

Short Paper

Candidate Selection in On-Line Chinese Character Recognition System Using Voting Scheme

CHIEN-CHENG TSENG*, BOR-SHENN JENG AND KUO-SEN CHOU

**Department of Computer and Communication Engineering
National Kaohsiung First University of Science and Technology
Yuanchau, Kaohsiung, Taiwan 824, R.O.C.
Applied Research Laboratory
Telecommunication Laboratories
Chunghwa Telecom Co., Ltd.,
Yang-Mei, Taoyuan, Taiwan 326, R.O.C.*

In this paper, a candidate selection method using a voting scheme is proposed for speeding up a on-line Chinese character recognition system. Three steps in this method are described as follows: First, several simple features are extracted from the input ink data, such as peripheral code. Then, the number of votes which denote the coarse matching scores between input features and 5401 reference features of Chinese characters are computed. Finally, the templates of Chinese characters whose votes are higher than a prescribed threshold are selected to perform detailed matching. Some experimental results show that the proposed selection method is a suitable tool for speeding up a character recognition system under the condition of maintaining the recognition rate.

Keywords: on-line Chinese character recognition, candidate selection, voting scheme, peripheral code, scalar feature

1. INTRODUCTION

Friendly human-machine interfaces are important for office automation. For a long time, the keyboard has been the most popular device for communicating with a computer. Yet when dealing with non-letter-spelling symbols, like Chinese characters with an immense practical vocabulary of 5401 categories, it is a formidable job to key in huge amounts of data encountered in daily life, especially for a person who is not familiar with the use of any Chinese input keyboard. Therefore, developing reliable on-line Chinese character recognition (OLCCR) techniques has been a promising research area [1-5].

In an OLCCR system, pen and ink data read from a digital tablet is first preprocessed to extract useful features. Then, based on the extracted features, the recognition engine of the OLCCR classifies the input character into a Chinese code, e.g., a Big5 code. The pri-

Received January 15, 1996; accepted February 24, 1998.
Communicated by Zen Chen.

many difficulties of OLCCR is that (1) there are 5401 categories, (2) the structures of some Chinese characters are very complicated and many of them are very similar, and (3) the style in which the same character is written may be quite different from person to person.

Nevertheless, many efficient techniques have been developed to recognize on-line handwritten Chinese characters. The OLCCR system is usually decomposed into two phases: candidate selection and detailed matching, as depicted in Fig.1 [6]. Some obvious fundamental features used as the candidate selection parameters are the number of input strokes, peripheral shape, structural type etc. Two requirements for candidate selection are that the selection time for each candidate should be as short as possible, and that the number of candidates selected should be as small as possible. As for detailed matching, several matching methods have been adopted, such as string matching, dynamic programming matching, relaxation methods etc. Two requirements for detailed matching are fast matching and a high recognition rate.

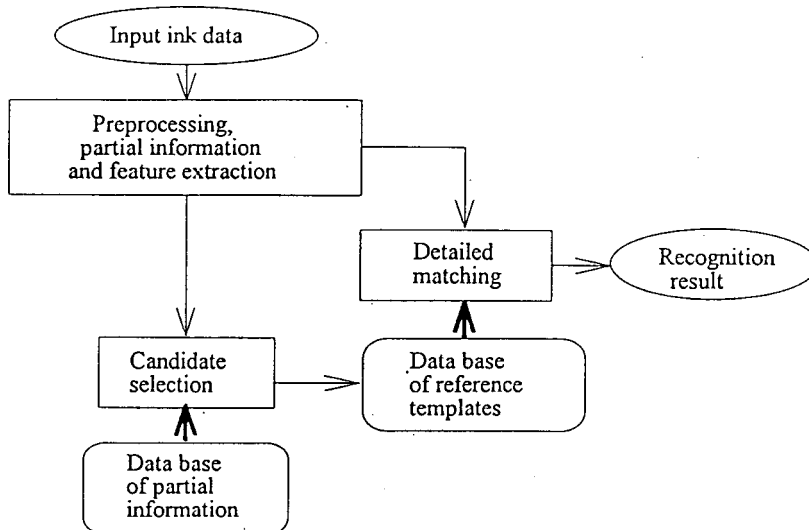


Fig. 1. Basic block diagram of a recognition system.

