Curved segmentation path generation for unconstrained handwritten Chinese text lines

Nanxi Li, Xue Gao, Lianwen Jin
School of Electronics and Information Engineering
South China University of Technology
Guangzhou 510641, China
nanxi.li@mail.scut.edu.cn

***Abstract*—A new method of generating curved segmentation paths of handwritten Chinese text line is proposed. For overlapped characters, instead of background thinning, analysis of connected components in the overlap region of two neighboring characters is carried out, and a curved segmentation path is generated by connecting a series of points that separate the connected components belonging to different characters. As for touched characters, the candidate points are identified by using the histogram and skeleton information. After they are split at all the candidate points, curved segmentation paths are generated the same way as in overlapped characters. The experimental results on the unconstrained handwritten offline Chinese text lines show that our algorithm can improve the correct segmentation rate by 3.7% and valid segmentation rate 10.7%, indicating its effectiveness.**

I. INTRODUCTION

In handwritten Chinese text line recognition system, the segmentation path generation is an essential phase to produce a series of radicals or single characters for sequential processing. Then the generated segmentation paths are usually verified by structural or statistical rules, or even recognition information, to get final segmentation and recognition results. Thus, the segmentation path generation is very important for the performance of handwritten Chinese character recognition system. The more accurate the generated segmentation paths are, the better the recognition results.

Methods of segmentation path generation can be mainly categorized into three types: histogram projection analysis [1,2], stroke extraction and merging [3,4], and background thinning [5,6]. Histogram projection analysis always generates straight line, which is not appropriate for segmenting unconstrained handwritten Chinese characters. Stroke extraction and merging also cannot work well because the strokes of unconstrained handwritten Chinese characters are difficult to extract exactly. Background thinning can usually generate curved segmentation paths, which is appropriate for segmenting unconstrained handwritten Chinese characters. But its computational complexity is high, in that both thinning algorithm and stroke tracing algorithm are time consuming.

In this paper, a new method for generating curved segmentation paths is proposed for unconstrained handwritten Chinese text line recognition. Instead of using background thinning, analysis of connected components in the overlap region of two neighboring characters is carried out, and a curved segmentation path is generated by simply connecting a series of points that separate the connected components belonging to different characters. The proposed method shows good performance on unconstrained handwritten offline Chinese text lines.

II. SEGMENTATION ALGORITHM

The block diagram of the proposed method is shown in Fig. 1. The method mainly includes three steps: segmentation of naturally separated characters, segmentation of overlapped characters, and segmentation of touched characters.

A. *Segementation of naturally separated characters*

Histogram projection is used to segment naturally separated characters, i.e. characters that neither overlap in horizontal direction nor touch with each other. A straight line passing vertically through the mid of the zero region in histogram is a segmentation path. Fig. 2a shows the generated straight segmentation paths for naturally separated characters and Fig. 2b shows the histogram of the text line image.

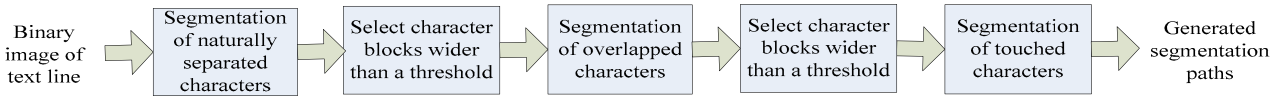
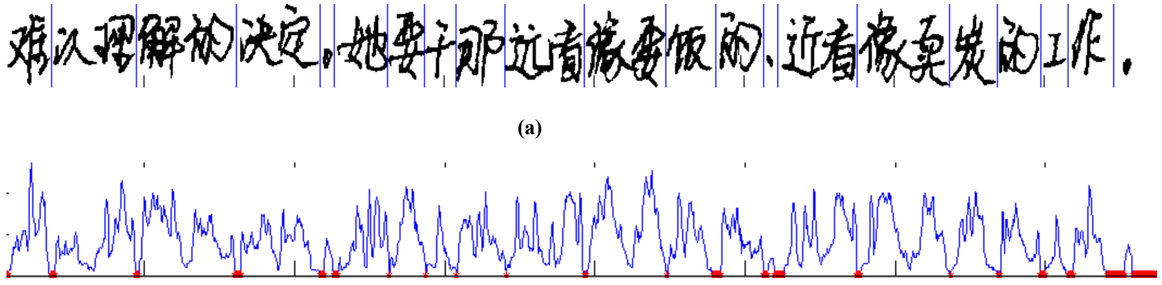

Binary image of text line
Segmentation of naturally separated characters
Select character blocks wider than a threshold
Segmentation of overlapped characters
Select character blocks wider than a threshold
Segmentation of touched characters
Generated segmentation paths

Figure 1. The block diagram of the proposed method.

