Journal of Universal Computer Science, vol. 17, no. 1 (2011), 16-29
submitted: 8/4/10, accepted: 12/12/10, appeared: 1/1/11 © J.UCS

Text Line Detection and Segmentation: Uneven Skew Angles and Hill-and-Dale Writing

Ergina Kavallieratou
(University of the Aegean, Karlovasi, Samos, Greece
kavallieratou@aegean.gr)

Fotis Daskas
(Greek Open University, Patra, Greece
daskasfotis@hotmail.com)

Abstract: In this paper a line detection and segmentation technique is presented. The proposed technique is an improved version of an older one. The experiments have been performed on the training dataset of the ICDAR 2009 handwriting segmentation contest in order to be able to compare, objectively, the performance of the two techniques. The improvement between the older and newer version is more than 24% while the average extra CPU time cost is less than 200 ms per page.

Keywords: Line detection, line segmentation, OCR
Categories: I.7.5

1 Introduction

The text line segmentation is a preprocessing task for OCR and Document Image Processing Systems. Although the printed text line detection and segmentation is considered an easy task due to white streams, present in printed pages, the handwritten text line detection and segmentation is still considered an open problem due to the special circumstances of handwritten documents, such as the presence of uneven skew angles in the same document image, differences in space, style and character size, hill-and-dale writing.

In this paper we present the improvement of a technique for line segmentation used in the system [kavallieratou et al, 02]. The mentioned system is a complete system of document image processing and OCR. The goal of the proposed version is to improve the performance of the line segmentation technique in the presence of uneven skew angles on the same page or even on the same text line (hill-and-dale writing).

This paper is organized as follows: the section 2 refers some previous work in the field, while the proposed technique and its subtasks are presented in section 3. Its evaluation and its comparison to the previous version are given in section 4, and finally, we conclude in section 5.

2 Previous work

In [Razak et al, 08], a classification of the text line segmentation techniques is proposed:
i) Projection-based approaches,
ii) Hough Transform methods,
iii) Smearing methods,
iv) Grouping methods,
v) Active Contour methods,
vi) Graph-based methods.

The projection-based approaches are making use of the structural characteristics of the documents. They are top-down techniques, simple and easy in implementation. The black pixels are projected on the vertical axis. The resulted histogram consists of regions with larger and lower concentrations of pixels. This methodology is used as main or auxiliary in [Bruzzone et al, 99], [Tripathy et al, 04], [Bar-Yosef et al, 09], [Pal et al, 03], [Zahour et al, 01], [dos Santos et al, 09], [Arivazhagan et al, 07].

Hough Transform is also a popular methodology in the area of text line segmentation ([Pu et al, 98], [Louloudis et al, 06], [Louloudis et al, 09]). It describes parametric geometric shapes and identifies geometric locations that suggest the existence of the sought shape. The purpose of this technique is to detect fuzzy snapshots of objects in a certain category of shapes and under a voting procedure. The voting procedure takes place in a parametric space where the candidate objects are obtained as local maxima in a table made explicitly by the Hough transform. Serious drawback of this method is the computational complexity.

The smearing methodology is a bottom-up technique ([Stamatopoulos et al, 08], [Shi et al, 2005]). It is the process of converting a set of background pixels located between foreground pixels into foreground pixels whether their amount is less than a certain threshold. Smearing methods strengthen by local techniques, solve specific problems and overlapping touched connected component. Moreover, these methods work successfully with documents that contain characters of variable height. However, they may have problems in the presence of skewing. Similarly, they cannot handle the variability in distances between words and characters. They usually make use of many thresholds and heuristic rules.

As it is obvious the grouping methods are also bottom-up. From the lower level, the pixel, starts a process of grouping according to specific constrains designed to result to a layer of text lines. The process is relatively easy in the case of printed documents, but it may be proved to be difficult and problematic in manuscripts. A well known study is [Feldbach et al, 01].

The Active Contour methods use the difference between the foreground and the background through characteristics such as brightness or color that occurs at the border contours of the object. The edge is a curved line from which derive all the properties and characteristics that describe the specific category of shapes, in our case text lines. It is well founded [Kass et al, 88] that if we create a specific curve around the contour of an object and then impose the appropriate equation of motion, it will force it to reach the curve forming the outline border of the object. These curves are called Active Contours and they have been used widely in text line segmentation ([Bukhari et al, 09], [Li et al, 08], [Du et al, 09]).

The representation of document images by graphs is an important tool of the line segmentation procedure. The graph is constructed as vertices of pixel or more complex connected components. The vertices are normally associated with weighted edges that depict distances between connected components. After the modelling of the document image, the treatment method can be chosen ([Yin et al, 07], [Boykov et al, 01], [Kumar et al, 06]).

You can see an objective evaluation of text line segmentation techniques in [Gatos et al, 07] and [Gatos et al, 09].

3 The Proposed Technique

The proposed technique consists of the sub-tasks of Figure 1, namely, segment estimation, text line detection and text line segmentation.