In this paper, a voting scheme is developed to select possible candidates from 5401 Chinese templates in order to reduce the time needed for detailed matching. Three steps involved are described as follows: First, several features used in candidate selection are extracted from the input ink data. Second, the number of votes which denote the coarse matching scores between the input features and 5401 reference features of Chinese characters are computed. Third, the templates of Chinese characters whose numbers of votes are higher than a prescribed threshold are fetched to perform detailed matching. As for the threshold values, we propose two methods to obtain them. One is a single threshold method, and the other is an individual threshold approach. The performance of these two methods will be investigated in detail.

This paper is organized as follows. In section 2, several simple features of Chinese characters will be described, including peripheral features and scalar features. Then, two candidate selection methods based on a voting scheme are developed in section 3. One is a single threshold method, and the other is an individual threshold approach. Finally, some experimental results are used to evaluate the performance of the proposed methods.

2. SOME FEATURES IN CHINESE CHARACTER

In this section, some simple features of Chinese character will be described. Based on these features, a candidate selection algorithm using a voting scheme will be developed in the next section. In order to extract these features, three preprocessing tasks of ink data must be performed in advance. They are to (1) remove redundant points and noise points, (2) extract the ending points of each segment, and (3) normalize the size of the ink data to 128×128 . After preprocessing, the ink data of Chinese character c_i are composed of several segments with two ending points (x_{ij0}, y_{ij0}) and (x_{ij1}, y_{ij1}) for $j = 1 \dots N$. Also, the directional angle of each segment is calculated as

$$\theta_{ij} = \tan^{-1}\left(\frac{y_{ij1} - y_{ij0}}{x_{ij1} - x_{ij0}}\right). \tag{1}$$

Thus, the ink data of character c_i are represented by the string $s_{i1}, s_{i2}, \dots, s_{iN}$, where s_{ij} are given by

$$s_{ij} = (x_{ij0}, y_{ij0}, x_{ij1}, y_{ij1}, \theta_{ij}). \tag{2}$$

Note that the string s_{ij} is arranged according to the writing order of the ink data, so θ_{i1} is the angle of the first segment and θ_{iN} is the angle of the last segment. These two angles will be used as scalar features for performing candidate selection. Fig. 2 shows a typical example

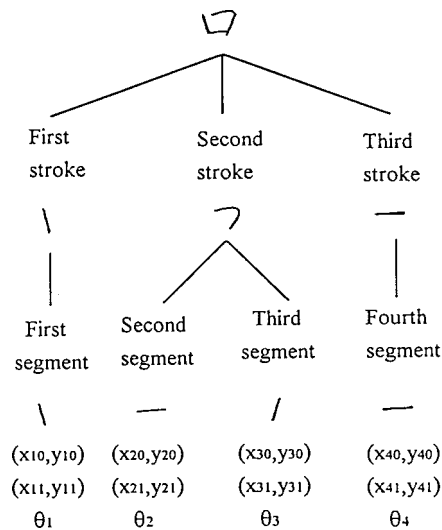


Fig. 2. An example of a Chinese character with three strokes and four segments.

of a Chinese character with four segments and three strokes. Now, several features will be described. For each feature f , we need to provide the following three rules: First is an extraction rule, i.e., how to extract feature f from the preprocessed ink data of a character. Second is a construction rule, i.e., how to combine N features $f(1), f(2), \dots, f(N)$ into a reference feature t . Note that $f(k)$ is the feature f of the ink data written by the k th user. Third is a scoring rule, i.e., how many votes will be given when we check the agreement between feature f and reference t . Based on these rules, we discuss some typical features below.

2.1 Scalar Feature

The extraction rule of a scalar feature defines a mapping T which transforms the string $s_{i1}, s_{i2}, \dots, s_{iN}$ of character c_i into a scalar number z_i . As an example, the scalar feature is the sum of the successive angle difference whose mapping T is defined by

$$z_i = T(s_{i1}, s_{i2}, \dots, s_{iN}) = \sum_{j=2}^N \theta_{ij} - \theta_{i,j-1}. \quad (3)$$

Assume there are M ink data of the character c_i written by M different persons. Using the mapping T , we obtain M scalar features $z_i(n)$ ($n = 1, \dots, M$). The construction rule of reference w^i is used to find an interval $[a_i, b_i]$ such that z_i is distributed on it. To do this, we define the minimum z_i^{\min} and the maximum z_i^{\max} as

$$\begin{aligned} z_i^{\min} &= \min\{z_i(1), z_i(2), \dots, z_i(M)\} \\ z_i^{\max} &= \max\{z_i(1), z_i(2), \dots, z_i(M)\}. \end{aligned} \quad (4)$$