978-1-4244-2342-2/08/$25.00 ©2008 IEEE.



(a)

(b)

Figure 2. (a) Straight segmentation paths for naturally separated characters, (b) histogram of the text line image.

The threshold for the width of character blocks, ThreshWid, is calculated as

$$\mathrm{ThreshWid}=\mathrm{M\_Width}-k_1*(\mathrm{Var\_Width})^{1/2} \tag{1}$$

where M_Width is the mean value of the width of character blocks, Var_Width is the variance of the width of character blocks, and k_1 is a constant. Here we assume that k_1 equals to 0.1. Character blocks wider than the threshold are selected, which may be overlapped characters or touched characters.

B. Segmentation of overlapped characters

A fast connected component analysis algorithm [7] is carried out on the selected character blocks. Then connected components are merged in the following two steps:

Step1. For two connected components in a character block, if the overlap rate in vertical direction (rateOLV defined by (2)) is larger than k_OLV and the overlap rate in horizontal direction (rateOLH defined by (3)) is larger than k_OLH, then merge them. Repeat step 1 until none of the connected components in the character block changes. Here we assume that both k_OLV and k_OLH are 0.4.

$$\mathrm{rateOLV}=(r2-r3)/\min(r2-r1,\ r4-r3) \tag{2}$$

$$\mathrm{rateOLH}=(c2-c3)/\min(c2-c1,\ c4-c3) \tag{3}$$

where r1 and r3 are the indices of the uppermost rows of the two connected components, r2 and r4 are indices of the lowermost rows of the two connected components, c1 and c3 are the indices of the leftmost columns of the two connected components, c2 and c4 are indices of the rightmost columns of the two connected components. And we assume that $r3\le r2$, $r4\ge r1$, $c3\le c2$, and $c4\ge c1$.

Step2. For two connected components in the character block, if rateOLH (defined by (3)) is larger than k_OLH2, then merge them. Repeat step 2 until none of the connected components in the character block changes. Here we assume that k_OLH2 equals to 0.4.

To eliminate the merging errors resulted from too long strokes in connected components, rateOLH is modified as

$$\mathrm{rateOLH}=(\mathrm{c\_g2}-\mathrm{c\_g3})/\min(\mathrm{c\_g2}-\mathrm{c\_g1},\ \mathrm{c\_g4}-\mathrm{c\_g3}) \tag{4}$$

and we define that

$$\mathrm{c\_g1}=\mathrm{c\_gravi}-\min(\mathrm{c\_gravi}-c1,\ c2-\mathrm{c\_gravi}) \tag{5}$$

$$\mathrm{c\_g2}=\mathrm{c\_gravi}+\min(\mathrm{c\_gravi}-c1,\ c2-\mathrm{c\_gravi}) \tag{6}$$

$$\mathrm{c\_g3}=\mathrm{c\_gravj}-\min(\mathrm{c\_gravj}-c3,\ c4-\mathrm{c\_gravj}) \tag{7}$$

$$\mathrm{c\_g4}=\mathrm{c\_gravj}+\min(\mathrm{c\_gravj}-c3,\ c4-\mathrm{c\_gravj}) \tag{8}$$

where c_gravi, c_gravj are the horizontal gravities of the two connected components. And we assume that $\mathrm{c\_g3}\le\mathrm{c\_g2}$, and $\mathrm{c\_g4}\ge\mathrm{c\_g1}$. Experiments show that better merging result will be achieved by using the modified rateOLH.

