Segmentation of Touching Handwritten Japanese Characters Using Graph Theory Method

Misako Suwa
Fujitsu Laboratories LTD., 4-1-1 Kamikodanaka, Nakahara-ku, Kawasaki-shi 211-8588, Japan

Abstract
Projection analysis methods have been widely used to segment Japanese character strings. However, if adjacent characters have overhanging strokes or a touching point doesn’t correspond to the histogram minimum, the methods are prone to result in errors. In contrast, non-projection analysis methods being proposed for use on numerals or alphabet characters cannot be simply applied for Japanese characters because of the differences in the structure of the characters. Based on the oversegmenting strategy, a new pre-segmentation method is presented in this paper: touching patterns are represented as graphs and touching strokes are regarded as the elements of proper edge cutsets. By using the graph theoretical technique, the cutset matrix is calculated. Then, by applying pruning rules, potential touching strokes are determined and the patterns are oversegmented. Moreover, this algorithm was confirmed to be valid for touching patterns with overhanging strokes and doubly connected patterns in simulations.

Key words: Japanese handwritten character, touching character, oversegmentation, segmentation, edge cutset, graph

1. INTRODUCTION

Segmentation of unconstrained handwritten character strings is a difficult task because characters may touch one another. For segmentation of Japanese characters, vertical and horizontal histograms have commonly been used. However, if adjacent characters have overhanging strokes or a touching point doesn’t correspond to the histogram minimum, the methods are prone to result in errors. Non-projection analysis methods including profile analysis, vertex directed segmentation have been proposed for simply connected numerals [2],[3],[4],[5],[6],[7], but these techniques cannot be simply applied for Japanese characters because of the differences in the structure of the characters. Japanese is composed of four types of characters; Katakana, hiragana (both of which are Japanese phonetic syllabaries), kanji (imported Chinese characters), and numerals. Compared to numerals or letters in the Roman alphabet, every kanji generally has a highly compound structure; consisting of several components called radicals [8]. In addition, there is no space separating Japanese words. It is therefore almost impossible to segment such character strings without using recognition. An example of a kanji is shown in Fig. 1. It has left and right radicals (seven connected components).

Fig. 1. Kanji meaning “language”

According to Casey and Lecolinet’s classification [1], the “oversegmenting strategy” is thought to be a suitable approach to segmenting compound structured characters such as kanji or Chinese characters. In the pre-segmentation step of oversegmentation,
selected patterns are oversegmented; in other words, the pattern is cut at enough places to ensure the correct touching boundaries are contained. Along this strategy, Fujisawa et al. proposed “pattern-oriented segmentation” for touching numerals [3]. Not much research has been done, however, about the non-projection analysis of the segmentation of kanji or Chinese character strings. Gao et al. proposed split and merge method for Chinese numerals on bank checks [9]. They thinned patterns and extract horizontal stroke segments that terminate at either crossing point in a pattern. Among these segments, the ones that satisfy certain conditions are regarded as candidate ligatures. They target ligatures rather than touching strokes, so only simple connecting types are handled. Ikeda et al. suggest the use of the oversegmentation method with stroke width analysis for Japanese strings [10]. All points in which a vertical stroke joins a horizontal stroke are checked and if a point satisfies of some specific conditions, it is determined as being a potential touching points. Their method splits only simple types of touching strokes. Therefore histograms are used for the touching types that cannot be split by stroke width analysis.

The purpose of this research is to realize a segmentation algorithm of non-projection analysis for structured characters, such as kanji, to segment general touching types of characters. Furthermore, it must also be applicable to numerals and kana, which have simpler letterforms. Projection analysis is a histogram-based segmentation, in contrast with that, we define stroke-based segmentation as non-projection analysis in which the possibility of touching is judged for each stroke and cuts are done for potential touching locations.

The contents of this paper are as follows. The next section shows the results of an investigation of touching types for handwritten Japanese lateral writing. In section three, the technical terms of the graph theory used in this paper are explained. Moreover, a new algorithm to split a touching pattern using graph theory algorithms is proposed in the section. Section 4 has the results of simulations for evaluating the performance of the algorithm. The last section contains conclusions and remarks.