Then, a simple and efficient way to estimate the interval $[a_i, b_i]$ is given by

$$\begin{aligned} a_i &= z_i^{\min} - \alpha d \\ b_i &= z_i^{\max} + \alpha d, \end{aligned} \quad (5)$$

where $d = z_i^{\max} - z_i^{\min}$ and α is a small positive number. A typical value of α is $\frac{1}{8}$. Once the reference $w^i = [a_i, b_i]$ is obtained, the scoring function $v(p, w^i)$ used to measure the agreement between reference w^i and feature p is defined by

$$v(p, w^i) = \begin{cases} 1 & p \text{ in the interval } [a_i, b_i] \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

2.2 Peripheral Feature

The peripheral features are the outstanding directional features at the top, bottom, left, and right sides of a character, as shown in Fig. 3. The extraction rule for the top side of character c_i is given as follows: Take the first four segments from top to down and arrange them from left to right. The corresponding angle sequence of the segments is denoted by $\theta_{11}^i, \theta_{12}^i, \theta_{13}^i, \theta_{14}^i$. Look up Table 1; we obtain the top code as $z_{11}^i, z_{12}^i, z_{13}^i, z_{14}^i$. Apply the same concept, we can obtain the bottom code $z_{21}^i, z_{22}^i, z_{23}^i, z_{24}^i$, left code $z_{31}^i, z_{32}^i, z_{33}^i, z_{34}^i$, and right code $z_{41}^i, z_{42}^i, z_{43}^i, z_{44}^i$.

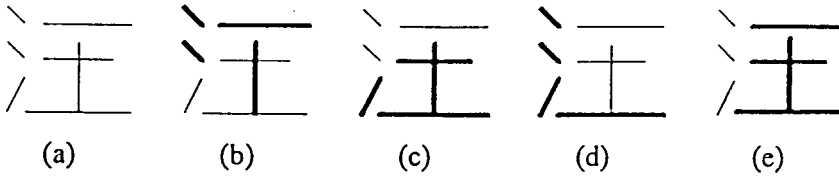


Fig. 3. (a) Ink data (b) the first four outstanding segments in the top direction, (c) the first four outstanding segments in the bottom direction, (d) the first four outstanding segments in the left direction, and (e) the first four outstanding segments in the right direction.

Table 1. The encoding table of the direction angle θ of the segment.

Range of angle θ	Binary code
185-245	(0, 1, 0, 0)
245-255	(0, 1, 1, 0)
255-285	(0, 0, 1, 0)
285-295	(0, 0, 1, 1)
295-340	(0, 0, 0, 1)
340-350	(1, 0, 0, 1)
350-360 and 0-70	(1, 0, 0, 0)

Before discussing the construction and scoring rules, we will introduce the following two operations and one function. (1) The OR operation is used to combine two codes into one code using the logical OR operation bit by bit. For example, $(1, 0, 1, 0)OR(1, 1, 0, 0) = (1, 1, 1, 0)$. (2) The AND operation is used to combine two codes into one code using the logical AND operation bit by bit. For example, $(1, 0, 1, 0)AND(1, 1, 0, 0) = (1, 0, 0, 0)$. (3) The notch function is defined by

$$S(code) = \begin{cases} 1 & code \neq (0, 0, 0, 0) \\ 0 & code = (0, 0, 0, 0). \end{cases} \quad (7)$$

Assume there are M ink data of the character c_i written by M different persons. After extraction is performed, the peripheral codes are denoted by $z_{mn}^i(j)$ for $1 \leq j \leq M$ and $1 \leq m, n \leq 4$. Then, the reference feature t_{mn}^i is constructed by

$$t_{mn}^i = z_{mn}^i(1)ORz_{mn}^i(2)OR\dots ORz_{mn}^i(M). \quad (8)$$

Once t_{mn}^i is obtained, the scoring function $u(t_{mn}^i, z_{mn}^i)$ used to measure the agreement between reference t_{mn}^i and feature z_{mn}^i is defined by

$$u(t_{mn}^i, z_{mn}) = \sum_{m=1}^4 \sum_{n=1}^4 S(z_{mn} \text{ AND } t_{mn}^i). \quad (9)$$