After merging, we focus on the overlap region of two connected components (cci and ccj, whose rateOLH is larger than zero) in the character block. A fast connected component analysis algorithm is carried out again in the overlap region, and each new connected component belongs to either cci or ccj. Suppose that a vertical straight line in the overlap region crosses a series of new connected components from top to down, which are recorded as {ccOL1, ccOL2,…, ccOLk, ccOLk+1,…, ccOLN}. If ccOLk (k=1, 2,…, N-1) belongs to cci (ccj) but ccOLk+1 belongs to ccj (cci), then a segmentation point is set between ccOLk and ccOLk+1 on the vertical straight line. Several points may be set on the line and each of them has an index. When the vertical straight line moves from left to right in the overlap region, points having the same indices are connected, forming several curved lines. The curved segmentation path is generated by connecting the curved lines along two vertical borders of the overlap region. The process is illustrated in Fig. 3. Fig. 3a shows the overlap region of cci and ccj, Fig. 3b shows a segmentation point on the vertical straight line, Fig. 3c shows a curved segmentation line formed by connecting the points having the same indices, and Fig. 3 d shows the curved segmentation path generated by connecting the curved lines along two vertical borders of

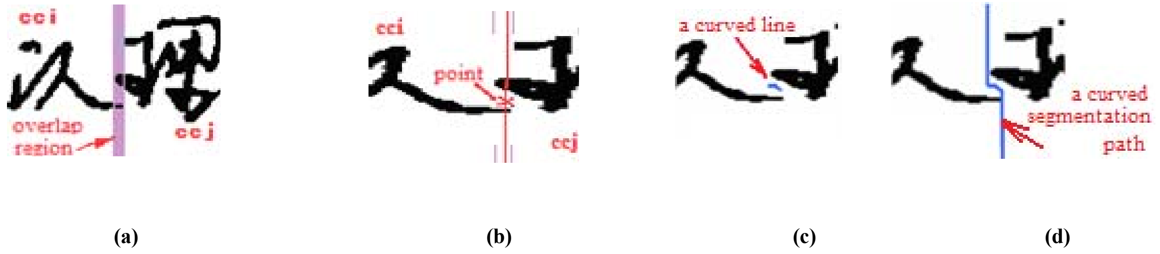


Figure 3. The process of generating a curved segmentation path for overlapped characters. (a) overlap region, (b) segmentation point, (c) curved line, (d) segmentation path.

the overlap region.

Some mis-segmentation can occur while setting points on vertical straight line in some cases, for examples as shown in Fig. 4, and modification must be made to find the correct segmentation paths. Fig. 4 shows four types of mis-segmentation and the corresponding modification. Among all the new connected components in an overlap region, suppose that those components whose leftmost and rightmost columns are at two vertical borders of the overlap region are identified from top to down as: {cc_OLcov1, cc_OLcov2, …, cc_OLcovp, cc_OLcovp+1,.., cc_OLcovM}. Fig. 4a shows that if cc_OLcov1 is not equal to ccOL1, then some points are missing in the region above cc_OLcov1. Modification is made by adding a virtual connected component in top row of the overlap region. If cc_OLcov1 belongs to ccj (cci), then let the added connected component belong to cci (ccj). It can be seen that a complete segmentation path is formed after modification. Fig. 4b shows that if cc_OLcovM is not equal to ccOLN, then some points are missing in the region below cc_OLcovM. Similar modification is made by adding a virtual connected component in bottom row of the overlap region. If cc_OLcovM belongs to ccj (cci), then let the added connected component belongs to cci (ccj). Fig. 4c shows that both cc_OLcovp (p=1, 2,…, M-1) and cc_OLcovp+1 belong to ccj (cci), and some points are missing in the region between them. Modification is made by adding a virtual connected component above cc_OLcovp+1, and let it belong to cci (ccj). Fig. 4d shows that in the region between cc_OLcovp and cc_OLcovp+1, there are two connected components overlapping with each other, one belongs to cci

