2009 10th International Conference on Document Analysis and Recognition

Line segmentation for degraded handwritten historical documents

Itay Bar-Yosef, Nate Hagbi, Klara Kedem
Computer Science Department
Ben-Gurion university
Beer-Sheva, Israel
{itaybar,natios,klara}@cs.bgu.ac.il

Itshak Dinstein
Computer and Electrical Engineering department
Ben-Gurion university
Beer-Sheva, Israel
dinstein@ee.bgu.ac.il

Abstract

We propose a novel approach for text line segmentation based on adaptive local projection profiles. Our algorithm is suitable for degraded documents with text lines written in large skew. The main novelty of our approach is applying the local algorithm in an incremental manner that adapts to the skew of each text line as it progresses. The proposed approach achieves very accurate results on a set of degraded documents with lines written in different skew angles and curvatures.

1. Introduction

The rising interest in historical document image analysis has created many challenges for researchers. Degraded conditions of historical documents (e.g., bleed-through, ink stains, torn pages, etc) have motivated researches to develop binarization and enhancement algorithms suitable for these challenges [1, 2]. Algorithms for subsequent steps, such as recognition, skew detection, and page/line segmentation have mainly been developed for binarized data.

We present a robust text-line segmentation method for hand-written, degraded, historical documents, which is applied directly to gray scale images. Such documents often contain curved text lines and are very difficult to binarize. Most of the existing text-line segmentation methods are applicable only to binary images [3]. Very few of them are suitable for direct gray scale segmentation (e.g, [7],[8]). For a recent survey on text-line segmentation in historical documents, please refer to [3]. Our novel approach for text-line segmentation is based on oriented local projection profiles. We first propose a fast algorithm suitable for degraded documents containing lines with different moderate skew angles. Then we extend the algorithm to admit any skew angle by adapting to the skew of each line in the document as it progresses. The proposed approach achieves very accurate results on a set of degraded documents with different skew angles and with curved text-lines.

The rest of the paper is organized as follows: In Section 2 we give a detailed description of the local projection profile (*LPP*) based methods. Section 3 presents our *LPP* method and Section 3.1 discusses important parameters which influence the *LPP* methods. Section 3.2 describes our skew adaptation which is based on projective transformation, and in Section 4 we present experimental results. Conclusions and future work are outlined in Section 5.

2. Local projection profile analysis

A natural choice for line segmentation of gray scale images is the projection profile method [3]. By summing the pixel values along the horizontal direction of the document, gaps between text-lines can be found by searching for the projection's valleys. There are two main advantages for the projection profile approach in the context of historical document. First, it does not require binarization of the image, which makes it directly applicable to gray scale images. Second, it is very robust to noise and other degradations. The main weakness of this approach is its sensitivity to the skew of the lines in the document. Even a slight skew angle makes this approach impractical. Deskewing the document as a preprocessing step is one possible approach for tackling this problem. However, as often seen in historical documents (as well as in handwritten documents), text-lines can be curved in different skew angles. For that purpose, some authors ([4, 5, 6]) divided the document into non overlapping vertical stripes, and applied the projection to each stripe. For each vertical stripe, the valleys (minima) of the smoothed projection profile are detected, where each valley indicates a possible gap between two text lines. Valleys of two consecutive stripes are then linked together according to some predefined rules to form the final text line segmentation. We note that in both papers ([5, 6]) the segmentation results in a "stair-case" appearance as valleys are not connected directly (see Figure 1(d)). Another approach based on local projection profiles is proposed in [7]. Given a gray