3. CANDIDATE SELECTION USING A VOTING SCHEME

Three steps in the proposed candidate selection method using a voting scheme are described as follows. When the input ink data is given, we first extract its peripheral feature z_{mn} ($1 \leq m, n \leq 4$) and four scalar features, which are the sum of the successive angle difference p_1 , the number of horizontal segments p_2 , the angle of the first segment p_3 and the angle of the last segment p_4 . Then, we visit the reference features of i th character prestored in memory and compute the votes β_i by checking the agreement between the reference features and input ink features for $i = 1 \dots 5401$. Finally, the templates with votes β_i higher than the threshold thd_i need to perform the detailed matching process. So far, candidate selection using the voting scheme has been described. The remaining problem is to determine the threshold value thd_i for $i = 1 \dots 5401$. In the following, two methods will be discussed. One is a single threshold method, and the other is an individual threshold method.

3.1 Single Threshold Method

In this method, we want to control the number of templates which enter the detailed matching process to be a constant Φ . Furthermore, the threshold values thd_i for $i = 1 \dots 5401$ are chosen to be a constant THD , i.e.,

$$thd_i = THD \quad i = 1 \dots 5401. \quad (10)$$

The block diagram of this candidate selection method is shown in Fig. 4, and the procedure is described below:

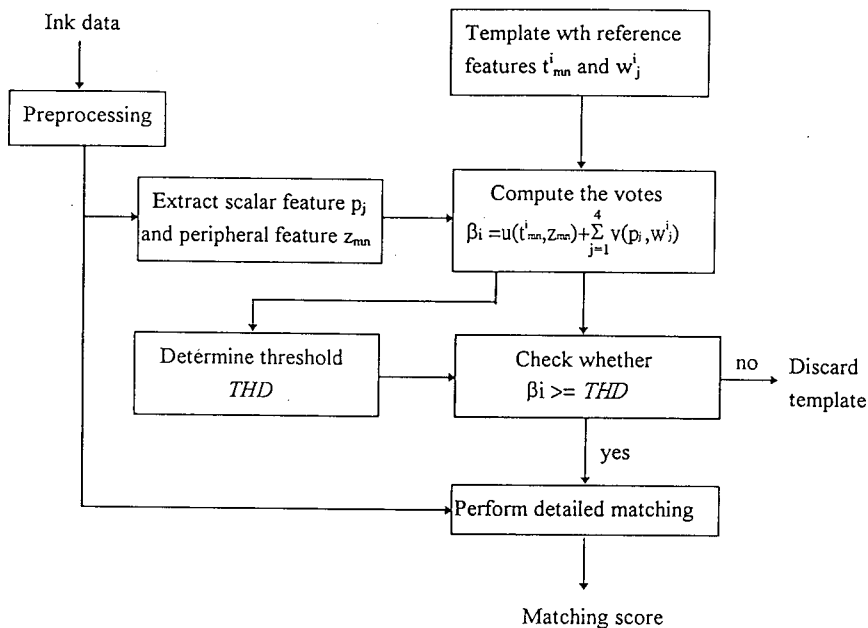


Fig. 4. The implementation block diagram of the single threshold method.

Step 1: Extract the peripheral feature z_{mn} and scalar features p_j from the input ink.

Step 2: For $i = 1$ to 5401 do

- (a) From memory, fetch the reference peripheral feature t_{mn}^i ($1 \leq m, n \leq 4$) and reference scalar features which consist of the sum of the successive angle difference ω_1^i , the number of horizontal segments ω_2^i , the angle of the first segment ω_3^i and the angle of the last segment ω_4^i .
- (b) Compute the votes β_i using the following formula:

$$\beta_i = u(t_{mn}^i, z_{mn}) + \sum_{j=1}^4 v(p_j, \omega_j^i), \quad (11)$$

where the scoring functions $u(\cdot)$ and $v(\cdot)$ are defined in eqs.(6) and (9). Note that the maximum value of β_i is 20, and that the minimum value is zero.

Step 3: Let the number of characters with vote $\beta_i = q$ be $N(q)$. Then, the threshold THD can be determined by finding a THD which satisfies the following two constraints:

$$\begin{aligned} \sum_{q=THD+1}^{22} N(q) &< \Phi \\ \sum_{q=THD}^{22} N(q) &\geq \Phi \end{aligned}$$

Step 4: For $i = 1$ to 5401 do

- (a) If $\beta_i \geq THD$, then go to the detailed matching process between input ink and the template of the i th character.
- (b) If $\beta_i < THD$ then skip the detailed matching process between input ink and the template of the i th character; i.e, discard the template.