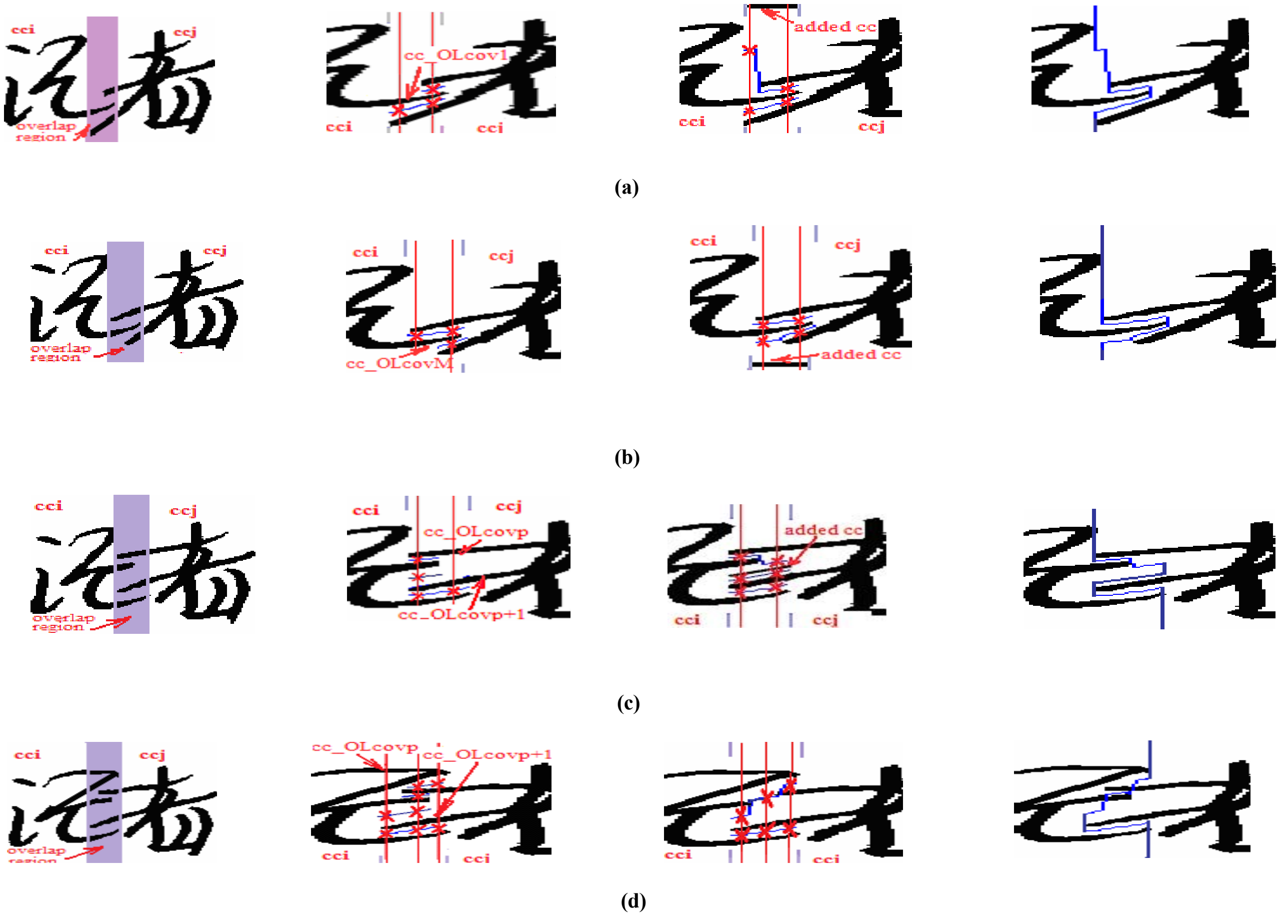


Figure 4. Four Types of mis-segmentation in setting points and the corresponding modification are shown in (a)~(d), respectively.

and the other belongs to ccj. Some points between the two connected components will be superfluous, and modification is made by virtually erasing the overlapped part of the two connected components.

The curved segmentation paths for overlapped characters after modifications are shown in Fig. 5. Clearly, the touched characters are still unsegmented. To deal with the touched characters, a threshold for the width of character blocks, ThreshWid1, is defined as

$$ThreshWid1=k_3*(M\_Width-k_2*(Var\_Width)^{1/2}) \qquad (9)$$

where k_2 and k3 are chosen experimentally, we choose 0.01 and 0.95 respectively. Character blocks wider than the threshold are selected as the candidate touched characters for further processing.

