A similarity Measure for On-line Handprinted Kanji Character Recognition*

X. Li and W. P. Dodd
School of Computer Science, University of Birmingham
Edgbaston, Birmingham B15 2TT, U.K.

Abstract

All on-line handprinted Kanji characters consist of strokes which are the loci from pen down to pen up positions. Such strokes can be classified into predefined shape primitives and hence an on-line handprinted Kanji character can be regarded as a set of shape primitives having different direction, position and size. In this paper we define a similarity measure between characters and utilise probability judgement to estimate the similarity. Using this approach, we have obtained satisfactory results in our experiment of on-line recognition which involved 20 people as the 'writers' and covered 2,000 characters.

On-line handprinted Kanji character recognition, Stroke primitive, Similarity measure, Probability judgement and best match, Similarity estimation

1 Introduction

Kanji/Chinese characters are ideograph symbols which consist of strokes of different shape, position and size. In Chinese dictionaries, such constituent strokes are classified into basic categories such as horizontal stroke, vertical stroke, left-falling stroke, right-falling stroke, turning stroke, and so on. With respect to on-line handwriting, a stroke is a locus from pen down to pen up positions, and a specific stroke is usually written in a specific direction or directional sequence.

Because Kanji characters are ideograph symbols, many researchers have concentrated on using geometrical and topological features such as line segments, corners, cross points, shapes in their recognition schemes[1-12]. On the other hand, some workers continued to base their algorithms on various transformations to define the feature space. Theoretically, geometrical and topological features can help to process characters at high speed, and hence is more adaptable to on-line recognition.

In this paper we use sixteen categories of stroke primitives (reference figure 1) as the basic elements of on-line handprinted Kanji characters and these are obviously a kind of geometrical and topological features. In our system the stroke primitives of an input character are identified by a data preprocessing procedure similar to that described in [15]. Firstly, the input character is normalised in the manner that its maximum width/height is 128 pixels and it is centered in such a 128×128

*This research is supported by Apricot Computers Limited.

BMVC 1994 doi:10.5244/C.8.24

cat.	shape	model	cat.	shape	model
0		E	8		E/NE-S/SW
1		SE	9		E/NE-S/SW-NW
2		S	10		S/SE-NW
3		SW	11		S/SW-E/SE
4		W	12		S/SE-NE/N S-E-N
5		NW	13		E/NE-S/SE-NE/N E/NE-S-E-N
6		N	14		S/SW-E-S/SW S/SW-E-S/SW-NW
7		NE	15		E-SW-SE/E-SW E-SW-E-SW-NW

Figure 1: Basic stroke primitives

square, with the top-left corner of the square located at the origin $O(0,0)$. Secondly, by stroke corner detection and stroke shape classification, each stroke of the normalised character is then classified into one of the predefined categories according to its writing direction from a current dominant(corner) point to the next one. This corner detection technique makes use of the Freeman's chain code to reflect and filter the corner information along individual strokes, see [15] for details. Figure 2 presents an example of stroke primitive classification, where the input character is shown on the top-left and the constituent strokes are shown below the input. The corners of a stroke are marked by small black squares and each stroke is classified into a category.

Based on this stroke classification, an on-line handprinted Kanji character can be regarded as a set of shape primitives having different direction, position and size. In this paper, we define a similarity measure to compare sets of primitives with each other. We also propose a probability judgement to estimate the similarity and attenuate the computation. For a handprinted input and a group of character candidates[16], the highest similarity gives the recognition.

Unlike the dynamic programming matching method[13] and the attributed string match approach[14] which use the 1-D stroke sequence to recognise an input, our approach considers the 2-D stroke positions instead of the stroke orders. The details of this approach are discussed in the following sections.

