

## HANDWRITTEN NUMERICAL RECOGNITION BASED ON MULTIPLE ALGORITHMS

F. KIMURA\* and M. SHRIDHAR†

The University of Michigan-Dearborn, Dearborn, Michigan, U.S.A.

(Received 26 October 1990; in revised form 28 February 1991; received for publication 20 March 1991)

**Abstract**—In this paper, the authors combine two algorithms for application to the recognition of unconstrained isolated handwritten numerals. The first algorithm employs a modified quadratic discriminant function utilizing direction sensitive spatial features of the numeral image. The second algorithm utilizes features derived from the profile of the character in a structural configuration to recognize the numerals. While both algorithms yield very low error rates, the authors combine the two algorithms in different ways to study the best polling strategy and realize very low error rates (0.2% or less) and rejection rates below 4%.

Character recognition    Statistical pattern recognition    Bayes classifier  
Quadratic discriminant function    Structural pattern recognition  
Combined character recognition algorithm

### 1. INTRODUCTION

In the last decade, significant advances have been made in machine recognition of shapes, patterns and characters. Of particular interest to many investigators is the recognition of handwritten characters, both isolated and cursive. In this paper, the authors focus their attention on the recognition of isolated handwritten numerals, such as those encountered in Zip codes, bank checks, etc. It suffices to mention that such systems are also capable of recognizing machine printed numerals.

Typical requirements of a commercial numeral recognition system (for example Zip code reading in an address block) can be summarized as follows:

- (a) the recognition system is writer-independent,
- (b) the recognition system can handle numerals of arbitrary size,
- (c) the recognition system is robust in the presence of noise or varying background,
- (d) the recognition system will have very low error rates and low rejection rates,
- (e) the system can operate at high speeds for commercial application.

It is generally agreed that such stringent requirements on the recognition system would require the use of multiple recognition algorithms and a polling strategy to determine the identity of the unknown sample. Further, it would be desirable to control the error rates for a specific application, by simply adjusting the rejection thresholds.

Several high accuracy algorithms have recently been proposed for recognition of handwritten numerals.<sup>(1-12)</sup> Among the many techniques that have been proposed the authors would like to mention the works of Shridhar *et al.*,<sup>(5,6)</sup> Kimura *et al.*,<sup>(7,8)</sup> Lam and Suen,<sup>(9)</sup> Suen *et al.*<sup>(10)</sup> and Srihari *et al.*<sup>(11,12)</sup>

In an interesting paper,<sup>(9)</sup> Lam and Suen describe a recognition system that consists of a sequential combination of a fast structural classifier and a robust relaxation algorithm. The classification is based on the configuration of a set of primitives derived from the image of the numeral. Although very low error rates are realized, the method is relatively slow, owing to an extensive preprocessing of the numeral image prior to feature extraction and the complexity of the relaxation algorithm.

Srihari *et al.* proposed a recognition system<sup>(11)</sup> that utilizes three algorithms: (1) a template matching algorithm, (2) a mixed statistical and structural classifier utilizing features derived from the contours, and (3) a structural classifier utilizing features such as size and stroke placement, etc. The results derived from the three algorithms are combined in a logical manner to arrive at the final decision on the identity of the test numeral. The final accuracies were reported to be significantly higher than those achieved with the individual algorithms.

Kimura and his co-workers had developed a statistical classification technique<sup>(7,8)</sup> that utilized the histogram of the direction vector derived from the contours of the character. Although the technique was developed for recognition of Chinese characters, the method could easily have been adapted for numeral recognition.

Shridhar *et al.* had reported a high speed accuracy structural recognition algorithm that utilized features

\* On leave from the Department of Electronics, Mie University, Kamihama-cho 1515, Tsu 514, Japan.

† Author to whom correspondence should be addressed.

derived primarily from the left and right profiles of the numeral images.<sup>(5,6)</sup> High speed and good accuracies were reported; however, this algorithm as well as the statistical classifier did not provide for the rejection of bad samples.

In this paper the authors further develop the statistical classification technique of Kimura *et al.* and the profile analysis technique of Shridhar and Badreldin to realize a fast, very low error rate recognition algorithm. This algorithm allows the user to adjust the error rate by simply adjusting the rejection threshold.

At the time of writing this paper, the authors had achieved the following error and rejection rates:

- (1) error 0.07%, rejection 10.38%
- (2) error 0.13%, rejection 4.79%
- (3) error 0.25%, rejection 3.52%.