C. Segmentation of touched characters.

Segmentation of touched characters consists of the following five steps:

Step1. Find candidate valleys on the histogram of the character block.

The candidate valleys are the region that touched strokes may exist. In our algorithm for segmenting the touched characters, firstly record a peak sequence {P1, P2,…, Pi, Pi+1,…, PN}, which meets both (10) and (11):

$$Hist(Pi)>0.8*M\_peak \qquad (10)$$

$$Pi+1-Pi>0.2*ThreshWid1 \qquad (11)$$

where Hist(Pi) is the histogram value of Pi, M_peak is the mean value of all peaks in the histogram, and ThreshWid1 is defined by (9).

In interval [Pi, Pi+1] (i=1, 2,…, N-1), find the peaks whose values are larger than 0.5*M_peak, and insert them in the recorded peak sequence. Check recorded peaks one by one to find valleys between the peak and its next peak, and the valley P_{VMin} whose histogram value is minimal is recorded as a candidate valley, if it meets

$$Hist(P_{VMin})< 0.6*M\_hist \qquad (12)$$

where M_hist is the mean value of the histogram.

Step2. Find candidate strokes on foreground skeleton of the character block.

Suppose that a vertical straight line passing through a candidate valley (P_{ImpV}) crosses several strokes. Require a candidate stroke must meet either (13) or (14):

$$Stroke\_c2-Stroke\_c1>0.25*ThreshWid1 \qquad (13)$$

$$Hist(P_{ImpV}) \leq 2*Min\_ImpV \qquad (14)$$

where Stroke_c1 is the index of the leftmost column of the stroke, Stroke_c2 is the index of the rightmost column of the stroke, ThreshWid1 is defined by (9), Hist(P_{ImpV}) is the histogram value of $P_{ImpV,}$ and Min_ImpV is the minimal histogram value of all the candidate valleys.

Step3. Find candidate points on each candidate stroke.

All fork points and corner points on a candidate stroke are recorded as candidate points.

Step4. Split the touched characters at all the candidate points.

Step5. Generate curved segmentation paths in the split characters.

After splitting, the fast connected component analysis algorithm is carried out on the character block, followed by the two-step merging process. Here k_OLV and k_OLH are set to 0.4 and 0.45 respectively, and k_OLH2 0.8. The curved segmentation paths are generated the same way as in overlapped characters.

The process of generating curved segmentation paths for touched characters is shown in Fig. 6.

Add on the text line image the curved segmentation paths generated for touched characters, as is shown in Fig. 7.

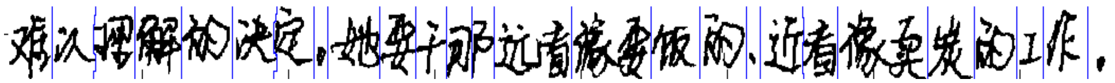


Figure 5. Curved segmentation paths for overlapped characters.

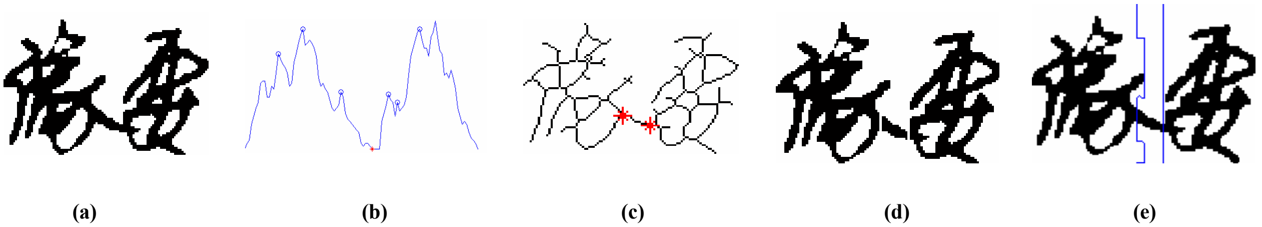


Figure 6. The process of generating curved segmentation paths for touched characters. (a) two touched characters, (b) histogram of the touched characters, (c) charactr foreground skeleton with identified candidate points, (d) touched characters split at the candidate points, (e) curved segmentation paths.