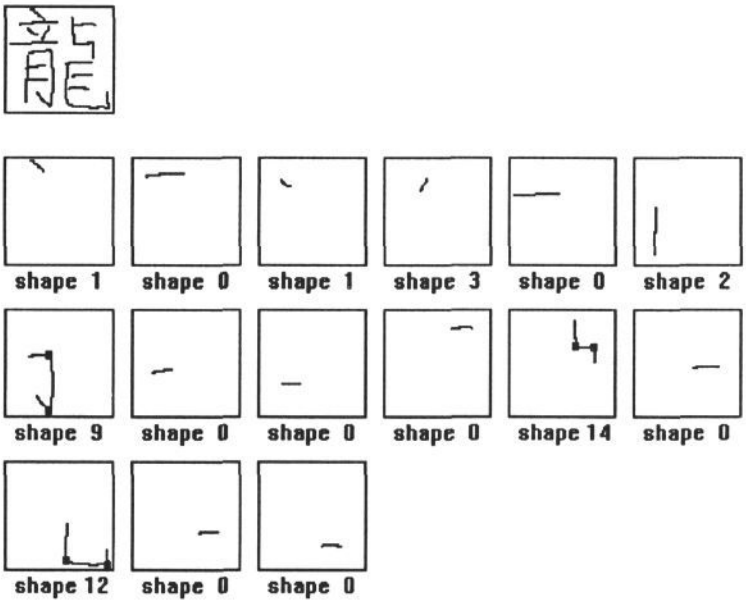

Figure 2: Stroke primitive classification

2 Similarity definition

2.1 Similarity between two stroke primitives

Suppose a and b are two stroke primitives defined in figure 1 such that

$$\begin{aligned} a &= [(x_a^{(0)}, y_a^{(0)}), (x_a^{(1)}, y_a^{(1)}), t_a, l_a] \\ b &= [(x_b^{(0)}, y_b^{(0)}), (x_b^{(1)}, y_b^{(1)}), t_b, l_b] \end{aligned} \tag{1}$$

where $(x^{(0)}, y^{(0)})$ denotes the pen-down point, $(x^{(1)}, y^{(1)})$ denotes the pen-up point, t denotes the primitive category number, and l denotes the length of the stroke. The similarity between a and b is defined as

$$s(a,b) = \begin{cases} e^{-2d(a,b)/(l_a+l_b)} & \textit{under conditions (1) or (2)} \\ 0 & \textit{otherwise} \end{cases} \tag{2}$$

where

$$d(a,b) = \sqrt{(x_a^{(0)} - x_b^{(0)})^2 + (y_a^{(0)} - y_b^{(0)})^2} + \sqrt{(x_a^{(1)} - x_b^{(1)})^2 + (y_a^{(1)} - y_b^{(1)})^2} \tag{3}$$

$$\textit{conditions (1)}: \quad \begin{array}{l} t_a, t_b \le 7,\ l_a \le 3l_b,\ l_b \le 3l_a,\ \textit{and} \\ |t_a - t_b| \le 1 \textit{ or } |t_a - t_b| = 7 \end{array} \tag{4}$$

$$conditions\ (2): \quad 7 < t_a, t_b,\ t_a = t_b,\ l_a \leq 2.5l_b,\ l_b \leq 2.5l_a \tag{5}$$

In the above equations, $d(a,b)$ is the distance between two stroke positions; conditions (1) concern with the simple strokes without 'corners' and require that only those strokes having a similar direction and a similar length can match with each other, while conditions (2) concern with the complex strokes and require that only those strokes belonging to a same category and having a similar length can match with each other.
In view of this definition, it is obvious that

$$0 \leq s(a,b) \leq 1 \quad \forall a, b \tag{6}$$

It can also be seen that for a stroke primitive a and a group of stroke primitives $\{b\}$ having identical position distances from a and satisfying the category and length conditions with respect to a, this similarity produces finer orders within this group: the longer a primitive is, the higher the similarity between it and a.

2.2 Similarity between two characters

Now let us consider two sets of stroke primitives such that

$$A = \{a_i \quad | \quad i = 1, 2, \cdots, m\} \tag{7}$$

$$B = \{b_j \quad | \quad j = 1, 2, \cdots, n\} \tag{8}$$

where $m \leq n$. Constructing a similarity matrix in the manner that

$$S = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \\ s_{21} & s_{22} & \cdots & s_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ s_{m1} & s_{m2} & \cdots & s_{mn} \end{bmatrix} \tag{9}$$

where $s_{ij} = s(a_i, b_j)$, we can define a similarity s between A and B as below:

$$s(A,B) = r_{mn} \frac{1}{m} \max\{\sum_{i=1}^{m} s_{ik_i} \quad | \quad s_{ik_i} \in S, k_i \neq k_j \forall i \neq j\} \tag{10}$$

where

$$r_{mn} = e^{-\frac{n-m}{m}}. \tag{11}$$

In view of this definition, it is obvious that

$$0 \leq s(A,B) \leq 1 \quad \forall A, B \tag{12}$$

From the above one can see that this similarity measure is stroke-order independent: it allows not only various stroke primitives having different direction, position and size to compete with each other, but also various sets having different number of stroke primitives to match with each other. In this similarity measure, although the spatial relationships between stroke primitives are not specifically defined, they are implicated in the distances between stroke positions(see eq. (3)) and reflected by the globally maximum matching score produced from the competition. This score is averaged by m and then is evaluated by the factor r_{mn}.

