A NOVEL APPROACH OF SEGMENTING TOUCHING
AND KERNED CHARACTERS
Yangxing LIU1  Yupin LUO1  Fei LIU1  Zhongqi QIU2

1Department of Automation, Tsinghua University, Beijing  100084,  P.R.China
{lyx,luo,liufei}@mail.au.tsinghua.edu.cn
2Research & Development Center, Brother Industries, LTD, Nagoya  467-8561,  Japan
zhongqi.qiu@brother.co.jp

Abstract Character segmentation is a critical step of OCR system. In this paper we discussed segmentation approaches of touching and kerned characters. A non-linear segmentation path-based algorithm for segmenting touching and kerned characters is put forward. First, touching and kerned characters are extracted and segregated with other characters by using character projections and recognition results. Then in order to find the non-linear segmentation path of touching and kerned characters, a heuristic method seeking minimal-penalty curved cut is used. Finally several regions belonging to the same character are merged. Experiment results show that the proposed method achieves high segmentation rate in documents written in both English and Japanese characters.

Keywords character segmentation; character recognition; touching and kerned characters; segmentation path

1. Introduction

Character segmentation is a technique that partitions images of lines or words into individual character. It is a critical step of OCR system because incorrectly segmented characters are not likely to be correctly recognized[1]. Segmenting well-formed and well-spaced printed texts is a relatively simple process. However, in documents with many touching and kerned characters, the segmentation accuracy is considerably lower.

Statistics have showed that touching and kerned characters are responsible for many errors in automatic reading of machine-printed text[1]. Touching and kerned characters are the characters that overlap with neighboring characters or that are components of connected multiple characters. Figure 1 shows examples of touching and kerned character images. Touching and kerned characters are likely to cause segmentation errors. If touching and kerned characters are incorrectly segmented, the character recognizer cannot recognize the characters and this error affect succeeding characters as well. So character segmentation influences the global performance of OCR system.

Segmentation of touching and kerned characters has been the most difficult problem in character segmentation. Many techniques were reported on this subject[2-6]. There are two key issues involved in this problem: determining which segments contain multiple characters and finding the correct segmentation path. Segmentation path is consisted of continuous points that isolate each single character. Fig.2 shows example of segmentation path. In Fig.2, sp and ep are start and end point of the segmentation path respectively.

In this paper, we present a novel means of segmenting touching and kerned characters, which are located in horizontal or vertical blocks. First, character projection is employed to isolate touching and kerned characters with other characters and perform the extraction of touching and kerned characters. Second the non-linear segmenting paths are constructed to segment touching and kerned characters by using heuristic method. Then some characters that are segmented into several regions in above process are correctly merged.

The rest of this paper is organized as follows. Section 2 explains existing segmentation algorithm. Section 3 describes the algorithm put forward in this paper. Experimental results are explained in Section 4. Conclusion remarks are given in the last section.

2. Existing segmentation algorithm

2.1 Segmentation based on character projections

Fig.1 Example of touching and kerned characters

Fig.2 Example of segmentation path
Character projection is widely used in character segmentation. The vertical projection $V(x)$ is the histogram obtained by counting the number of black bits in each vertical scan at position $x$. In a single line vertical projection, a peak occurs for each vertical stroke of the character. If the characters are well separated, $V(x)$ should have zero values between characters[7]. Fig.3 shows an example of vertical projection of a character line.

![Fig.3 Vertical projection of a character line](image)

In a single line vertical projection, a peak occurs for each vertical stroke of the character. If the characters are well separated, $V(x)$ should have zero values between characters. Fig.3 shows an example of vertical projection of a character line.

2.2 Segmentation by searching non-linear segmentation path

Wang and Jean [8] (1994) derived a non-linear cutting path from touching characters by identifying the shortest path. While searching the path for horizontal text-lines, they considered only three types of moves: a downward one and two diagonal ones. A large cost was assigned to a move that went towards a black pixel than through a white pixel. An additional penalty was applied to diagonal moves to encourage vertical cuts. The higher probability of a segmentation path implies a lower number of black pixels it passes through and the more straight it is. Fig.2 shows an example of segmentation path.

2.3 Recognition-based segmentation

Many people think that the existence of reliable features to distinguish boundaries in all fonts from interior regions is arguable, open-loop approaches, segmentation-to-recognition, render errors irrecoverable [9]. Therefore character segmentation should be closely coupled with character recognition. The recognizer will return a value $R$ after a region being recognized. The value $R$ indicates whether the region only includes a single character. If the value $R$ is greater than the threshold $T$, the region is to be regarded as a single character component; otherwise it is to be regarded as a multi-character component.

3. Character segmentation

Some well-formed and well-spaced characters exist in our machine-printed text. They can easily be segmented by their projection. But character projection is difficult to find the correct segmenting point of touching and kerned characters. If all characters are treated as touching and kerned characters, it is time consuming. But if touching and kerned characters are not considered in our algorithm, segmentation rate will be considerably lowered. Segmentation rate is defined as the percentage of characters that are segmented precisely.

In order to assure segmentation efficiency and high accuracy, we segment text using the following three steps:

1) Touching and kerned characters are extracted and separated with other characters by using character projection and recognition results.
2) A heuristic method is used to search the non-linear segmentation path of touching and kerned characters.
3) Some characters are correctly merged. Fig.4 shows the flow chart of our algorithm.