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.191

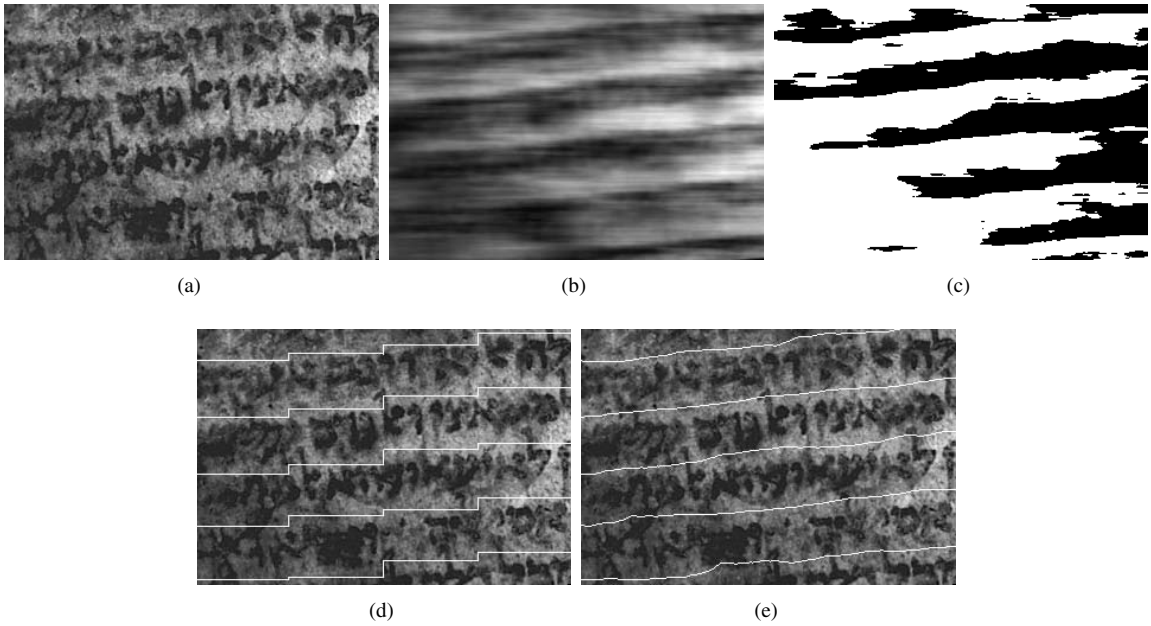


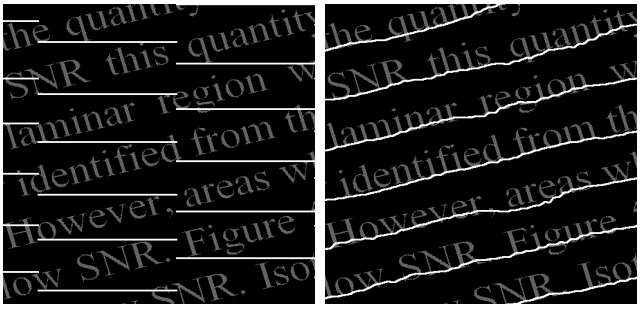
Figure 1. (a) Portion of gray scale image with slight skew. (b) The ALCM image [7]. (c) Binarization result of the ALCM image in (b). (d) Segmentation result of the vertical stripes approach [5]. (e) Segmentation result of our algorithm. In (d),(e), the segmentation results are superimposed in white.

scale image, the authors calculate for each pixel its projection profile along a predefined distance. This produces a smeared image (Figure 1(b)), called *ALCM* (Adaptive Local Connectivity Map), which is then binarized to yield the text line segmentation (Figure 1(c)). There are two main limitations to their method. First, it can handle only slight curvature of text lines. Second, the uniformity of the smeared image depends directly on the amount of degradation of the gray scale image. As a result, binarization of the smeared image is as difficult as binarizing the original gray scale image.

3. The local projection profile approach (*LPP*)

Our algorithm consists of two steps. First, we calculate the *LPP* for the leftmost vertical stripe (rightmost in case of Hebrew or Arabic) of the document. As the algorithm progresses we update the *LPP* by adding one column on the right (left) of the sliding stripe and subtracting one column on its left (right) side, getting the *LPP* for each pixel. We note that the this step is a fast implementation of the *Adaptive Local Connectivity Map* reported in [7].