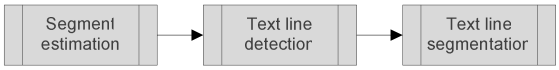


Figure 1: The proposed technique.

In segment estimation, the system splits the original document image to the necessary number of VERTICAL ZONEs of equal width (Figure 2) in order to cope with the existence of different skew angles in the page or even in the same text line (hill-and-dale writing). The number of VERTICAL ZONES is determined by the system and depends on the variety of skewing in the page. For VERTICAL ZONE=1, the document image remains in one piece. In the same task, the positions of STARTs, ENDs, and MINIMAs are also estimated by the histogram of the VERTICAL ZONEs. Histogram (Figure 2) is the horizontal projection profile of each VERTICAL ZONE after applying 5-point smoothing. This smoothing is done in order to avoid small ups and downs (noise) due to the handwriting irregularity. START (Figure 3) is the beginning of a text line and it is determined by the point that the histogram value rises over a LIMIT. There is a different START for each text line i symbolized as START[i]. Similarly, END (Figure 3) is the end of a text line and it is determined by the point that the histogram value falls under a LIMIT. There are different ENDs for each text line symbolized as END[i]. The area that is determined by the END of the previous text line and the START of the next one is called SEGMENT AREA (Figure 3). Finally, MINIMA is the minimum of histogram in the SEGMENT AREA and determines the point that the text line detection procedure will start in the current VERTICAL ZONE. As there is a different MINIMA in each SEGMENT AREA, it is symbolized as MINIMA[i].

The line detection task proceeds with the localization of all the points that will be part of the segment between the text lines in SEGMENT AREAs. The goal is to establish a segment between the text lines that includes as many white pixels as possible while dealing with the presence of black ones whenever is impossible to avoid them (e.g. touching ascenders and descenders). This task makes use of the estimations of the previous task for the SEGMENT AREAs and takes into account the limits of the VERTICAL ZONEs and the estimated MINIMA whenever is necessary.

In fact it examines the SEGMENT AREAs in detail using the estimations as guides, only when it is necessary.

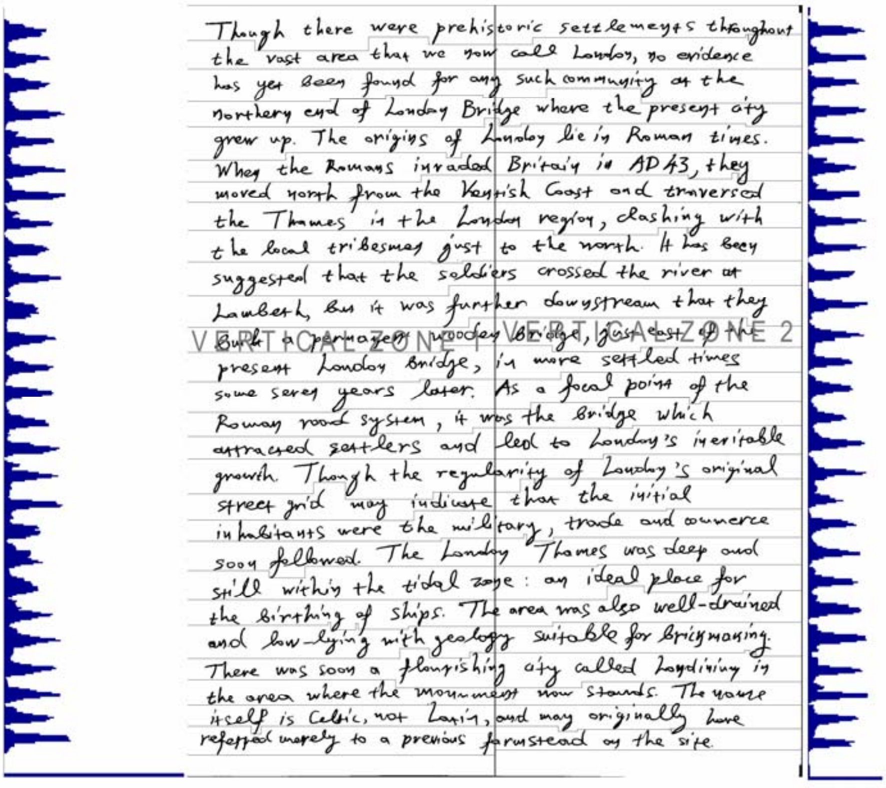


Figure 2: An example of document image with its necessary VERTICAL ZONEs and the corresponding histograms.

Finally, the line segmentation task prepares the system output according to the application. Considering the segment localization by the previous task either it draws it on the original image (figure 2) or it copies each text line and saves it to distinct image files.

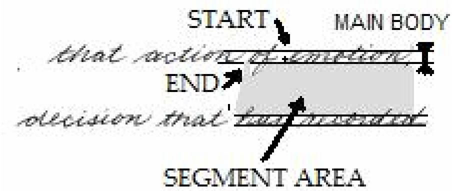


Figure 3: Basic terms of the methodology.