2. TOUCHING-TYPE INVESTIGATION

To clarify the discussion, the terms are defined as follows. If stroke $A$ of pattern $P$ and stroke $B$ of pattern $Q$ are touching, and an end point of $A$ is touching to one side of $B$, then $A$ is touching $B$. Furthermore $A$ is defined as a touching stroke. If an end point of $A$ is touching an end point of $B$, then $A$ and $B$ are touching each other. Consequently, $A$ and $B$ are defined as touching strokes. If one side of $A$ and one side of $B$ are touching, then $A$ and $B$ are overlapping. Moreover, if pattern $P$ and pattern $Q$ are touching or overlapping at a single place, $P$ and $Q$ are defined as being simply connected. If pattern $P$ and pattern $Q$ are touching or overlapping at two places and a loop is formed, then $P$ and $Q$ are defined as being doubly connected. An example of a doubly connected pattern is shown in Fig. 2

![Fig. 2. (a) Doubly connected pattern (b) Answer](image)

Before the segmentation algorithm was constructed, the touching patterns of Japanese characters in 200 images of handwritten addresses were examined and classified. The results are shown in Table 1. Typical touching patterns can be classified into the following types:
Table 1. Typical touching types of lateral Japanese writing

<table>
<thead>
<tr>
<th>Touching Type</th>
<th>Touching Stroke Relation</th>
<th>Rate</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simply Connected</td>
<td>S1</td>
<td>57.6%</td>
<td>平田</td>
</tr>
<tr>
<td></td>
<td>S2</td>
<td>26.6%</td>
<td>萩木</td>
</tr>
<tr>
<td>Doubly Connected</td>
<td>D</td>
<td>13.3%</td>
<td>北田</td>
</tr>
<tr>
<td>Others</td>
<td>O</td>
<td>2.5%</td>
<td>中町</td>
</tr>
</tbody>
</table>

| Combinations of S1 and S2 | ex |

S1: a horizontal or a slanted stroke is touching a vertical or a slanted stroke
S2: two strokes are touching each other
S3: two strokes are crossing
S4: two strokes are overlapping

Type S1 of the simply connected pattern is the most frequently observed pattern, with a ratio of about 58%. The second is type S2, and its ratio is about 27%. The appearance frequency of the doubly connected type, D, is 13%. Although it occurs less frequently than S1 and S2, D is not negligible. Of course, if characters are more sparsely filled in on a form, the ratio for the doubly connected case would be lower. To segment general Japanese character strings, however, not only simply connected patterns but also doubly connected patterns must be handled.

3. SPLITTING A TOUCHING PATTERN USING GRAPH THEORY ALGORITHMS

3.1. Motivation

In segmenting doubly connected patterns, one of the difficulties is how to determine potential touching points. If contour analysis or stroke width analysis is simply applied for touching patterns, a lot of stroke-crossing points are obtained. An example is shown in Fig. 3 in gray. How does the algorithm determine a non-redundant cut, which divide the pattern into 2 connected components? The simplest answer is to use proper edge cutset, a concept in the graph theory. If we represent a touching pattern as a graph, touching strokes correspond to proper edge cutsets of the graph. The edge cutsets of the graph can be calculated using graph theory algorithms.
3.2. Definition of terms in graph

This section explains the definitions of the graph theory terms necessary for this study. The notations are respond to those in Ref. [12].

Graph \( G(V,E) \) is composed of a finite nonempty set of vertices, \( V(G) \), and a set of edges \( E(G) \). Let \( u \) and \( v \) be vertices in a graph \( G \). \( G \) is connected if \( G \) contains path \( uv \) for any pair of \( u \) and \( v \). Degree \( d(v) \) of vertex \( v \) is the number of edges incident with \( v \). End point, turning point, T-joint point and crossing point correspond to vertices of degrees 1, 2, 3 and 4, respectively. Let \( E_i \) be a subset of \( E \) and \( G(V,E-E_i) \) is disconnected, then \( E_i \) is an edge cutset of \( G \). A minimum cutset is called a proper cutset. A proper cutset divides \( G \) into two connected components. An example of a cutset and a proper cutset are shown in Fig. 5.