![Flow chart of our algorithm](image)
height of it, we can first extract some touching and kerned characters by calculating width of each region. If a region is wider than 1.5 times of its height, we regard it as a touching and kerned character.

- Then each region whose width is less than 1.5 times of their height is recognized. When the return value $R$ is less than the threshold, the region must also consist of touching and kerned characters.

From the examples shown in fig.5, we can identify that regions ((a), (b) and (c)) in Fig.6 consist of touching and kerned characters by using their width and recognition result. In region (b) and (c), touching and kerned characters are not correctly segmented. In region (a), character $\beta'$ was mistakenly segmented because of its touching with character $\beta$. So in the following segmentation procedure, not only touching and kerned characters must be correctly segmented but some regions belonging to the same character must be merged. The sequential merging procedure needs to be developed to merge such regions, which is presented in section 3.3.

---

### 3.2 Deriving non-linear segmenting paths by a heuristic method

In searching the correct segmenting path of touching and kerned characters, the number of possible segmentation path is very large. Moreover we must compute the cost of each path, which calls for large amount of time.

In order to reduce the computation burden and improve segmentation speed, a heuristic method is proposed in searching the correct segmentation path. The proposed heuristics evaluate each position and get the optimal one in searching procedure so many redundant paths are reduced from candidate path and raise segmentation efficiency. Therefore, an evaluation function must be established for determining the possible segmenting paths. When the criterion of the evaluation function is excessively severe, proper segmenting paths may be removed from candidate paths, which will induce segmentation errors. When the criterion is too low, on the other hand, some evidently mistaken path can not be removed from candidate paths, which will greatly increase computational cost. Based on outline properties of touching and kerned characters and experimental results, we find that the following two factors are effective in getting rid of many redundant paths.

- The number of black pixels that a segmentation path passes through
- The distance $d$ between the start and end point of a segmentation path

For touching and kerned characters, a correct segmentation path can not pass through many black pixels and the distance $d$ must not be too large. So $C_1(x,y)$ is defined to evaluate the number of black pixels that a segmentation path passes through; $C_2(x,y)$ is defined to evaluate the distance $d$.

Let $B_{num}(i)$ denote the number of black pixels that a segmentation path passes through and $W_s$ denotes the approximate stroke width of a text-line and $W_a$ denotes the average character width of a text-line. $(x_1,y_1)$ and $(x_2,y_2)$ represent starting and end point of a segmentation path(Fig.3).

\[ C_1(x,y) = W_s - B_{num}(i) \]

\[ C_2(x,y) = \gamma \times W_a - \left| x_2 - x_1 \right|, \]

where $\gamma$ is a constant.

Finally, our evaluation function $f(i)$ is defined as

\[ f(i) = C_1(x,y) \times C_2(x,y) \]

If $f(i) < 0$, which indicates that the segmentation path passes many black pixels or the distance $d$ is so large, then the path is removed; else the path is reserved.

Using the proposed heuristics, we can quickly find the correct segmentation path of touching and kerned characters in Fig.5. Fig.7 shows the segmentation path of touching and kerned characters in Fig.5.

---

### 3.3 Merging procedure

If the region is not a single character, then a merging procedure is invoked to merge small neighboring regions. The merging procedure uses the recognition result to combine the neighboring regions that belong to a single character. So some rules are presented in the following to merge some regions with neighboring regions.

First a region whose recognition result $v_1$ is less than the threshold $T$ is merged with its neighboring region whose recognition result is $v_2$ to form a new region and $v_1$ is the recognition result of the new region. In order to compare recognition result, $v_{temp}$ is calculated as

\[ v_{temp} = a \times v_1 + \beta \times v_2 \] (where $a + \beta = 1$)
where $\alpha$ and $\beta$ are constants and their value varies with the quality of the document. When $v_r$ is greater than $\max(v_{temp}, T)$, which indicate that recognition result of the new region are better than than of the two separating regions and the new region is a single character component, so the two regions are merged; otherwise two regions can not be merged.

Fig.8 is the final segmentation result of Fig.1. Touching and kerned characters in Fig.1 are correctly segmented.

4. Experimental results and analysis

The proposed system was tested on many printed documents written in both English and Japanese characters. When more than two characters form touching and kerned characters(Fig.9), we can use the proposed method repeatedly to isolate every single character which greatly increases self-adapting ability of our algorithm. Fig.10 shows segmentation result of character line in Fig.9.

Experimental results show that the proposed method can extract and segment touching and kerned characters with high efficiency and accuracy. The segmentation rate of the proposed method can reach 97%. So the proposed method is a valuable solution to segment touching and kerned characters.

5. Conclusions

This paper has proposed a method for segmenting touching and kerned characters from a document written in both English and Japanese characters. By this method, the touching and kerned characters are extracted from character line by analyzing character projection and recognition result. Then a heuristic method is used to determine candidate paths from all possible segmentation paths to remove redundant paths and reduce the computational cost. Finally some parts of a single character, which has been segmented into several parts by mistake in previous procedure, are merged. Through experiments with a great deal of printed documents, it has been shown that the proposed method is very effective for the segmentation of touching and kerned characters.

Reference