3.1 Segment Estimation

The segment estimation procedure is shown in Figure 4.

At first the VERTICAL ZONE is set to 1 (whole document image). The histogram of the page is calculated and smoothed. As first LIMIT, the tenth of the maximum of the histogram is set:

$$LIMIT = \max(histogram)/10 \tag{1}$$

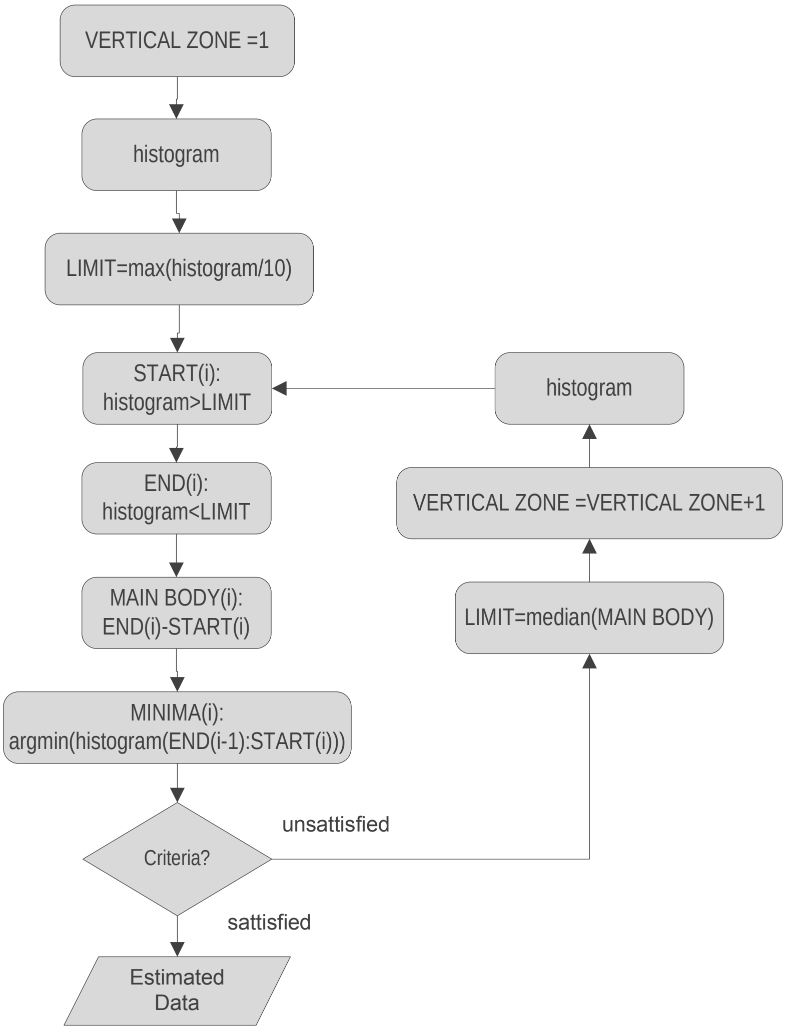


Figure 4: The Segment Estimation procedure.

First the START and END points are estimated by checking the smoothed histogram and noting the points that the histogram values cross the LIMIT. If between two successive values there is a histogram peak the first is START point and the second END point, otherwise vice versa. As it is already mentioned, the area between an END point and the next START one is a SEGMENT AREA, which is the area that the segment is expected to be found. On the other hand, the distance between a START point and the next END one is expected to be the value MAIN BODY (figure 3) in accordance to [Kavallieratou et al, 02]:

$$MAINBODY = \underset{i=1}{\overset{N}{median}}(END(i) - START(i)) \quad (2)$$

As MAIN BODY is considered the height of a word without the ascenders and the descenders (figure 3) and it is a very important characteristic of a person's writing [Kavallieratou et al, 04]. In this application, it is used as an indication of the document image resolution and the character size. Since various values of MAIN BODY can be extracted for each i from the corresponding ENDs and STARTs, the median value is finally considered (eq.2). The LIMIT is set to MAIN BODY for VERTICAL ZONE>1.

Next, the MINIMA are calculated as the points that histogram is minimized in the SEGMENT AREA. These points are considered to be the possible starting points for the text line segments as we are going to see in the text line detection subtask (§3.2).

As it is already mentioned, in order to build an algorithm able to deal with uneven skews between text lines or even in the same text line (hill-and-dale writing) in handwritten documents, it is necessary to have the document image split in VERTICAL ZONEs of equal width (figure 2) in a way that each VERTICAL ZONE would include text without skew at all or few enough to be able to handle. The question is how many vertical zones. If we use few VERTICAL ZONEs the above mentioned problem is not solved, while many and unnecessary VERTICAL ZONEs would cumber the computational cost. The proposed system detects the necessary number of VERTICAL ZONEs by incrementing the amount of zones by one in each repetition, splitting vertically the page in zones of equal width and repeating the segment estimation procedure for each VERTICAL ZONE as described above using the local histogram (of each zone), till the termination criterion is satisfied.