In this paper, the constant Φ is set to be 75. Note that the larger Φ is, the longer the recognition time is and the higher the recognition rate is. Thus, in choosing Φ , the tradeoff between recognition time and recognition rate must be considered. The advantage of the single threshold method is that it is not necessary to save 5401 threshold values thd_i in memory in advance. However, it is necessary to compute the value THD for each input ink datum.

3.2 Individual Threshold Method

In this method, the threshold values of each character are different, and the number of templates which enter the detailed matching process is a random number. In order to determine the threshold value thd_i , we assume that there are M ink data of the character c_i written by M different persons. After performing feature extraction, we obtain the peripheral features $z_{mn}^i(k)$ and scalar features $p_j^i(k)$ for $k = 1 \dots M$. We fetch the reference features t_{mn}^i and ω_j^i from memory, and we obtain $\beta_i(k)$ ($k = 1, \dots, M$) using the formula

$$\beta_i(k) = u(t_{mn}^i, z_{mn}^i(k)) + \sum_{j=1}^4 v(p_j^i(k), \omega_j^i). \quad (12)$$

Now, the threshold value thd_i is given by

$$thd_i = \beta_i^{min}, \quad (13)$$

where β_i^{min} is defined by

$$\beta_i^{min} = \min\{\beta_i(1), \beta_i(2), \dots, \beta_i(M)\}. \quad (14)$$

Note that M must be large enough to guarantee a better estimate of the threshold value thd_i . However, when the estimate of thd_i is asymptotically stable, it is no use to increase M . Based on extensive experiments, we suggest that $M = 40$ is a good choice. Once the threshold values thd_i are obtained, they are stored in memory in a knowledge table. The block diagram of the individual threshold method is shown in Fig. 5, and the procedure is described below:

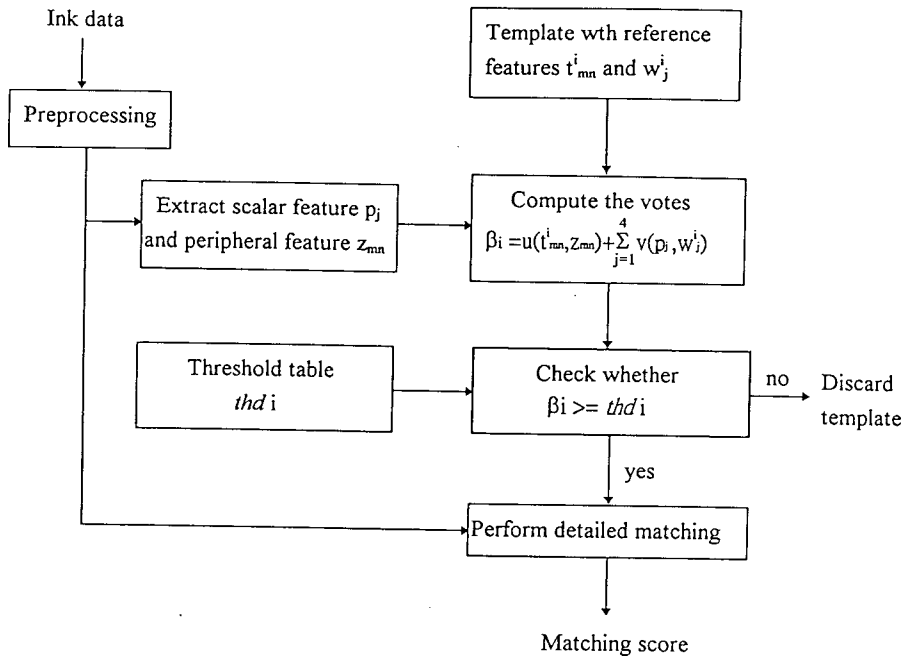


Fig. 5. The implementation block diagram of the individual threshold method.

Step 1: Extract the peripheral features z_{mn} and scalar features p_j from the input ink.

Step 2: For $i = 1$ to 5401 do

- (a) Fetch the reference peripheral feature t_{mn}^i ($1 \leq m, n \leq 4$) and reference scalar features, which consist of the sum of the successive angle difference ω_1^i , the number of horizontal segments ω_2^i , the angle of the first segment ω_3^i and the angle of the last segment ω_4^i .