The above results were based on a study with 20,394 samples (including 3000 design samples) of handwritten numerals supplied to the authors by CGA-Alcatel, France. Three thousand samples were used to train the classifier and the remaining 17,394 samples were used to test the classifier. The classifier implemented on a 25 MHz 80386 IBM compatible computer operated at three numerals per second.

The paper is laid out as follows: Section 2 describes the data collection, processing and feature extraction; Section 3 describes the recognition algorithms; Section 4 describes the test results with the two algorithms; Section 5 describes the improvement in performance when the two algorithms were combined in different ways; Section 6 outlines the conclusions of this study.

## 2. DATA COLLECTION AND PROCESSING

The numeral samples used in this study were supplied to the authors by a French company. The numeral samples were specifically collected for this test study. Each participant was asked to enter a random sequence of digits in a specially designed form. All of the digits used in this study were disconnected. A sample is shown in Fig. 1(a). A total of 225 participants contributed 50–80 samples each over a period of eight months. The form was processed by a scanner at 300 dpi, that yielded a binary image which was subsequently stored in a compressed format on a computer. Each numeral occupied a field that measured 64 W × 80 H pixel units. The actual height of the character varied from a low of 25 pixels to a high of 80 pixels. The width varied from 6 pixels (for the numeral “1”) to 64 pixels. Figure 1(b) shows examples of digitized numeric data.

### 2.1. Data processing for feature extraction

Since two classifiers were developed for this study, the data processing consisted of contour extraction

0 1 4 5 6 7    0 2 3 4 5 6  
 0 8 9 8 7 6    0 7 8 9 8 7  
 0 3 2 1 2 3    0 6 5 4 3 2  
 1 3 6 5 6 6    1 6 7 8 7 5

(a)

0 0 0 5 5 5  
 1 1 1 6 6 6  
 2 2 2 7 7 7  
 3 3 3 8 8 8  
 4 4 4 9 9 9

(b)

Fig. 1. (a) Samples of the numeric data; (b) a digitized image.

and normalization for extracting a set of features as well as profile extraction directly from the unpacked image of the numeral. It is interesting to observe that the profile features could also have been derived from the normalized contours of the image.

An eight-neighbor boundary extraction algorithm was applied to the binary image of the numeral to extract all the closed contours. As an example, the image of the numeral 8 yielded generally three sets of border elements—the outer boundary and the two inner boundaries. A scale normalization was applied to the extracted boundaries so that the rectangular frame enclosing the scaled contours measured 64 W × 80 H in pixel units. It is worth observing that gaps between pixels in this scaling process were filled according to a minimum error criterion. A further operation that would have been needed for the structural classifier would be the filling operation to yield a new normalized binary image of the numeral for profile extraction and the derivation of other features.

### 2.2. Feature extraction

In this subsection, the authors will describe the two sets of features that were used in the two recognition

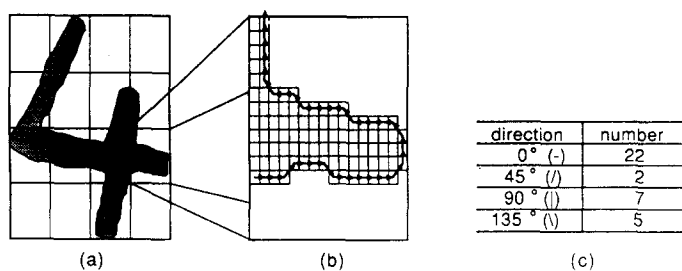


Fig. 2. Feature extraction for statistical classifier: (a) normalized numeral; (b) chain codes in a zone; (c) direction histogram in a zone.

algorithms. The first set of features is the histograms of the chain codes of the contour elements. The second set of features is evaluated from the profiles of the binary image of the numeral.

2.2.1. *Direction vector features.* In this process the rectangular frame enclosing the normalized contours is divided into  $4 \times 4$  rectangular zones. In each zone, a local histogram of the chain codes is calculated (Fig. 2). The feature vector is composed of

these local histograms. Since contour direction is quantized to one of four possible values ( $0^\circ$  or “-”,  $45^\circ$  or “/”,  $90^\circ$  or “|”, or  $135^\circ$  or “\”) a histogram in each zone has four components. Thus the feature vector has 64 components when all the 16 zones are included. Figure 2 illustrates the procedure for the numeral 4.