In case that VERTICAL ZONE > 1, it is necessary to find a correspondence between the MINIMAs of the different zones in order to establish the continuity of the segments from the left part of the page to the right one. The correspondence algorithm is shown in detail in Figure 5. In brief, the algorithm establishes correspondence between the MINIMAs that they form rectangular that includes the less black pixels.

```
Do
   Estimate LEFTs: MINIMA of LEFT_VERTICAL_ZONE
   Estimate RIGHTs: MINIMA of RIGHT_VERTICAL_ZONE
   While LEFT & RIGHT
      LM=middle_LEFT_VERTICAL_ZONE
      RM=middle_RIGHT_VERTICAL_ZONE
      Specify RECTANGULAR(LEFT,LM,RIGHT,RM)
      BLACK_PIXEL=count black pixels in RECTANGULAR
      BLACK_PIXEL1=BLACK_PIXEL-1
      While BLACK_PIXEL1<BLACK_PIXEL
         BLACK_PIXEL=BLACK_PIXEL1
         nextRIGHT
         Specify RECTANGULAR(LEFT,LM,RIGHT,RM)
         BLACK_PIXEL1=count black pixels in RECTANGULAR
      Correspond (LEFT,RIGHT)*
   While LEFT & ¬RIGHT
      RIGHT=LEFT
      nextLEFT
   While ¬LEFT & RIGHT
      LEFT=RIGHT
      nextRIGHT
While any BLACK_PIXEL> LIMIT*number_of_included_pixel_lines
```

Figure 5. Correspondence algorithm: LEFT and RIGHT are the current MINIMAs from left and right areas, respectively. The commands nextLEFT, nextRIGHT means 'proceed to next MINIMA'. By Correspond, the correspondence of current MINIMAs is indicated.

Similar is the termination criterion. In more detail, if VERTICAL ZONE=1, the termination criterion is satisfied when the histogram at MINIMAs is always less than the LIMIT. Otherwise if VERTICAL ZONE >1, the criterion is satisfied, when all the rectangular formed by two vicinal MINIMAs as y-coordinates and the middle of the corresponding zones as x-coordinates, include less black pixels than:

*LIMIT*number_of_included_pixel_lines*

3.2 Text Line Detection

The text line detection starts from the estimated MINIMA of the leftmost VERTICAL ZONE extracted in the segment estimation subtask. As already mentioned SEGMENT AREA is considered the area between each estimated END and the next estimated START (gray area in figure 7). Each time the system proceeds to the next VERTICAL ZONE, the corresponding STARTs and ENDs, as defined by the correspondence algorithm (figure 5) are considered to delimit the SEGMENT AREA (figure 7).

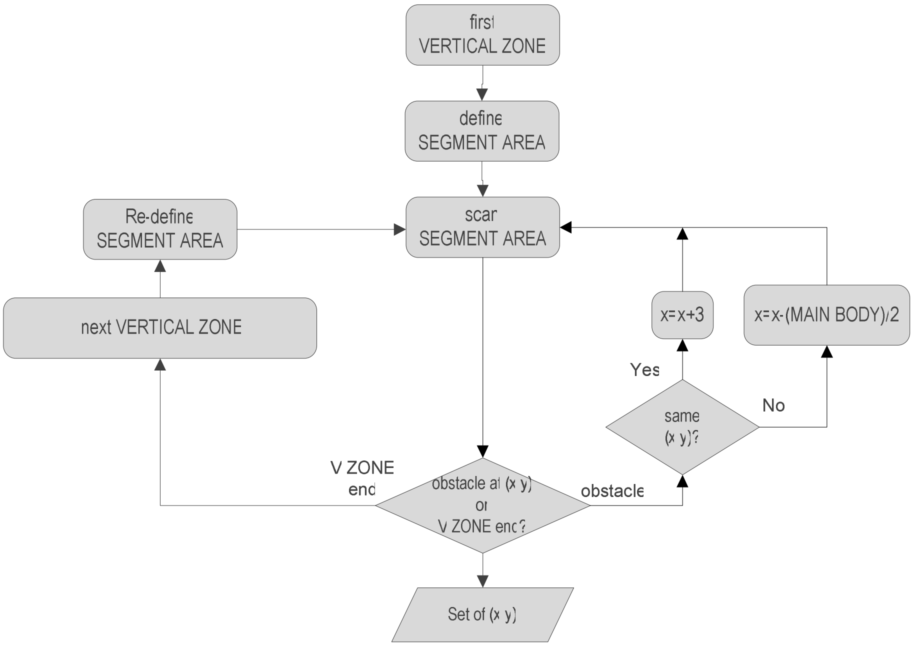


Figure 6: Text Line Detection Procedure.

Please remember that END and START are considered to be before the descenders and after the ascenders, respectively. This means that in the SEGMENT AREA there are ascenders and descenders and the SEGMENT AREA can vary slightly between VERTICAL ZONEs (gray areas in Figure 7).