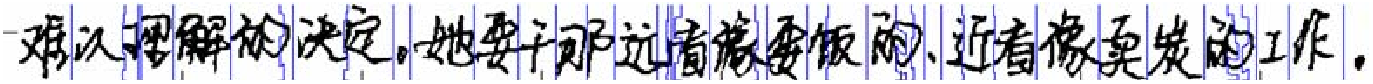


Figure 7. Curved segmentation paths for touched characters.

III. Experiment results

To test its effectiveness, the proposed method is applied to the HIT_MW database [8] where the text is naturally written by multiple writers under an unconstrained condition without predefined character boxes. And 50 text line images containing 1210 true segmentation paths are used in the experiment. For comparison, the method based on background thinning [5] is applied to the same text line images. We define the correct segmentation rate R_c and the valid segmentation rate R_v as follows:

$$R_c=N_c/N_t \tag{15}$$

$$R_v=N_c/N_a \tag{16}$$

where N_c is the number of correctly generated segmentation paths, N_a is the number of all generated segmentation paths, and N_t is the number of true segmentation paths in text lines. Experiment results are listed in Table I.

It can be seen from Table I, our method can improve R_c by 3.7% and R_v 10.7% comparing with the method based on background thinning. In comparison with our method without modifying rateOLH, R_c is improved by 0.9% but R_v is decreased by 2.1%. It indicates that the modified rateOLH is useful in generating correct segmentation paths, at the price of increasing the number of invalid segmentation paths.

TABLE I. Results of segmentation path generation Algorithm

	N_a	**N_c**	**R_c (%)**	**R_v (%)**
the proposed method	2435	1197	98.9	49.2
the proposed method (without modifying rateOLH)	2314	1186	98.0	51.3
the method based on background thinning	2989	1152	95.2	38.5

IV. Conclusions

In this paper, a new method of generating curved segmentation paths is proposed. A connected component analysis algorithm followed by a two-step merging process is presented to segment overlapped characters, and a curved segmentation path is generated in the overlap region of two horizontally overlapped connected components. Then, the candidate segmentation points for the touched characters are identified by using histogram and skeleton information, and the curved segmentation paths are generated the same way as in overlapped characters. The proposed method is applied to unconstrained handwritten offline Chinese text lines. Experiments show that significant improvements are achieved in both correctness and validity for generating segmentation paths.

References

[1] T. Yamaguchi, T. Yoshikawa, T. Shinogi, S. Tsuruoka, and M. Teramoto, “A segmentation method for touching Japanese handwritten characters based on connecting condition of lines,” Proc. 6th Int. Conf. Document Analysis and Recognition, pp. 837−841, 2001.

[2] Z. D. Feng, and Q. Huo, “Confidence guided progressive search and fast match techniques for high performance Chinese/English OCR,” Proc. 16th Int. Conf. Pattern Recognition, pp. 89−92, 2002.

[3] C. L. Liu, I. J. Kim, and J. H. Kim, “Model-based stroke extraction and matching for handwritten Chinese character recognition,” Pattern Recognition, vol. 34, pp. 2339−2352, 2001.

[4] Q. Fu, X. Q. Ding, T. Liu, Y. Jiang, and Z. Ren, “A novel segmentation and recognition algorithm for Chinese handwritten address character strings,” 18th Int. Conf. Patten Recognition, pp. 974−977, 2006.

[5] S. Zhao, Z. Chi, P. F. Shi, and H. Yan, “Two-stage segmentation of unconstrained handwritten Chinese characters,” Pattern Recognition, vol. 36, pp. 145−156, 2003.

[6] Z. Liang, and P. F. Shi, “A metasynthetic approach for segmenting handwritten Chinese character strings,” Pattern Recognition Letters, vol. 26, pp. 1498−1511, 2005.

[7] J. Bruce, T. Balch, and M. Veloso, “Fast and cheap color image segmentation for interactive robots,” Proc. Intelligent Robots and Systems 2000, 2000.

[8] T. H. Su, T. W. Zhang, and D. J. Guan., “Corpus-based HIT-MW database for offline recognition of general-purpose Chinese handwritten text,” Int. J. Document Analysis and Recognition, vol. 10, pp. 27−38, 2007.