The second step of our algorithm is to find the local minima (valleys) for each projection profile. Given a projection profile, we convolve it with a Gaussian derivative kernel and use the zero crossings of the derivative to detect intensity peaks. Since the result of the previous step is an image containing the *LPP* for each pixel, all we need to do is to convolve it with a 1D Gaussian derivative kernel and find its zero crossings. The result of this operation is a set of continuous segmentation border lines, each corresponding to a gap between two neighboring text lines (see our result in Figure 1(e)). Our approach has several important advantages. The results of our algorithm are smooth text line boundaries, which reflect the local skew of each text line individually. The results in [5, 6], on the other hand, have "stair-case" appearance which is fine when segmenting binary images, as these boundaries define the connected components to which each line belongs. However, for degraded gray scale images it is desirable to have an accurate description of the text lines, since connected component analysis cannot be applied. The second advantage is in the step of linking valleys of adjacent projection profiles. The one-pixel proximity of the profiles leads to natural linking of very close valleys (see Figure 2 (b)). In [5], the linking between two adjacent stripes is done by connecting each peak to the closet peak with that of its adjacent stripe. As can be seen in Figure 2(a), this rule does not necessarily yield the correct results.



(a) (b)

Figure 2. (a) Dividing the document into distinct stripes [5]. As seen, linking close peaks will cause erroneous results. (b) Results of our algorithm. The segmentation lines are not in the middle of the line's gap due to too large stripe width (in both algorithms we use the same width).

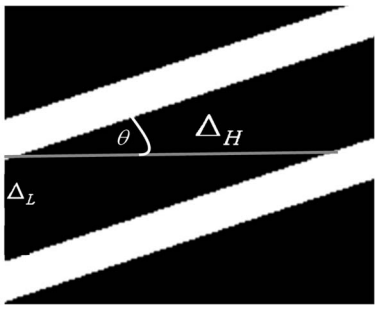


Figure 3. Illustration of two synthetic text lines. The parameters which influence on the choice of the stripe width are depicted: Δ_L, the gap between text lines. Δ_H, the line perpendicular to Δ_L. Θ, the line skew angle.

3.1. Analysis of the *LPP* approach

In [7, 5, 6] the width of the stripe is defined heuristically (e.g., average word length). A careful observation shows that there are several important parameters affecting the stripe's width, which highly influences the performance of the *LPP* approach. In the following, we describe and analyze these parameters. Figure 3 shows an example of two skewed lines. We denote by θ the skew angle, Δ_L the vertical gap between two text lines, and Δ_H the line perpendicular to Δ_L. A good choice of the width of the local stripe would result in correct detection of the valley between these two lines. As depicted in Figure 3, choosing Δ_H as the width will result in a single minimum of the projection profile. Lower widths will still result in minimum values in the correct range, but will be prone to mistakes of local minima, caused by, e.g., gaps between words. On the other hand, picking a stripe wider than Δ_H will not provide the correct minimum position of the projection profile.

The relation $\Delta_H = tan(90 - \theta) \cdot \Delta_L$ explains the high dependency of the stripe's width on the local skew angle and the gap between lines. As the skew angle θ increases, the width of the stripe should be reduced. Although the techniques reported in [5, 6, 7] were used on documents with small skews, the influence of the lines gap is still significant. The meaning of the above is that the local projection approaches are prone to errors when dealing with highly skewed images, or text lines with small gaps. Last important issue regarding the influence of the above parameters, is the effect of applying these methods directly to degraded gray scale images. Ink stains, faded characters, and bleed-through, all complicate the analysis of the minima of the projection profile. In this case the penalty for too narrow or too wide stripe is much higher. In addition, using wide stripes is of great importance when dealing with degraded documents. Gaps between lines can be missed in the presence of ink stains, and faded characters can lead to false minima of the projection profile. In order to overcome this limitation of the *LPP* and to allow wider stripes, we propose to apply the *LPP* method in an adaptive way according to the text-line orientation, instead of performing horizontal projections.

3.2. Local skew detection and oriented projection profile

Since in most documents text line orientation changes gradually, our idea is to apply the *LPP* method in an incremental way, i.e., stripe by stripe. During the progress of the algorithm we calculate the orientation of text-lines in the current stripe, and continue with the incremental algorithm in the new orientation. Two crucial steps are involved in the incremental algorithm: local skew estimation and progress in a given orientation.