The system starts the scanning of the pixel lines inside the SEGMENT AREA and the pixel line with the longest white run before any obstacle is kept. For this white pixel run, xy-coordinates of both of its edges are kept.

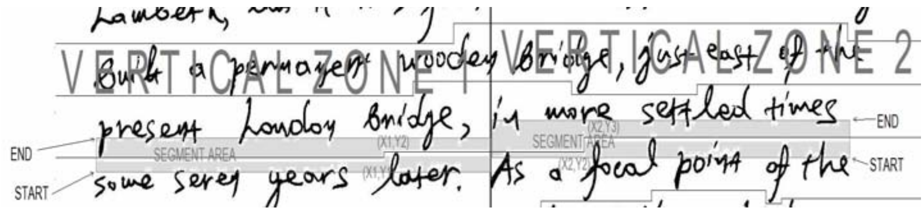


Figure 7: Basic terms of the line detection procedure.

In case that an obstacle is met, the system precedes backwards MAIN BODY/2 pixels ((x1,y1) in Figure 7) and the whole procedure is repeated. The backward action is done for the case that it is trapped inside an ascender or descender loop. If a longer run is not succeeded and the procedure stops at the same x-coordinate for a second time, the system precedes 3 pixels forward, considering that it is limited by touching ascenders and descenders across the SEGMENT AREA. This action will be repeated till the obstacle is crossed.

If the end of a VERTICAL ZONE is met, the line detection procedure continues into the next vertical zone, considering the new START and END as delimiters of the SEGMENT AREA. This way the procedure can deal better with the existence of different skews in the same text line. Please note that the check for the end of a VERTICAL ZONE happens only when an obstacle is met, otherwise it goes on straight e.g. in Figure 7 the line segment goes on straight in VERTICAL ZONE 2 at y2 till the point (x2,y2) that an obstacle is met, and then the new SEGMENT AREA is considered.

In Figure 6, the text line detection procedure is presented.

3.3 Text Line Segmentation

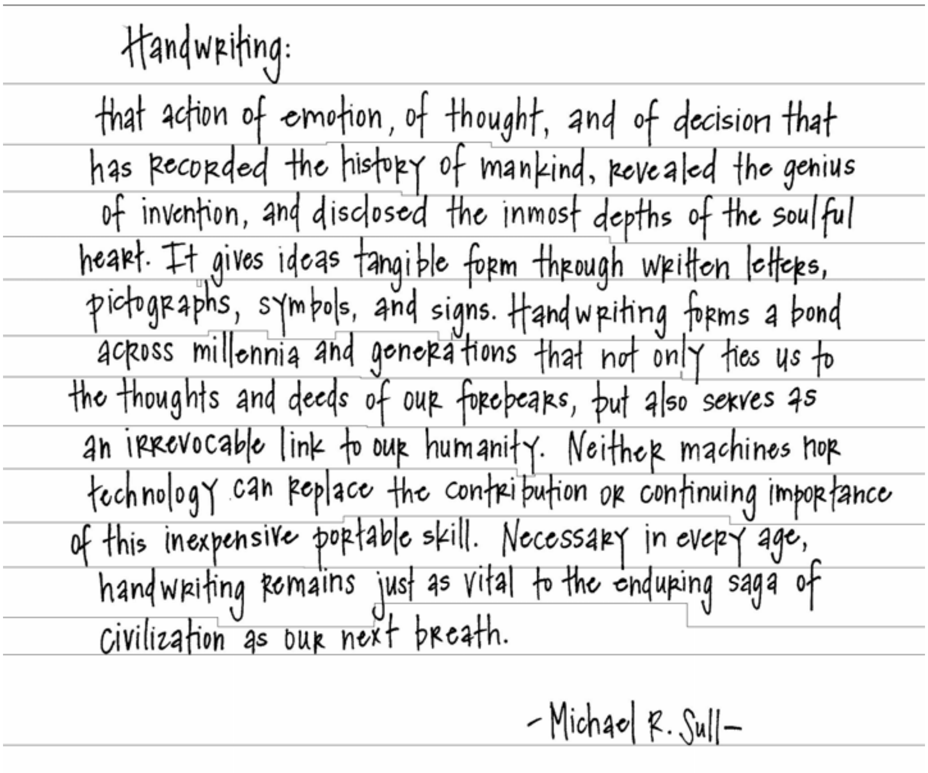
In text line segmentation subtask, the information extracted at the previous tasks is used according to the application. The image can be produced either appropriately for the ICDAR contest evaluation software, or designing the segments on the original image as the examples in this paper (Figures 2,9,10), or each text lines can be cut and saved in separate image files (Figure 8). In the last case, an empty matrix is created, with height equal to the difference between the maximum and minimum y-coordinates estimated by the text line detection procedure, and width equal to the page width.

In order to copy the corresponding area of the text line, the xy-coordinates extracted in text line detection, are considered. For the transition between the different white pixel runs, vertical cut on the same pixel line is considered.

Figure 8: Example of text line.