3 Similarity estimation

The similarity measure defined in the previous section is stroke-order free. However, its computation complexity is $O(A_n^m)$ because it needs to calculate A_n^m number of sums to determine the global maximum, where $A_n^m = n(n-1)\cdots(n-m+1)$. This is very time consuming even if m and n are not very large. In this section we propose a probability judgement to attenuate the complexity and estimate the similarity.

3.1 Probability judgement and best match

Suppose A and B are two sets of stroke primitives and S is their similarity matrix as described in previous section. We denote each possible event that $a_i \in A$ matches with $b_j \in B$ as $< a_i, b_j >$, then construct a matrix P such that

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ p_{m1} & p_{m2} & \cdots & p_{mn} \end{bmatrix} \tag{13}$$

where

$$p_{ij} = \frac{s_{ij}}{\sum_{k=1}^{n} s_{ik}}. \tag{14}$$

In view of the above equations, it is obvious that

$$\begin{array}{ll} p_{ij} \geq 0 & \forall i, j \\ \sum_{j=1}^{n} p_{ij} = 1 & i = 1, 2, \cdots, m. \end{array} \tag{15}$$

Thus, for a given i, p_{ij} form a probability distribution which indicates that how well a_i matches with b_j, i.e.

$$p_{ij} = P\{< a_i, b_j >\} \quad j = 1, 2, \cdots, n. \tag{16}$$

Based on this probability judgement, we define the globally best match of two stroke primitives in the manner below:

$$< a_p, b_q > \ \textit{is the best match} \longleftrightarrow p_{pq} = \max\{p_{ij} | p_{ij} \in P\}. \tag{17}$$

3.2 Similarity estimation

Based on the concept of the probability judgement and the globally best match, we can estimate the similarity between two characters as follows.
Let us denote the similarity matrix S as $S^{(0)}$. Starting from $S^{(0)}$ we can construct the corresponding probability matrix $P^{(0)}$ and obtain the first best match $< a_{p_1}, b_{q_1} >$. Afterwards, we can delete row p_1 and column q_1 of $S^{(0)}$ and obtain a matrix $S^{(1)}$ in reduced dimensions. Then from $S^{(1)}$ we can construct the corresponding probability matrix $P^{(1)}$ and obtain the second best match $< a_{p_2}, b_{q_2} >$.

writer's index	1	2	3	4	5	6	7	8	9	10
recognition percentage	76	98	87	83	90	90	85	96	87	91
writer's index	11	12	13	14	15	16	17	18	19	20
recognition percentage	75	88	97	93	81	90	74	76	89	92

Table 1: Recognition rates of the experiment

By repeating the above procedure, we can get m pairs of best matches $< a_{p_i}, b_{q_i} >$, $i = 1, 2, \cdots, m$. The estimate of the similarity then is

$$s(A, B) \approx r_{mn} \frac{1}{m} \sum_{i=1}^{m} s_{p_i q_i} \quad s_{p_i q_i} \in S \tag{18}$$

In view of the above algorithm, it requires $m \times n$ division operations plus some addition operations to determine the probability matrix $P^{(i)}$, $0 \leq i \leq m-1$. Therefore, the computation complexity of this algorithm is $O(nm^2)$ division operations plus some addition/subtraction operations.

4 Experimental results

So far we have described the similarity measure and its estimation. Using this approach we performed an experiment of on-line handwriting recognition which involved 20 people as the 'writers' and covered 2,000 Chinese characters.
As the first stage of this experiment, a Kanji table containing 2,118 characters was built in the system. These characters are defined by a Chinese-English dictionary and were written onto the digitizer in block style all by one person. As the second stage of this experiment, 20 people were invited to be the 'writers' and each person was assigned 100 characters to write. The first person wrote characters No. 1 to No. 100, the second No. 101 to No. 200, and so on. Instructions to participants require that each character should be written in block style, one character a time, with correct stroke number, stroke direction and stroke shape. In our system the recognition procedure contains two substages: 1) pattern candidate selection; and 2) robust matching. In the first substage, any character c_k is regarded as a repeatable combination of stroke primitives ignoring their positions and sizes:

$$c_k = \prod_{l=0}^{15} p_l^{k_l} \tag{19}$$

where p_l denotes stroke primitive of category l and $p_l^{k_l}$ denotes k_l number of appearances of p_l in c_k. For an input character a discrete similarity measure[16] is firstly applied to select a limited number of candidates(≤ 24) from all prototypes:

$$s(c_i, c_j) = \frac{\sum_{l=0}^{15} \min(i_l, j_l)}{\sum_{l=0}^{15} \max(i_l, j_l)}, \tag{20}$$

