On-line handwritten Kanji character recognition using hypothesis generation in the space of hierarchical knowledge

Kenji Ohmori
College of Engineering
Hosei University
Koganei-shi, Tokyo 184, Japan

Yoshihito Haruki
Graduate School of Engineering
Hosei University
Koganei-shi, Tokyo 184, Japan

Abstract
For on-line handwritten Kanji recognition, a new approach which cyclically generates a more concrete hypothesis from the current hypothesis by using hierarchically represented knowledge and has achieved high recognition rate in reduced processing time, is described.

1 Introduction

The study of on-line handwritten Kanji character recognition[1] has recently been carried out extensively. However, a recognition algorithm applicable to practical usage has not yet been found. There are two dominant styles of writing Kanji characters, block and cursive. The block style is written carefully and slowly, with careful attention to proper stroke number and order. The cursive style is written faster with fewer strokes by connecting some of the block style strokes. Some papers study on the block style character recognition and a paper[2] does on the cursive character recognition.

In the cursive style, there are many variations which expand from a style close to the block style, to a style from which the original block style cannot be discerned. Up to now, the studies on the cursive style[2] had been carried out mainly for the style very close to the block style. In this case, the search space is limited to a reasonable size. However, the style used in ordinary writing is more distorted than the studied style. The distortion causes the search space to be enormously large. The main problem for the ordinary writing style becomes how to reduce the search space to a reasonable size without missing the correct results.

2 Hierarchical knowledge

The scheme of hierarchical knowledge representation employed in this paper is shown in Figure 1. The knowledge of characteristic strokes, which change hardly their block style shape when they are written in the cursive style, is presented in the high-level. Figure 1(a) represents an example of a characteristic stroke whose shape is a long straight vertical stroke which lies on the left side. This stroke appears in Kanji characters with subpatterns "木", "末", "木", and "禾".

The knowledge of subpatterns in Kanji characters is represented in the middle-level. Many Kanji characters are composed of two simpler Kanji characters. These simple characters are treated in the middle-level as subpatterns. Patterns "木", "末", "丁", and "才" are examples of subpatterns.

The knowledge of each Kanji character is represented in the low-level. That is, reference patterns written in the block style with proper stroke number and order are stored in this level.

Fig. 1. Hierarchical knowledge representation
On-line handwritten Kanji recognition is carried out by generating hypotheses using hierarchical knowledge. The algorithm is described below. 1) Find one or more characteristic strokes using the high-level knowledge from the input pattern and generate one or more hypotheses showing which subpattern is included in the input pattern. 2) Verify each of these hypotheses using the middle-level knowledge. From each of the hypotheses which have been accepted, generate new hypotheses using the middle-level knowledge. A hypothesis shows which Kanji character is similar to the input pattern. 3) Verify each of these hypotheses using the low-level knowledge and output an accepted hypothesis as a recognition result.

3 Search space limitation

By examining cursive characters carefully, it is found that some strokes of Kanji characters do not change very much from the block style shape even when they are written in the cursive style. For example, the Kanji character "フ" has a long straight vertical stroke on the left side. Even when this stroke is written as part of a connected stroke, its characteristic remains unchanged in the connected stroke as shown in Figure 2.

In this paper, the properties of a characteristic stroke are represented in the form of fuzzy rules. Characteristic strokes, each of which is close to part of an input pattern stroke, are found by fuzzy inference, where each input pattern stroke is investigated by using the fuzzy rules whether it includes a characteristic stroke.

4 Hypothesis generation

When the pattern in Figure 2 is given as an input pattern, the characteristic stroke of a long straight vertical stroke on the left side will be found by the fuzzy inference. From the hierarchical knowledge, subpatterns "フ", "扌", "卄", and "卄" are obtained. From these subpatterns, new hypotheses are generated such that subpattern "扌" is a component of the input pattern. The input stroke is divided at the point where subpattern "扌" is connected with another pattern.

Each hypothesis is verified by using the middle-level knowledge. Assume the case where we verify the hypothesis that subpattern "扌" is a component of the input pattern. As this subpattern lies on the left side, the left part of the input pattern is selected as the input subpattern.

Subpattern "扌" consists of three strokes, but the input subpattern has only one stroke. It is assumed that the input subpattern was written by connecting these strokes. For comparison of these subpatterns, a new subpattern "扌" with only one stroke is generated by connecting neighboring strokes of the original subpattern "扌". From the new subpattern, a fuzzy rule is generated. Through the use of this rule, the hypothesis that subpattern "扌" is a component of the input pattern is verified.

When the hypothesis has been accepted, reference patterns are obtained from the hierarchical knowledge. A more concrete hypothesis is generated such that the input pattern is the same Kanji character as the reference pattern "扌". The concrete hypothesis is verified through the use of low-level knowledge.

<table>
<thead>
<tr>
<th># of strokes</th>
<th>success rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>96.6 80.0 95.0 71.6</td>
</tr>
<tr>
<td>9</td>
<td>97.9 84.4 95.4 75.1</td>
</tr>
<tr>
<td>14</td>
<td>96.4 88.6 95.2 81.5</td>
</tr>
<tr>
<td>aver.</td>
<td>97.0 84.3 95.2 76.1</td>
</tr>
</tbody>
</table>

Table 1. Recognition results

5 Evaluations

In Table 1, the evaluation results are shown in comparison with an approach[2] without hypothesis generation. As seen in this table, the approach described in this paper obtained higher recognition results. The processing time has also been decreased to one-third.

References