4 Evaluation

For the evaluation of our technique, the training dataset and the evaluation software of ICDAR2009 handwriting segmentation contest were used. The training dataset consists of 100 handwritten document images that *"came from (i) several writers that were asked to copy a given text; (ii) historical handwritten archives, and (iii) scanned handwritten document samples selected from the web. None of the documents included any non-text elements (lines, drawings, etc.) and were written in several languages including English, French, German and Greek."*[Gatos et al, 07].



Handwriting:
that action of emotion, of thought, and of decision that has recorded the history of mankind, revealed the genius of invention, and disclosed the inmost depths of the soulful heart. It gives ideas tangible form through written letters, pictographs, symbols, and signs. Handwriting forms a bond across millennia and generations that not only ties us to the thoughts and deeds of our forebears, but also serves as an irrevocable link to our humanity. Neither machines nor technology can replace the contribution or continuing importance of this inexpensive portable skill. Necessary in every age, handwriting remains just as vital to the enduring saga of civilization as our next breath.

-Michael R. Sull-

Figure 9: an example of Document Image with VERTICAL ZONE=1. This image has a detection rate of 100% by both techniques.

The above described dataset were used to evaluate the old and the new algorithms. As evaluation metrics the detection rate and the recognition accuracy, provided by the evaluation software, where used. The CPU time stands for algorithms in Matlab, in a system with Core 2 Duo CPU 2.40 GHz.

	Old Technique [3-4]	**New Technique**
Mean Detection Rate (%)	72.783	96.76%
Mean Recognition Accuracy (%)	83.615	96.77%
Mean Computational Time (ms)	349.34	490.73
Total Detected Text Lines	1597	1771

Table 1: Comparative evaluation of the two techniques.

In table 1, the mean values per page of the above mentioned metrics are presented as well as the correct detected text lines per algorithm according to the evaluation software. It has to be noted that in no page were extracted more lines than the existed ones, while in some cases were extracted less.

	BESUS	***DUTH-ARLSA***	***ILSP-LWSeg***	***PARC***	***UoA-HT***	***RLSA***
Detection Rate (%)	*86,6%*	*73,9%*	*97,3%*	*92,2%*	*95,5%*	*44,3%*
Recognition Accuracy (%)	*79,7%*	*70,2%*	*97,0%*	*93,0%*	*95,4%*	*45,4%*
Total Detected Text Lines	*1904*	*1894*	*1773*	*1756*	*1770*	*1877*

Table 2: Comparative results of ICDAR 2007 handwriting text line segmentation contest [Gatos et al, 07].

In table 2, you can see the results of the ICDAR2007 handwriting text line segmentation contest. There is no CPU time, since there is no indication in the paper. In figure 9 an image example with VERTICAL LINES=1 and perfect result in both systems is shown, while in figure 10, the old result for the image of figure 2 is presented.

5 Conclusions

In this paper, a technique of text line detection and segmentation for handwritten document images was presented. The intention of this work was to create a robust system for line segmentation, able to deal with the existence of various skew angles and hill-and-dale writing. Our results demonstrated a higher performance with low aggravation of the computational cost. Moreover, the accuracy is comparative to other state-of-the-art approaches that deal with similar problems, as [Bukhari et al, 09] that presents 91.10% of accuracy and it can automatically detect text baselines with any orientation.

The division of the document page in vertical zones made the procedure more robust in the presence of various skews in the same text line. Moreover the incorporation of small tricks, like backward stepping in the case of obstacle or forward stepping after multiple stops on the same object improved significantly the accuracy of line segmentation.

In future work, we would like to test our split-page technique with more state-of-the-art methodologies for line segmentation. Moreover, we intent to experiment more with the LIMIT value and the MAIN BODY calculation, since we believe that those values could further improve our results.

