Performance Improvement Strategies on Template Matching for Large Set Character Recognition

Soo Hyung Kim
Department of Computer Science, Chonnam National University
300 Yongbong-Dong, Buk-Gu, Kwangju, 500-757 Korea
E-mail: shkim2@orion.chonnam.ac.kr

Abstract

This paper proposes an algorithm for offline recognition of handwritten characters, especially effective for large set characters such as Korean and Chinese characters. The algorithm is based on a template matching method which is easy to implement but suffers from low recognition performance. Two strategies have been developed to improve the performance of the template matching: one is multi-stage pre-classification and the other is pairwise reordering. Effectiveness of the algorithm has been proven by an experiment with a handwritten Korean character database named PE92, where 86.0% of recognition accuracy and 15 characters per second of processing speed have been obtained.

1 Introduction

Offline recognition of handwritten characters is a crucial issue in oriental countries using large set characters. It should resolve simultaneously two contradictory problems of absorbing variations of the same characters among different writing styles and resolving graphical similarity of different characters.

This paper proposes an algorithm for offline recognition of handwritten characters, especially effective for large set characters such as Korean and Chinese characters. The algorithm is based on a template matching method which is easy to implement due to its algorithmic simplicity and higher degree of flexibility to the change of recognition target classes. Higher degree of flexibility is very useful in forms processing applications where the recognition target classes are different for individual fields. In spite of these advantages, the template matching is not used in practical applications because of low recognition performance[1, 4, 6, 8].

To improve the recognition performance of the template matching, we have developed two kinds of performance improvement strategies, i.e., multi-stage pre-classification and pairwise reordering. The multi-stage pre-classification is to reduce the processing time of the template matching by cutting off a number of recognition target classes[7]. It is desirable to cut off as many classes as possible with little or no degradation of recognition accuracy. The pairwise reordering is to enhance the recognition accuracy by performing a fine detail classification on the recognition candidates generated from the template matching [5].

The resulting algorithm consists of three processing stages of multi-stage pre-classification, template matching and pairwise reordering. An experiment with a handwritten Korean character database named PE92 has been performed to prove the effectiveness of the proposed algorithm.

2 Proposed Algorithm

There are three independent modules in the proposed algorithm as shown in Figure 1. The multi-stage pre-classification module reduces the
number of recognition target classes which will be considered in the template matching by a series linear classifications with scalar features. The template matching module generates an ordered set of recognition candidates for the input character pattern by comparing the distance from the model vectors. The pairwise reordering module rearranges the recognition candidates by a series of pairwise discriminations.

**Figure 1. Algorithm structure**

### 2.1 Multi-Stage Pre-classification

The multi-stage pre-classification (MP) is to reduce the number of recognition target classes or templates which will be considered in the template matching, and hence to speed up the template matching process. The MP algorithm consists of six successive stages of linear classifications, each of which cuts off some classes by use of a scalar feature. A scalar feature $S_i$ ($1 \leq i \leq 6$) is computed from the input image and then compared to the distributions of scalar values for individual classes. Those classes whose distributions do not include the scalar value $S_i$ are removed from the set of recognition target classes.

![Linear classification by a scalar feature](image)

**Figure 2. Linear classification by a scalar feature**

An example situation of the linear classification is presented in Figure 2. Among five class distributions, just two distributions include the scalar value $S_i$, so the two recognition target classes are left after the linear classification. By applying the six-stage pre-classification, more than 60% of recognition target classes are excluded in general with little degradation of recognition accuracy of the template matching.

### 2.2 Template Matching

Template matching (TM) is a statistical pattern classification technique where the input character pattern is transformed to a vector in a multi-dimensional feature space and then compared to a set of model vectors or templates for individual recognition target classes. We adopt a $7 \times 9 \times 4$ (252 dimension) structured segment features [8] for the transformation of input patterns into the feature space - see Figure 3. The model vectors have been trained by the use of an LVQ (learning vector quantization) technique [3] with given learning samples. As a recognition result for the input character pattern, the TM module generates best five candidates which are ordered according to Euclidean distances from the input vector to the individual model vectors.

![Feature extraction for template matching](image)

**Figure 3. Feature extraction for template matching**

### 2.3 Pairwise Reordering

Given an ordered set of recognition candidates computed from the template matching, the pairwise reordering (PR) module rearranges them by the use of pairwise classifiers. The reordering starts with invoking a pairwise classifier for the pair of two candidates of higher distance (or smaller similarity),
and then the winner of the classification and the next candidate are prepared for another reordering, and so on.

The number of possible pairs increases exponentially as the recognition target classes increases, so we selected those pairs which appear most frequently during the reordering. We have observed that just 1.4% of most frequent pairs cover more than 40% of the entire pairs. Classifiers for these pairs are trained by using a three-layered feed-forward neural network and given learning samples. We have also observed that about 14% of most frequent pairs cover more than 90% of the entire pairs, so we handle these pairs by the TM module which uses a 32D sub-vectors of the 252D model vectors used in the TM process. The sub-vector is prepared based on the Fisher measure. The reordering is not performed for the pair whose classifier does not exist.

3 Experimental Results

The algorithm has been implemented in a C programming language on a Pentium 120 MHz PC. Currently, the recognition target classes are restricted to a set of 574 Korean characters which are used most frequently in forms processing applications.

![Example patterns for ‘한’ in PE92](image1)

Figure 4. Example patterns for ‘한’ in PE92

Recognition performance of the algorithm has been evaluated with a handwritten Korean character database named PE92[2]. Some example patterns in PE92 are shown in Figure 4. We have partitioned the 100 sets in the database into 67 and 33 sets for learning and testing, respectively. Samples which are labeled incorrectly (about 4% of the database) have been removed manually. The recognition accuracy and speed are summarized in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>First candidate hit ratio (%)</th>
<th>recognition speed (char/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TM</td>
<td>76.8</td>
<td>13.7</td>
</tr>
<tr>
<td>MP+TM</td>
<td>77.6</td>
<td>18.7</td>
</tr>
<tr>
<td>MP+TM+PR</td>
<td>86.0</td>
<td>15.0</td>
</tr>
</tbody>
</table>

As can be seen from the table, both MP and PR modules improve the recognition performance of the TM module. The first candidate hit ratio has been improved by about 38.3% (from 76.8% to 86.0%), and the recognition time has been reduced by about 8.7% (from 13.7 char/sec to 15.0 char/sec).

4 Conclusion

Two kinds of performance improvement strategies are proposed to be integrated with the template matching method. One of major contributions of the proposed integration is to make it effective to utilize the template matching for the recognition of large set handwritten characters.

Effectiveness of the algorithm has been proven by a recognition of handwritten Korean characters. It can also be verified with the recognition of Chinese characters. Using the proposed approach, the author is currently working on the recognition of mixed Korean and alphanumeric handwritten characters.

Acknowledgment

This work was done when the author was a manager in Samsung Electronics Co., Ltd.

References