The necessary matrices associated with graphs are as follows.

**Incidence Matrix** \( B = [b_{ij}] \) of \( G(V,E) \):

Each row of \( B \) corresponds to a vertex of \( G \), and each column of \( B \) corresponds to a vertex of \( G \) and

\[
b_{ij} = \begin{cases} 1 & \text{if } e_j \text{ is incident with } v_i \\ 0 & \text{otherwise} \end{cases}
\]

where \( e_j \in E, v_i \in V \)

**Adjacency Matrix** \( S = [s_{ij}] \) of \( G(V,E) \):

The rows and columns correspond to the vertices of \( G \) and

\[
s_{ij} = \begin{cases} 1 & \text{if } v_i v_j \in E \\ 0 & \text{otherwise} \end{cases}
\]

where \( v_i, v_j \in V \)
Circuit Matrix $C = [c_{ij}]$ of $G(V,E)$:

Each row of $C$ corresponds to a circuit in $G$, and each column corresponds to an edge in $G$ and

$$c_{ij} = \begin{cases} 
1 & \text{if } e_j \text{ is included in circuit } C_i \\
0 & \text{otherwise}
\end{cases}$$

where $e_j \in E$

Edge Cutset Matrix $K = [k_{ij}]$ of $G(V,E)$:

Each row of $K$ corresponds to a proper cutset of $G$, and each column corresponds to an edge of $G$.

$$k_{ij} = \begin{cases} 
1 & \text{if } e_j \text{ is included in proper cutset } K_i \\
0 & \text{otherwise}
\end{cases}$$

where $e_j \in E$

If each proper cutset of $K$ is linear independent mod 2, then $K$ is called a cutset basis matrix.

Once $B$ is calculated, circuit basis matrix $C$ can be calculated from $B$. Then, cutset basis matrix $K$ can be calculated from $C$. Detailed algorithms are also explained in Ref. [12].

3.3. Graph-representation of a touching pattern

This section describes the way to represent a touching pattern as a connected graph is described. At first, a candidate touching pattern (single connected component) is thinned using Hilditch’s algorithm [15]. Next, the 8-neighborhood of each point of the thinned pattern is checked and vertices of degrees 1,3 and 4 are extracted [16]. Then, every edge between 2 vertices is pursued. If an edge is a bend, a degree-2 vertex is extracted using the corner detection method [13], and the edge is divided into two parts. The coordinates of all vertices and the points of all edges are recorded in a data file. After that, the incident matrix and the adjacency matrix are obtained from the information about the vertices and the edges. Finally, the edge cutset matrix is calculated from the incident matrix. An example of a touching pattern and its graph-representation is shown in Fig. 7(a) and Fig. 7(b), respectively. The graph consists of 25 edges and 21 vertices. Figure 8 shows the cutset basis matrix of the graph. Each column corresponds to a cutset base of the graph. The targets are simply and doubly connected patterns, so all proper cutsets whose number of elements are equal to or less than 2 are calculated from the cutset basis. When proper cutsets with 1 or 2 elements, $K'_0, K'_1, \ldots, K'_{m-1}$ are obtained, all true touching strokes are within one of the sets

![Fig. 6. Addition of degree-2 vertex](image-url)
3.4. Decision on potential touching strokes

To obtain the potential touching strokes, pruning rules are applied for the proper cutsets, $K'_0$, $K'_1$, ..., $K'_{m-1}$. The rules concern the length of an edge, the slant of an edge, the types of vertices that join edges and the sizes and positions of the connected components after individual edges that belong to one cutset are removed. In the last rule, instead of cutting thinned image, virtual cut is done by replacing elements of the adjacency matrix $1$ with $0$, that correspond to the elements of the cutset. The adjacency matrix $S'$ after removal of edge(s) that belong to a cutset can be block-diagonalized as

$$S' = \begin{bmatrix} H_{11} & 0 \\ 0 & H_{12} \end{bmatrix}$$

Each block corresponds to the vertex group of a connected component. The sizes and position of each connected component after removal of the edge(s) can be calculated by the information about the vertices belonging to the blocks. Left-and-right separated is an example of the pruning rules for lateral writing.

