text line number.

The process of selecting each connected component is repeated until all of the text candidates (Connected components) are labelled with the respective line number. The Euclidean distance is used to perform the final refinement of the labelling of the text candidates. The Euclidean distance is used to select the nearest text candidates, such as dots and inverted commas which are not in the text line and label them with the respective line number. In figure 6, a demonstration of small components is shown with a red bounding box. Figure 2 shows the basic flow of the proposed algorithm.

III. EXPERIEMENT RESULTS

The experiment was conducted with all 150 images from the ICDAR 2013 Handwritten Segmentation Contest [9] dataset with a real time application environment based on C++. The average time taken to process this dataset using our system is 5.20 sec on a 3.20 GHz processor. On the ICDAR 2013 dataset, a standard resolution of 1024x768 and minimum distance of 10,000 (pixels) are selected for better performance.

The main parameter of the Gaussian kernel is selected to be 10 times the average width based on an evaluation with the database. The ICDAR dataset covers a wide range of cases, including handwritten English and Indian Bangla script [9]. The performance evaluation is based on matching the entities detected by the algorithm with the ground truth.

$DR= o2o/N$, $RA= o2o/M$, $FM= 2DR\ RA/DR+RA$

Table 1. Comparison between methods used in ICDAR 2013 Handwritten text line segmentation contest [9] and our method.

	M	*o2o*	*DR(%)*	*RA(%)*	*FM(%)*
MSHK	2696	2428	91.66	90.06	90.85
QATAR-a	2626	2404	90.75	91.55	91.15
QATAR-b	2609	2430	91.73	93.14	92.43
CVC	2715	2418	91.28	89.06	90.16
NSCR(SoA)	2646	2447	92.37	92.48	92.43
Our Method	**2615**	**2475**	**93.43**	**94.64**	**94.03**

In the dataset, there are a total of 2,649 text lines. In our method, 2615 lines are detected. In the Matchscore(i,j) table, the j-th (line) represents the ground truth region and the i-th (line) represents the resulting region. If M and N are the numbers of ground-truth then the result elements are respectively, DR (Detection rate), RA (Recognition accuracy) and FM (Harmonic mean) are calculated as in the ICDAR 2013 Handwritten segmentation contest [9].

Table 1 shows a comparison between the proposed method and other state of the art methods. The participants are from ICDAR 2013 Handwritten text line segmentation contest [9]. The area mapping of the text line and Euclidean tracing of the small connected components produce better results than the other state-of-the-art text line segmentation methods. In figure 1, the proposed method can generate good results on multilingual images having variations in handwriting skew and character size.

IV. CONCLUSION

This paper presents a method of segmenting handwritten text lines. First, the average width is calculated. Then, a Gaussian blurring operation is performed on the input binary image. After that, adaptive binarization is applied to form the text

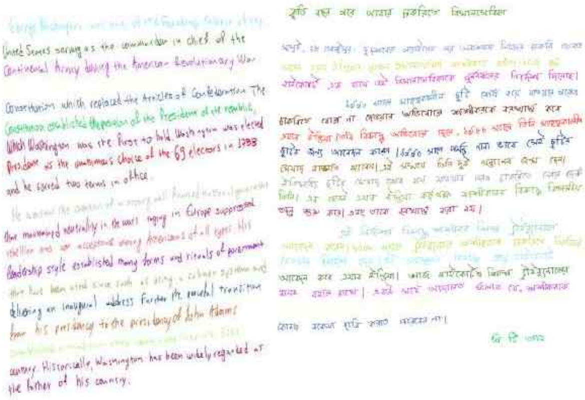


Fig. 7. Text line segmentation results

lines. The connected components are labelled using area mapping with the text line and the Euclidean distance tracing of the text candidate. The experimental results show promising results on the well-known ICDAR 2013 dataset.

REFERENCES

[1] Likforman-Sulem, L.; Zahour, A.; Taconet, B.; "Text line segmentation of historical documents: a survey," *Int. J. Doc. Anal. Recognit.*, vol.9, no.2, pp.123-138, 2007.

[2] Kumar, V.; Negi, A.; "Fringe Map Based Text Line Segmentation of Printed Telugu Document Images," *Document Analysis and Recognition (ICDAR)*, 2011, pp.1294-1298, Sep 2011.

[3] Papavassiliou, V.; Katsouros, V.; "A Morphological Approach for Text-Line Segmentation in Handwritten Documents," *12th International Conference on Frontiers in Handwriting Recognition* 2010, pp.16-24, Nov 2010.

[4] Clausner, C.; "A Robust Hybrid Approach for Text Line Segmentation in Historical Documents," *International Conference on Pattern Recognition (ICPR)* 2012, pp.335-338, Nov 2012.

[5] Saabni, R.; El-Sana, J.; "Language-Independent Text Lines Extraction Using Seam Carving," *International Conference on Document Analysis and Recognition (ICDAR)* 2011, pp.563-568, Sep 2011.

[6] Manohar, V.; Vitaladevuni, S.N.; Cao, H.; Prasad, R.; Natarajan, P.; "Graph Clustering-based Ensemble Method for Handwritten Text Line Segmentation," *International Conference on Document Analysis and Recognition (ICDAR)* 2011, pp.574-578, Sep 2011.

[7] Bukhari, S.S.; Shafait, F.; Breuel, T.M.; "Text-Line Extraction using a Convolution of Isotropic Gaussian Filter with a Set of Line Filters," *International Conference on Document Analysis and Recognition (ICDAR)* 2011, pp.579-583, Sep 2011.

[8] Li, Y.; Zheng, Y.; Doermann, D.; "Detecting Text Line in Handwritten Documents," *International Conference on Pattern Recognition (ICPR)*, 2012, pp.1030-1033, 2006.

[9] Stamatopoulos, N.; Gatos, B.; Louloudis, G.; Pal, U.; Alaei, A.; "ICDAR 2013 Handwriting Segmentation Contest," *12th International Conference on Document Analysis and Recognition (ICDAR)*, 2013, pp.1402-1406, Aug 2013.

Authors

Boragule Abhijeet received the Bachelor in Computer Science from Shivaji University, Kolhapur City, India in 2012. He is currently a MS student in Department of Electronics and Computer Engineering in Chonnam National University, S. Korea. His research interests are multimedia and image processing, vision tracking, and pattern recognition.

GueeSang Lee received the B.S. degree in Electrical Engineering and the M.S. degree in Computer Engineering from Seoul National University, Korea in 1980 and 1982, respectively. He received the Ph.D. degree in Computer Science from Pennsylvania State University in 1991. He is currently a professor of the Department of Electronics and Computer Engineering in Chonnam National University, Korea. His research interests are mainly in the field of image processing, computer vision and video technology.