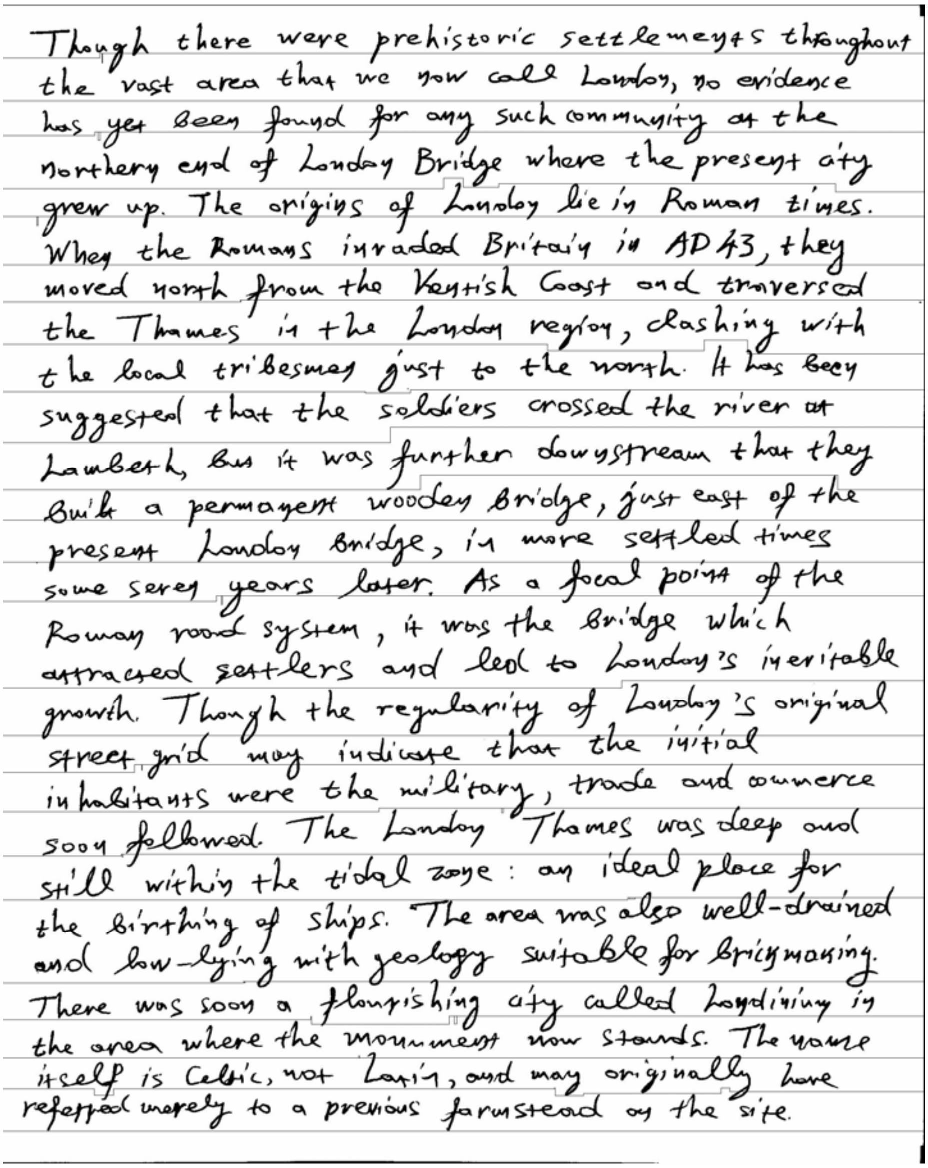

Though there were prehistoric settlements throughout
the vast area that we now call London, no evidence
has yet been found for any such community at the
northern end of London Bridge where the present city
grew up. The origins of London lie in Roman times.
When the Romans invaded Britain in AD 43, they
moved north from the Kentish Coast and traversed
the Thames in the London region, clashing with
the local tribesmen just to the north. It has been
suggested that the soldiers crossed the river at
Lambeth, but it was further downstream that they
built a permanent wooden bridge, just east of the
present London Bridge, in more settled times
some seven years later. As a focal point of the
Roman road system, it was the bridge which
attracted settlers and led to London's inevitable
growth. Though the regularity of London's original
street grid may indicate that the initial
inhabitants were the military, trade and commerce
soon followed. The London Thames was deep and
still within the tidal zone: an ideal place for
the berthing of ships. The area was also well-drained
and low-lying with geology suitable for brickmaking.
There was soon a flourishing city called Londinium in
the area where the monument now stands. The name
itself is Celtic, not Latin, and may originally have
referred merely to a previous farmstead on the site.

Figure 10: the example of Figure 1 by the old technique. This image had a detection rate of 70,3% by the old technique, while it reaches 100% by the new one.

References

[Arivazhagan et al, 07] M.Arivazhagan, H.Srinivasan, S.Srihari: A statistical approach to line Segmentation in handwritten documents, in: Proceedings of SPIE, 2007.

[Bar-Yosef et al, 09] I. Bar-Yosef, N. Hagbi, K. Kedem, I. Dinstein: Line Segmentation for Degraded Handwritten Historical Documents, Barcelona, Spain July 26-July 29 2009 10th International Conference on Document Analysis and Recognition, Barcelona, Spain, 2009.

[Boykov et al, 01] Y. Boykov, O. Veksler, and R. Zabih: Fast Approximate Energy Minimization via Graph Cuts, Proc. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23, no. 11, pp. 1222-1239, 2001.

[Bruzzone et al, 99] E. Bruzzone and M. C. Coffetti: An algorithm for extracting cursive text lines, in Proceedings of the Fifth International Conference on Document Analysis and Recognition, IEEE Computer Society, p. 749, 1999.

[Bukhari et al, 09] Bukhari, S.S., Shafait, F., Breuel, T.M.: Script-independent handwritten textlines segmentation using active contours, In: Proc. 10th Int. Conf. on Document Analysis and Recognition, Barcelona, Spain, 2009.

[dos Santos et al, 09] P. dos Santos, S. Clemente, I. Ren, and C. Calvalcanti: Text line segmentation based on morphology and histogram projection, International Conference on Document Analysis and Recognition, 2009.