After applying the pruning rules, the remaining cutsets correspond to potential touching strokes. The start and the end points of the strokes can be regarded as potential touching points, and the touching pattern is over-split at both sides. In Fig. 10, the
potential touching strokes are indicated in gray. In this example, two sets of candidate potential strokes are obtained. The over-split pattern is shown in Fig. 11.

\[ S' = \begin{bmatrix} H_{11} & 0 \\ 0 & H_{22} \end{bmatrix} \]

Fig. 9. Calculation of sizes and positions of connected components after edges are removed. The sizes and the positions of each connected component can be calculated using the diagonalized adjacency matrix \( S' \) and the information about the vertices.

Fig. 10. Result of decision about potential touching strokes

Fig. 11. Over-split of touching pattern. Bounding boxes are added.

4. SEGMENTATION OF JAPANESE CHARACTER STRINGS

Oversegmentation is thought to be a suitable approach to handle structured characters such as kanji or Chinese characters and it generally consists of the following four steps, (1) decision about touching patterns (2) pre-segmentation (3) generation of multiple hypothesis (4) decision about an optimal segmentation path. The splitting algorithm mentioned above is used for the pre-segmentation step. The segmentation algorithm is constructed as follows.

a. Decision about touching patterns

First, the connected components in a character string image is labeled and their bounding box information is extracted. Each connected component is tested and only the components that meet certain conditions e.g., size, aspect ratio, are determined as being potential touching patterns.

b. Pre-segmentation

The potential touching patterns are over-split into sub-patterns using the splitting algorithm.

c. Generation of multiple hypotheses

Possible combinations of sub-patterns are checked and if they satisfy certain conditions e.g., size, aspect ratio, they are
merged and recognized. Subsequently, multiple segmentation hypotheses are generated. At this time, the candidate lattice method [11] is used. A candidate lattice is a directed graph made from multiple segmentation hypotheses, where candidate patterns and their recognition results (category codes and their weights) are attached between the nodes. An example of a candidate character lattice is shown in Fig. 12.

d. Decision of an optimal segmentation path

To decide an optimal segmentation path, DP matching is used at this time.

$$\text{Fig. 12. Candidate character lattice}$$

5. EXPERIMENTAL RESULTS

To evaluate the effectiveness of the algorithm a segmentation experiment was executed. 100 touching kanji-patterns from 100 people were used. The resolution of an image was 300 dpi. A pattern contained two or three characters. Directional code feature [14] was used for character recognition. We define

$$\begin{align*}
C &= \text{number of correct splitting points/number of correct touching points} \\
M &= \text{number of correct touching points/number of potential touching points} \\
R &= \text{recognition rate of characters}
\end{align*}$$

Segmentation and recognition results are shown in Table 2. About 93.8% of touching points were correctly split. The ratio of the case in which a split was constructed but a tail was incorrectly assigned to a character was 2.7%. The remainder failed.

<table>
<thead>
<tr>
<th>C</th>
<th>M</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>93.8% (106/113)</td>
<td>0.31 (113/365)</td>
<td>68.0%</td>
</tr>
</tbody>
</table>
6. CONCLUSIONS AND REMARKS

A new algorithm for the stroke-based segmentation of handwritten Japanese character strings is proposed in this paper. This algorithm is confirmed to be valid for various types of touching characters including doubly connected patterns in the experiment. At this time, we did not use contextual knowledge to decide an optimal segmentation path. If we use it, the recognition accuracy will be improved.

Acknowledgements

the author would like to thank Yutaka Katsuyama, Seiji Takahashi and Kanta Ohshige for their contributions to the constructions of this segmentation method.

References