(b) Compute the votes β_i using the following formula:

$$\beta_i = u(t_{mn}^i, z_{mn}) + \sum_{j=1}^4 v(p_j, \omega_j^i), \quad (15)$$

where the scoring functions $u(\cdot)$ and $v(\cdot)$ are defined in eq(6) and (9). Note that the maximum value of β_i is 20, and the minimum value is zero.

(c) If $\beta_i \geq thd_i$, then go to the detailed matching process between input ink and the template of the i th character.

(d) If $\beta_i < thd_i$, then skip the detailed matching process between input ink and the template of the i th character; i.e, discard the template.

The drawback of the individual threshold method is that we need memory to save the 5401 threshold values in advance. However, this method is not required computation of the value THD for each input ink data as the single threshold method does.

4. EXPERIMENTS

So far, we have introduced a voting scheme for candidate selection in an OLCCR system developed at Telecommunication Laboratories (TL), Taiwan. In this section, some experiments are described to evaluate the performance of this approach.

4.1 Experimental Data

Ten sets of handwritten Chinese characters in the TL database, i.e., from TL-OLCCR-A01 to TL-OLCCR-A10, were used to estimate the reference peripheral features t_{mn}^i and scalar features ω_j^i of the i th character. Each data set contained 5401 characters and was written by a different person in a natural way without any constraint imposed on the person. When one of the ten character sets was used to test the performance of proposed method, this test was called inside testing. In contrast, when a character set which did not belong to these ten sets was adopted to test the performance, we called this outside testing. The performance of both tests will be reported in the following.

4.2 Performance

Three experiments were done on the 4 sets of handwritten Chinese characters shown in Fig. 6. It is clear that character sets 1, 2, 4 are well written, but that set 3 is cursive writing with connected strokes. We will report the average performance of these 4 character sets. The first experiment concentrated on the performance of the original OLCCR system developed at TL. In this system, the stroke number and radical are used to perform candidate selection. The average number of candidates selected for detailed matching was about 130. The purpose of this study was to reduce this number by using other features and voting scheme. The second experiment used the selection method with a constant threshold to select possible candidates. The third experiment used the selection method with an individual threshold to perform candidate selection. That is:

Experiment 1: Original OLCCR system.

Experiment 2: Experiment 1 + candidate selection with the single threshold method.

Experiment 3: Experiment 1 + candidate selection with the individual threshold method.

乾 僭 偽 停
 借 偵 側 偷
 副 勒 務 勤
 曼 商 啗 啦

(a) set 1

乾 僭 偽 停
 借 偵 側 偷
 副 勒 務 勤
 曼 商 啗 啦

(b) set 2

乾 僭 偽 停
 借 偵 側 偷
 副 勒 務 勤
 曼 商 啗 啦

(c) set 3

乾 僭 偽 停
 借 偵 側 偷
 副 勒 務 勤
 曼 商 啗 啦

(d) set 4

Fig. 6. Chinese character samples of 4 test sets. (a) set 1, (b) set 2, (c) set 3, (d) set 4.

In Table 2, the number of candidates which selected for detailed matching is listed for the 4 character sets and 3 experiments. It is clear that the average number of candidates was reduced from 130 to 80 by using our methods. In experiment 2, the number of candidates was always equal to 75 because this number was used to choose the threshold *THD* under the single threshold method. In experiment 3, the number of candidates was not a fixed integer and depended on the test character set. In our experience, the average number of candidates in the individual threshold method was 85.

Table 2. The number of candidates selected for detailed matching.

	Experiment 1	Experiment 2	Experiment 3
set 1	120	75	80
set 2	137	75	85
set 3	143	75	86
set 4	136	75	83

In Table 3, the average recognition time and recognition rate of the 4 sets of data are shown. Comparing Exp.(1) with Exp.(2), we see that the recognition time was reduced in the inside and outside tests, but that the recognition rate improved in the inside test and was slightly degraded in the outside test. Comparing Exp.(1) with Exp.(3), the same result can be observed. Thus, the proposed candidate selection methods are suitable tools for speeding up an OLCCR system under the condition of keeping the recognition rate constant.

Table 3. Average recognition rate and recognition time (in seconds).