There were many samples that had broken components. The broken components were the result of the writing style (e.g. 5, 1) of the participants as

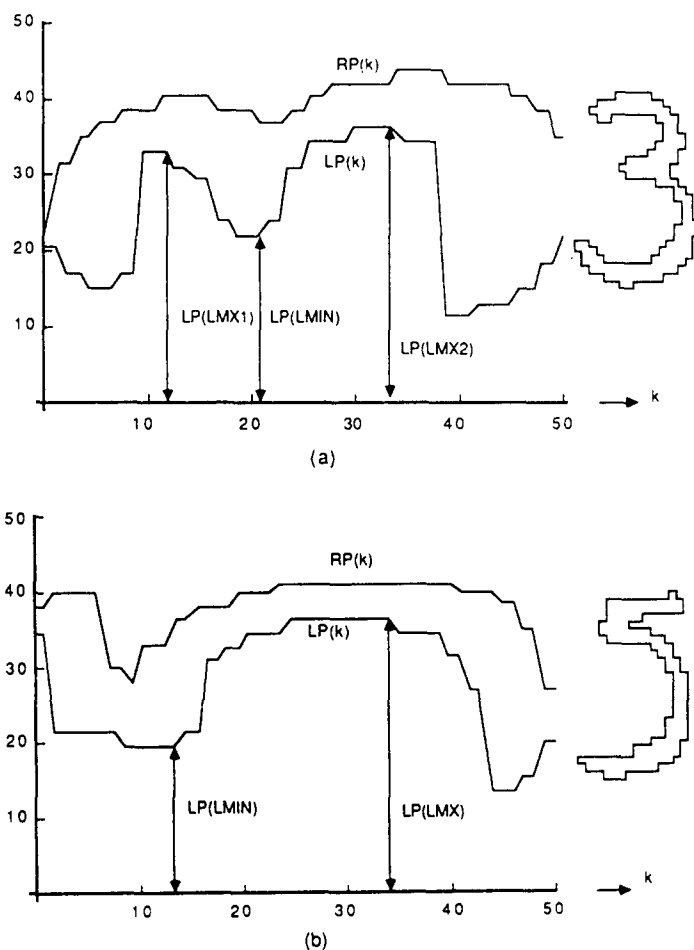


Fig. 3. (a) Right and left profiles of the character 3; (b) right and left profiles of the character 5.

well as the binarization technique used on the digit images. Although broken components affected the histogram of the direction vectors, their statistical properties were quite robust: the recognition accuracies for broken numerals were not noticeably worse than those of the unbroken numerals.

**2.2.2. Profile features.** In a structural classifier all of the features associated with the numerals are derived from profiles of their external contours. These are character widths, ratio, location of extrema, and discontinuities in character profiles. Each numeral and its subclasses are described in terms of these features.

The profiles are described by:

(a) left profiles  $\{LP(k), k = 1, 2, \dots, NP\}$  which is a collection of the distances of the left profiles from the left edge;

(b) right profiles  $\{RP(k), k = 1, 2, \dots, NP\}$  which is a collection of the distances of the right profile from the left edge.

The left and right profiles of the numerals 3 and 5 are shown in Fig. 3. The left and right profiles are used as global features. In addition, another set of global features is defined as the set of first differences of the left and right profiles:

(c)  $LDIF(k) = LP(k) - LP(k - 1), k = 2, 3, \dots, NP,$

(d)  $RDIF(k) = RP(k) - RP(k - 1), k = 2, 3, \dots, NP.$

A fifth global feature that pertains to the width is defined as:

(e)  $W(k) = RP(k) - LP(k), k = 1, 2, \dots, NP.$

Local features are then defined for obtaining specific characterization of the different numerals. These are:

(a) The ratio is defined as:

$$\text{Ratio} = NP / \text{MAX}_k \{W(k)\}.$$

(b) Location of maxima and minima:

$$\text{LMX} = \text{location of MAX}_{R_1} \{LP(k)\}$$

$$\text{RMX} = \text{location of MAX}_{R_1} \{RP(k)\},$$

where  $R_1$  is a specified range within which LMX and RMX are evaluated. In a similar manner, LMIN and RMIN are also defined.

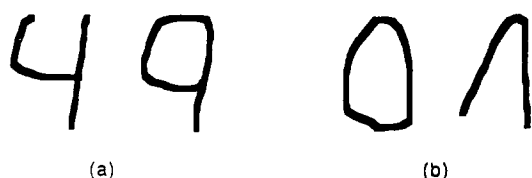


Fig. 4. An example involving the numerals, 4, 9 and 0, 1.