[Du et al, 09] Xiaojun Du, Wumo Pan, Tien. D. Bui: Text Line Segmentation in Handwritten Documents Using Mumford-Shah Model, Pattern Recognition Volume 42, Issue 12, December Pages 3136-3145, 2009.

[Feldbach et al, 01] Feldbach, M., Tönnies, K.D.: Line Detection and Segmentation in Historical Church Registers, Sixth International Conference on Document Analysis and Recognition, pp. 743-747, September, 2001.

[Gatos et al, 07] B. Gatos, A. Antonacopoulos and N. Stamatopoulos: ICDAR2007 Handwriting Segmentation Contest, 9th International Conference on Document Analysis and Recognition (ICDAR'07), pp. 1284-1288, Curitiba, Brazil, September 2007.

[Gatos et al, 09] Basilis Gatos, Nikolaos Stamatopoulos, Georgios Louloudis: ICDAR 2009 Handwriting Segmentation Contest, Document Analysis and Recognition, International Conference on, pp. 1393-1397, 2009 10th International Conference on Document Analysis and Recognition, 2009.

[Kass et al, 88] M.Kass, A.Witkin, D.Terzopoulos: Snakes: Active Contour Models, International Journal of Computer Vision, 321-331, 1988.

[Kavallieratou et al, 02] Kavallieratou, N., Fakotakis, N., & Kokkinakis, G.: An unconstrained handwriting recognition system, International Journal on Document Analysis and Recognition, 4(4):226–242, 2002.

[Kavallieratou et al, 04] E.Kavallieratou and E. Stamatatos “Discrimination of Machine-Printed from Handwritten Text Using Simple Structural Characteristics”, In Proc. of the 17th International Conference on Pattern Recognition (ICPR 2004) Volume 1., pp. 437-440, 2004.

[Kumar et al, 06] K. S. Sesh Kumar, Anoop M. Namboodiri, and C. V. Jawahar: Learning segmentation of documents with complex scripts, In ICVGIP, pages 749-760, 2006.

[Kuzhinjedathu et al, 08] Kamal Kuzhinjedathu, Harish Srinivasan and Sargur Srihari: Robust line segmentation for handwritten documents, Proc. SPIE, Vol. 6815, 68150C, 2008.

[Li et al, 08] Y. Li, Y. Zheng, D. Doermann, S. Jaeger: Script-independent text line segmentation in freestyle handwritten document, IEEE Trans. Pattern Analysis and Machine Intelligence, 30(8):1313–1329, 2008.

[Louloudis et al, 06] G. Louloudis, B. Gatos, I. Pratikakis, K. Halatsis: A Block-Based Hough Transform Mapping for Text Line Detection in Handwritten Documents, Proceedings of the Tenth International Workshop on Frontiers in Handwriting Recognition, La Baule, Oct. 2006.

[Louloudis et al, 09] G. Louloudis, B. Gatos, I. Pratikakis, and C. Halatsis: Text Line and Word Segmentation of Handwritten Documents, Pattern Recognition (42), No. 12, 3169-3183, 2009.

[Pal et al, 03] U. Pal, S. Datta: Segmentation of Bangla unconstrained handwritten text, Proc. 7th Int. Conf. on Document Analysis and Recognition, Vol.2, pp. 1128-1132, 2003.

[Pu et al, 98] Pu Y., Shi Z.: A natural learning algorithm based on Hough transform for text lines extraction in handwritten documents, In Proceedings of the 6 Intl. Workshop on Frontiers in Handwriting Recognition, Taejon, Korea, pp. 637- 646, 1998.

[Razak et al, 08] Z. Razak, K. Zulkiflee, M. Yamani, I. Idris: Off-line Handwriting Text Line Segmentation: A Review, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.7, July 2008

[Shi et al, 2005] Z. Shi, S. Setlur, V. Govindaraju: Text extraction from gray scale historical document image using adaptive local connectivity map, Proc. 8th Int. Conf. on Document Analysis and Recognition, Vol.2, pp. 794-798, 2005.

[Stamatopoulos et al, 08] N. Stamatopoulos, B. Gatos, S.J. Perantonis: A method for combining complementary techniques for document image segmentation, International Conference on Frontiers in Handwriting Recognition (ICFHR'08), Montreal, Canada, pp. 235–240, August 2008.

[Tripathy et al, 04] N. Tripathy and U. Pal.: Handwriting Segmentation of Unconstrained Oriya Text, in International Workshop on Frontiers in Handwriting Recognition, pp. 306–311, 2004.

[Yin et al, 07] F. Yin, C.L. Liu: Handwritten text line extraction based on minimal spanning tree clustering, Proc. 5th Int. Conf. on Wavelet Analysis and Pattern Recognition, Vol.3, pp. 1123-1128, 2007.

[Zahour et al, 01] A. Zahour, B. Taconet, P. Mercy, S. Ramdane: Arabic handwritten text-line extraction, Proc 6th Int. Conf. on Document Analysis and Recognition, pp. 281-285, 2001.