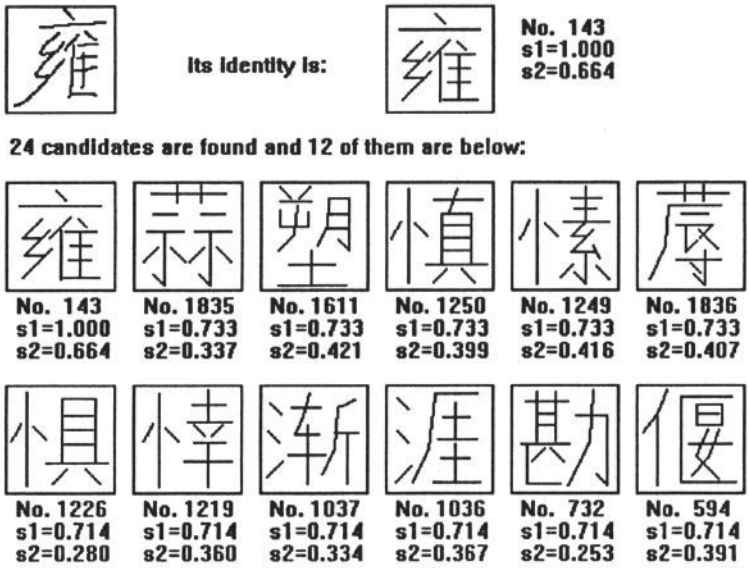

Figure 3: Recognition of character No.143

This can greatly reduce the computation intensity of robustly matching the input with all prototypes.

In the second substage, the similarity measure defined in this paper is then applied to finally identify the input from its candidates. The highest similarity which is above 0.6 gives the recognition. Otherwise, the input would be rejected.

The recognition rates of the experiment are shown in table 1. Figures 3 and 4 present two actual recognition results out of the 2,000.

From table 1 one can see that the highest individual recognition rate is 98 percent while the lowest is 74 percent. The average recognition rate from all the twenty participants is 86.9 percent.

In figures 3 and 4 each normalised input character is shown on the top-left with its identity shown on its right. Relevant information and some of the character candidates are shown below the input. In the text below each candidate, s_1 and s_2 denote the discrete similarity and the elastic similarity, respectively, and these discrete similarities have been sorted in descending order.

In this experiment, whenever a participant wrote a character with correct stroke number, stroke direction, stroke shape and stroke position, the system would identify the input correctly. On the other hand, a recognition error would occur when a participant merged two or more strokes into one, or paid no attention to stroke direction and stroke shape in his/her writing. This kind of handwriting deformation/variation remains a problem in the recognition system.