(c) Left and right peaks: the values of peaks in the first differences are very important local features in the characterization of numerals. These are:

$$\text{LPEAK}^+ = \text{MAX}_{R_2} \{LDIF(k)\}$$

$$\text{LPEAK}^- = \text{MIN}_{R_2} \{LDIF(k)\}$$

$$\text{RPEAK}^+ = \text{MAX}_{R_2} \{RDIF(k)\}$$

$$\text{RPEAK}^- = \text{MIN}_{R_2} \{RDIF(k)\}$$

$$\text{LPEAK} = |\text{LPEAK}^+| + |\text{LPEAK}^-|$$

$$\text{RPEAK} = |\text{RPEAK}^+| + |\text{RPEAK}^-|,$$

where  $R_2$  is a specified range on  $LDIF(k)$  or  $RDIF(k)$ .

In addition to the features described above, the locations of the peaks of the first differences are used to resolve conflicts. The bottom and top profiles are also used as local features to resolve conflicts between similarly shaped numerals. An example involving the numerals 4, 9 and 0, 1 is shown in Fig. 4.

The profile features can be affected by the presence of noise pixels and other spurious line segments. In this work, a preprocessing algorithm was used to eliminate many spurious features caused by noise pixels or breaks in the digit image. In the preprocessing stage, the profiles that exhibit transient discontinuities were subjected to a verification test that would confirm the validity of the profile features. This is illustrated in Fig. 5 for the digits 7, 6, 5 and 4. However, the preprocessing stage did not eliminate all the errors in the profile evaluator. The "reject" option of the recognition algorithm eliminated many of these problem digits.

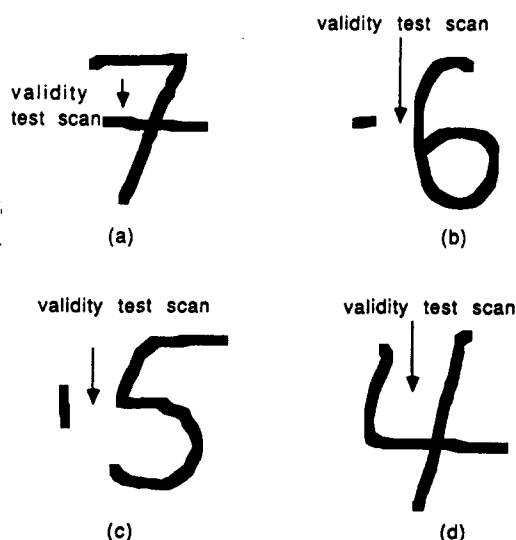


Fig. 5. Profiles with transient discontinuity: valid features (a), (d); spurious features (b), (c).

### 3. RECOGNITION ALGORITHMS

In this section, detailed descriptions of the statistical and structural classifiers are provided.

#### 3.1. Statistical classifier (K-algorithm)

The K-algorithm implements a statistical classifier employing the direction vector defined in 2.2.1. The discriminant function is a modified quadratic discriminant function (MQDF) which is less sensitive to the estimation error of the covariance matrix and requires less computation time and storage than the ordinary quadratic discriminant function (QDF).<sup>(7,8)</sup> It is given as follows.

$$g(\mathbf{X}) = \left\{ |\mathbf{X} - \hat{\mathbf{M}}|^2 - \sum_{i=1}^k \frac{\lambda_i}{\lambda_i + h^2} [\Phi_i^T (\mathbf{X} - \hat{\mathbf{M}})]^2 \right\} / h^2 + \ln [h^{2(n-k)} \prod_{i=1}^k (\lambda_i + h^2)], \quad (1)$$

where  $\mathbf{X}$  denotes the feature vector for an input numeral,  $\hat{\mathbf{M}}$  denotes the maximum likelihood estimate of the mean vector for each numeral class, and  $\lambda_i$  and  $\Phi_i$  denote the eigenvalues and eigenvectors of the maximum likelihood estimate of the covariance matrix.

Values of constants  $h^2$  and  $k$  are selected experimentally to achieve the best performance. In the following experiments,  $k$  and  $h^2$  are set to 12 and 0.2, respectively.

Given a test sample,  $g(\mathbf{X})$  is calculated for all numeral classes and the class achieving the minimum value is selected as the identity of the test sample.

The MQDF is derived from the Bayes decision rule for unknown Gaussian distribution under some assumption for simplification. The rest of this subsection briefly describes the derivation.

(1) *The Bayes discriminant function for known Gaussian distribution.* The Bayes decision rule for a multi-category case is represented as follows.

Select class  $i$  such that

$$P(\omega_i | \mathbf{X}) > P(\omega_j | \mathbf{X}) \text{ for all } j \text{ except } i \rightarrow \mathbf{X} \in \omega_i, \quad (2)$$

where  $\mathbf{X}$  denotes the feature vector of a given sample,  $\omega_i$  ( $i = 1, 2, \dots$ ) denotes  $i$ th class, and  $P(\omega_i | \mathbf{X})$  denotes the a posteriori probability of  $\omega_i$  given  $\mathbf{X}$ .