Local skew estimation. Performing the algorithm on the current stripe results in a set of segmentation lines, each corresponding to a gap between the text-lines (Figure 1(e)). The average orientation of a segmentation line is then approximated by its start and end points in the current stripe, which yields the approximate local skew of the text line. In the next step, adjacent text lines are grouped together according to their local skew angles. Since the *LPP* method can handle small skew angles ($\approx 10\,^\circ$), text lines which differ by up to $5\,^\circ$ are grouped together. In the following, we describe how to apply the *LPP* method for one group of consecutive lines with approximately the same orientation.

If there are several groups of lines of different orientations, the procedure is applied independently to each of them.

Progress in a group of a given orientation. The basic idea of the algorithm is to obtain the average skew of the previous stripe and then to project the new stripe in this skew direction. Figure 4(a) shows an image containing three adjacent stripes, each delimited by a dashed line, where the first one on the left has already been processed. We first calculate the *control points* of the segmentation lines in the first stripe, $CP_1 = \{T_1, T_2, B_1, B_2\}$. The control points are the intersection points of the top and bottom segmentation lines with the vertical lines defining the beginning and the end of the stripe (Figure 4(a)). Notice that in order to apply the segmentation algorithm to the next stripe, its succeeding stripe is also needed. Therefore, we calculate the control points $CP_2 = \{T_2, T_3, B_2, B_3\}$ of the image containing the succeeding two stripes. CP_2 is calculated by extending the segmentation line through (T_1, T_2) until it reaches the end of the third stripe (T_3). Similarly, we get B_3 (Figure 4(b)).

In the next step we calculate and apply a spatial transformation T that maps CP_2 to the corners of a rectangular image of size N×M $\{(0,0),(0,M),(N,0),(N,M)\}$. Here $N = |T_3 - B_3|$, $M = |T_3 - T_2|$. Since the line through (T_2, T_3) is not necessarily parallel to the line through (B_2, B_3), we use projective transformation. This transformation does not have to be carried out explicitly. Instead, the locations of pixels projected according to the inverse transformation can be used to interpolate the values of neighboring pixels and sum them up. The result of the warping is a rectified rectangular image (Figure 4(c)) where the text lines are almost horizontal. At this stage we can apply the *LPP* method to the second stripe, as defined in Section 3. The last step is to apply the inverse transformation T^{-1} to the obtained result to get the final result shown in Figure 4(d) (superimposed in white). By advancing stripe by stripe and updating the respective control points, we make sure that the *LPP* algorithm is always applied to a relatively horizontal text lines. By this we are not limited to any skew angle, and can use much larger stripes.

The width of each stripe is defined according to the relation $\Delta_H = tan(90 - \theta) \cdot \Delta_L$ as detailed in Section 3.1. Since the *LPP* algorithm can handle effectively skew angles in the range +/-10 °, we define $\Delta_H = tan(80) \cdot \Delta_L$, where Δ_L is approximated from the projection profile of the first stripe.

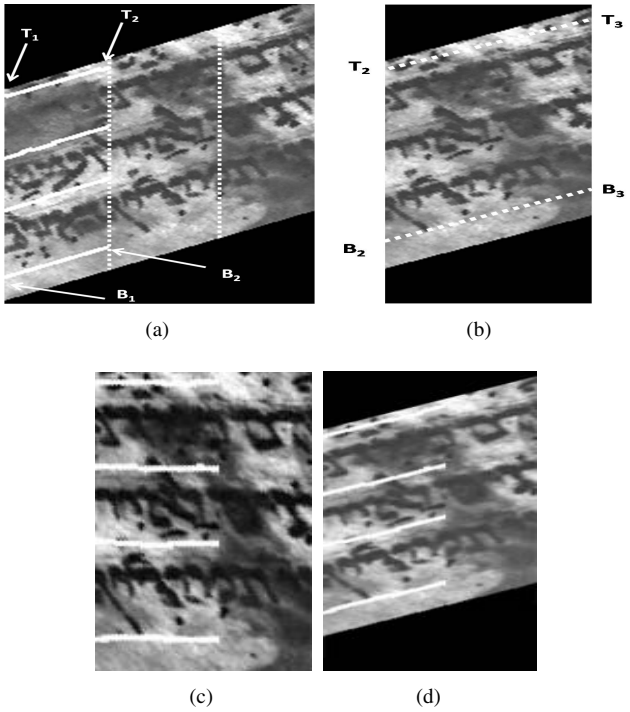


Figure 4. (a) Three stripes separated by dashed lines. (b) The next stripes to be processed. (c) Applying the *LPP* algorithm on the transformed image. (d) After applying the inverse transform we produce the final result.