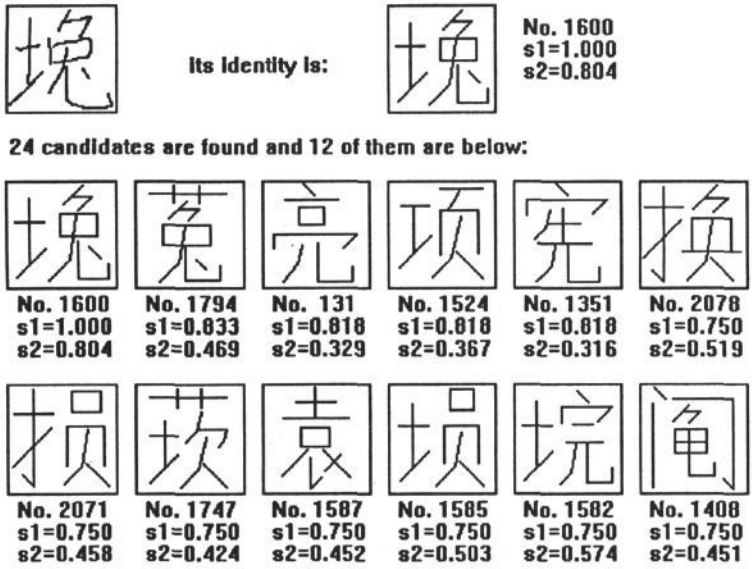

Figure 4: Recognition of character No.1600

5 Conclusions

In this paper we use sixteen kinds of stroke primitives as the basic elements of on-line handprinted Kanji characters. We regard a character as a set of stroke primitives having different direction, position and size. Based on this notation, we have defined a similarity measure to compare characters with each other. We have also proposed a probability judgement to simplify the computation and estimate this similarity. We have shown, through theoretical analysis and recognition experiment, that this similarity measure is stroke-order independent: it allows not only various stroke primitives to compete with each other, but also various sets having different number of stroke primitives to match with each other. Using this approach, we have obtained satisfactory results in our experiment of on-line recognition which involved 20 people as the 'writers' and covered 2,000 Kanji/Chinese characters. The average recognition rate was 86.9 percent with the individual rate varying from 74 to 98 percent. The recognition errors were mainly due to deformed writing, and this remains a problem in the recognition system.

References

1. C. C. Tappert, C. Y. Suen and T. Wakahara, 'The state of the art in on-line handwriting recognition', IEEE Trans. Pattern Anal. Machine Intell., Vol.12, No.8, pp.787-808, 1990

2. Katsuo Ikeda, Takashi Yamamura, Yasumasa Mitamura, Shiokazu Fujiwara, Yoshiharu Tominaga and Takeshi Kiyono, 'On-line recognition of handwritten characters utilising positional and stroke vector sequence', Pattern Recognition, Vol.13, No.3, pp.191-206, 1981

3. Horiki Arakawa, 'On-Line recognition of handwritten characters — Alphanumerics, Hiragana, Katakana, Kanji', Pattern Recognition, Vol.16, No.1, pp.9-16, 1983

4. Shunji Mori, Kazuhiko Yamamoto and Michio Yasuda, 'Research on machine recognition of handprinted characters', IEEE Trans. Pattern Anal. Machine Intell., Vol.PAMI-6, No.4, pp.386-405, 1984

5. S. L. Xie and Minsoo Suk, 'On machine recognition of handprinted characters by feature relaxation', Pattern Recognition, Vol.21, No.1, pp.1-7, 1988

6. S. L. Xie and Minsoo Suk, 'On machine recognition of handprinted Chinese characters by feature relaxation', Pattern Recognition, Vol.21, No.1, pp.1-7, 1988

7. Iwao Sekita et al, 'Feature extraction of handwritten Japanese characters by spline functions for relaxation match', Pattern Recognition, Vol.21, No.1, pp.9-17, 1988

8. Chia-Wei Liao and Jun S. Huang, 'A transformation invariant matching algorithm for handwritten Chinese character recognition', Pattern Recognition, Vol.23, No.11, pp.1167-1188, 1990

9. F. H. Cheng, W. H. Hsu and M. Y. Chen, 'Recognition of handwritten Chinese characters by modified Hough transform technique', IEEE Trans. Pattern Anal. Machine Intell., Vol.11, No.4, pp.429-439, 1989

10. F. H. Cheng, W. H. Hsu and C. A. Chen, 'Fuzzy approach to solve the recognition problem of handwritten Chinese characters', Pattern Recognition, Vol.22, No.2, pp.133-141, 1989

11. Hsi-Jian Lee and Bin Chen, 'Recognition of handwritten Chinese characters via short line segments', Pattern Recognition, Vol.25, No.5, pp.543-552, 1992

12. F. H. Cheng, W. H. Hsu and M. C. Kuo, 'Recognition of handprinted Chinese characters via stroke relaxation', Pattern Recognition, Vol.26, No.4, pp.579-593, 1993

13. C. K. Lin, K. C. Fan and F. T. Lee, 'On-line recognition by deviation-expansion model and dynamic programming matching', Pattern Recognition, Vol.26, No.2, pp.259-268, 1993

14. Y. T. Tsay and W. H. Tsai, 'Attributed string match by split-and-merge for on-line Chinese character recognition', IEEE Trans. Pattern Anal. Machine Intell., Vol.15, No.2, pp.180-185, 1993

15. X. Li and N. S. Hall, 'Corner detection and shape classification of on-line handprinted Kanji strokes', Pattern Recognition, Vol.26, No.9, pp.1315-1334, 1993

16. X. Li, N. S. Hall and G. W. Humphreys 'Discrete distance and similarity measures for pattern candidate selection', Pattern Recognition, Vol.26, No.6, pp. 843-851, 1993