		Exp. 1		Exp. 2		Exp.3	
		Recog. rate	Recog. time	Recog. rate	Recog. time	Recog. rate	Recog. time
Inside test	Set 1	97.76	0.595	98.06	0.327	97.91	0.396
	Set 2	97.48	0.575	97.61	0.314	97.57	0.382
Outside test	Set 3	95.93	0.637	94.87	0.334	95.31	0.412
	Set 4	95.28	0.589	95.13	0.325	95.19	0.388

5. CONCLUSIONS

In this paper, a voting scheme for candidate selection in an on-line Chinese character recognition system has been proposed. Some experimental examples have been given to show that the proposed selection methods are suitable tools for speeding up an OLCCR system under the condition of keeping the recognition rate constant. In the future, it will be important to find useful and efficient scalar features except some typical ones described in this paper.

ACKNOWLEDGMENTS

The authors would like to thank Dr. J.T. Wang, managing director of the Telecommunication Laboratories, and Dr. I.C. Jou for their constant encouragement.

REFERENCES

1. E.F. Yhap and E.C. Greanias, "A on-line Chinese character Recognition System," *IBM Journal of Research Development*, Vol. 25, No. 3, 1981, pp. 187-195.
2. Kazumi Odaka, et al, "On-line recognition of handwritten characters by approximating each stroke with several points," *IEEE Transactions on System, Man and Cybernetics*, Vol. 12, No. 6, 1982, pp. 898-903.
3. H. Arakawa, "On-line recognition of handwritten characters-alphanumerics, Hiragana, Katahana, Kanji," *Pattern Recognition*, Vol. 16, No. 1, 1983, pp. 9-16.
4. Y.T. Tsay and W.H. Tsai, "Attributed string matching by split-and-merge for on-line Chinese character recognition," *IEEE Transactions on Pattern Analysis Machine Intelligence*, Vol. 15, No. 2, 1993, pp.180-185.
5. K.S. Chou, K.C. Fan, T.I. Fan, C.K. Lin and B.S. Jeng, "Knowledge model based approach in recognition of on-line Chinese characters," *IEEE Journal on Selected Areas in Communications*, Vol. 12, No. 9, 1994, pp. 1566-1575.

6. T. Kumamoto, K. Toraichi et al, "On speeding candidate selection in handprinted Chinese character recognition," *Pattern Recognition*, Vol. 24, No. 8, 1991, pp.793-799.

Chien-Cheng Tseng(曾建誠) was born in Taipei, Taiwan, R.O.C., on August 25, 1965. He received the B.S. degree, with honors, from Tatung Institute of Technology, Taipei, in 1988, and the M.S. and Ph.D. degrees from National Taiwan University, Taipei, in 1990 and 1995, respectively, all in electrical engineering.

From 1995 to 1997, he was an associate research engineer at Telecommunication Laboratories, Chunghwa Telecom Co., Ltd. in Taoyuan, Taiwan. He is currently an assistant professor in the Department of Computer and Communication Engineering at National Kaohsiung First University of Science and Technology. Dr. Tseng is a member of IEEE. His research interests include digital signal processing, pattern recognition, and electronic commerce.

Bor-Shenn Jeng(鄭伯順) received the B.S. degree in physics from National Normal University, Taiwan, in 1969, the M.S. degree in geophysics in 1973 and the Ph.D degree in optical sciences in 1990, both from National Central University, Taiwan. He taught physics at Chinese Cultural University from 1973 to 1974 and has been with the Telecommunication Laboratories, Directorate General of Telecommunications, Taiwan, since 1974. Now he is the vice president of Telecommunication Laboratories. He has been working on multimedia, intelligent human/machine interface, Chinese character, recognition, generation and compression, etc. He has published more than 150 technical papers and 30 patents in the areas of physics and signal processing. Dr. Jeng was the recipient of the Distinguished Performance in Information Science Award conferred by the National Science Council, China, in 1988 and the Distinguished Performance in Technology Award conferred by Executive Yuan, China, 1989, which is the highest technology award conferred by the Chinese government.

Kuo-Sen Chou(周國森) received the B.S. and M.S. degrees in automatic engineering from Fung-Chang University, Taiwan, in 1987 and 1989, respectively. He received Ph. D. degree in 1997 from the Institute of Computer Science and Information Engineering, National Central University in Taiwan.

Since 1990 he has been an assistant researcher at Telecommunication Laboratories, Chung-Hwa Telecom Co., Ltd., in Taiwan. His research interests are in artificial intelligence, pattern recognition, and Chinese information processing.