4. Experimental results

We conducted two sets of experiments on 30 degraded historical documents supplied by Dr. Uri Erlich from the department of Jewish Thought, Ben-Gurion University, Israel. Each of the documents contains $20-30$ text lines, and 50% of the documents contain curved text lines in various angles (see Figure 5). In the first experiment we applied our algorithm to the entire set of documents. Since the experiments were conducted on gray scale images, the evaluation was done by visual inspection. In the second experiment we created another set by rotating manually each of the documents in 8 different angles in the range $[-45\,^\circ, 45\,^\circ]$ with $15\,^\circ$ difference. The purpose of the second experiment was to test the robustness of our algorithm to large skew angles. The results are very promising. In the first experiment 98% of the text lines were segmented correctly. The main cause for the 2% error was false detection of valleys in the local projection profile. The results of the second experiment were almost identical. In all the tested documents our algorithm correctly adapted to the skew angles, except for a few mistakes at the beginning of the document. The reason is mainly the fact that our algorithm is designed to adapt to the skew angle while it progresses. Once it had adapted to the skew of the document, its performance was the same as the documents in the first experiment.

(a)

(b)

Figure 5. (a) Sample document (b) Segmentation results superimposed in white.

5. Conclusion and future work

In this paper we proposed a novel approach for text line segmentation based on oriented local projection profile. We proposed a fast algorithm (*LPP*) suitable for degraded documents with moderate skew. The second part of our approach, and its main novelty, is applying the *LPP* algorithm in an incremental way, such that it adapt to the skew of the document as it progresses. The proposed approach achieves very accurate results on a set of degraded documents in different skew angles and with curved text lines. The information contained in the segmented text-lines can be very useful for warping of such documents. One possible approach for analyzing our algorithm is to perform warping based on our results then test its performance with respect to *word spotting*.

6. Acknowledgments

This work was partially supported by the Lynn and William Frankel Center for Computer Sciences, the Israel Ministry of Science and Technology, and by the Paul Ivanier Center for Robotics and Production Management at Ben-Gurion University, Israel. We also would like to thank Dr. Uri Erlich for supplying the images.

References

[1] I. Bar Yosef, "Input sensitive thresholding for ancient Hebrew manuscript", Pattern Recognition Letters, Vol 26, 2005 , pp. 1168-1173

[2] B. Gatos, I. Pratikakis, S. J. Perantonis, "Adaptive degraded document image binarization, Pattern Recognition, Vol 39, 2006, pp. 317-327

[3] L. Likforman-Sulem, A. Zahour, B. Taconet, "Text line segmentation of historical documents: a survey", International Journal of Document Analysis and Recognition. Vol 9, 2007, pp. 123-138.

[4] S. Tsuruoka, C. Hirano, T. Yoshikawa, and T. Shinogi, "Image-based Structure analysis for a Table of Contents and Conversion to XML Documents", Proc. of Document Layout Interpretation and its Applications (DLIA 2001), pp. 59-62, 2001.

[5] M. Arivazhagan, H. Srinivasan, and S. N. Srihari, "A statistical approach to handwritten line segmentation", Document Recognition and Retrieval XIV, Proceedings of SPIE, California, 2007, pp. 1-11

[6] A. Zahour, B. Taconet, P. Mercy, S. Ramdane, "Arabic hand-written text-line extraction", Proceedings of the Sixth International Conference on Document Analysis and Recognition, 2001, pp. 281-285

[7] S. Zhixin, S. Setlur, V. Govindaraju, "Text Extraction from Gray Scale Historical Document Images Using Adaptive Local Connectivity Map", Proceedings of the Eighth International Conference on Document Analysis and Recognition, 2005, pp. 794-798.

[8] D. J. Kennard, W. A. Barrett, "Separating Lines of Text in Free-Form Handwritten Historical Documents", Proceedings of the Second International Conference on Document Image Analysis for Libraries, 2006, pp.12-23