The Bayes decision rule is optimal in the sense that minimum error is achieved. The decision rule of (2) leads to several equivalent discriminant functions in different forms:

$$g_i(\mathbf{X}) = P(\omega_i | \mathbf{X}) \quad (3)$$

$$g_i(\mathbf{X}) = p(\mathbf{X} | \omega_i) P(\omega_i) \quad (4)$$

$$g_i(\mathbf{X}) = 2\{-\ln p(\mathbf{X} | \omega_i) - \ln P(\omega_i)\}, \quad (5)$$

where  $p(\mathbf{X} | \omega_i)$  denotes the conditional density function of  $\mathbf{X}$  given  $\omega_i$ , and  $P(\omega_i)$  denotes the a priori probability of  $\omega_i$ .

The discriminant function (4) is derived from (3) using Bayes theorem and neglecting the denominator  $p(\mathbf{X})$ , which is not dependent on class  $i$ . Given a feature vector  $\mathbf{X}$ , the value of  $g_i(\mathbf{X})$  of either (3) or (4) is calculated for all classes and the class achieving the maximum value is selected as the identity of  $\mathbf{X}$ . The discriminant function (5) is derived from (4) by taking the negative logarithm of  $g_i(\mathbf{X})$ . Because of the negative logarithm, the class achieving the minimum value is selected as the identity.

When the conditional distribution of  $\mathbf{X}$  given  $\omega_i$  is assumed to be Gaussian with known mean vector  $\mathbf{M}_i$  and covariance matrix  $\Sigma_i$ , the conditional density function  $P(\mathbf{X} | \omega_i)$  is given by:

$$P(\mathbf{X} | \omega_i) = (2\pi)^{-n/2} |\Sigma_i|^{-1/2} \times \exp\{- (\mathbf{X} - \mathbf{M}_i)^T \Sigma_i^{-1} (\mathbf{X} - \mathbf{M}_i) / 2\}, \quad (6)$$

where  $\mathbf{M}_i$  denotes the mean vector for class  $i$ ,  $\Sigma_i$  denotes the covariance matrix for class  $i$ , and  $n$  denotes the dimensionality of the feature vector.

Substituting (6) into (5), and neglecting the constant common to all classes, a discriminant function for known Gaussian distribution is obtained:

$$g_i(\mathbf{X}) = 2\{-\ln p(\mathbf{X} | \omega_i) - \ln P(\omega_i)\} = (\mathbf{X} - \mathbf{M}_i)^T \Sigma_i^{-1} (\mathbf{X} - \mathbf{M}_i) + \ln |\Sigma_i| - 2 \ln P(\omega_i). \quad (7)$$

The discriminant function (7) is Bayes (optimal) discriminant function for known Gaussian distribution.

(2) *The Bayes discriminant function for unknown Gaussian distribution.* If the mean vector  $\mathbf{M}_i$  and the covariance matrix  $\Sigma_i$  are not known a priori, they have to be estimated by using a set of training samples  $Z_i$ . The role of training samples can be explicitly denoted by the density function  $P(\mathbf{X} | \omega_i, Z_i)$ . Since each class can be treated independently, the class distinction  $i$  and  $\omega_i$  will be omitted in subsequent expressions. The density function is then given by

$$p(\mathbf{X} | Z) = \int p(\mathbf{X} | \Theta, Z) p(\Theta | Z) d\Theta = \int p(\mathbf{X} | \Theta) p(\Theta | Z) d\Theta, \quad (8)$$

where  $\Theta$  denotes the parameter vector of the distribution.<sup>(13)</sup> The condition  $Z$  in the conditional density  $p(\mathbf{X} | \Theta, Z)$  can be eliminated because the distribution of  $\mathbf{X}$  is known completely once we know the value of the parameter vector  $\Theta$ . The discriminant function can be re-evaluated from (8) when  $\mathbf{M}_i$  and  $\Sigma_i$  are unknown. However, for the sake of simplicity, only the case where  $\Sigma_i$  is unknown but  $\mathbf{M}_i$  is known will be considered. Under these conditions,

the density function is given by

$$\begin{aligned}
 p(\mathbf{X}|Z) &= \int p(\mathbf{X}|K)p(K|Z)dK \\
 p(\mathbf{X}|K) &= (2\pi)^{-n/2} |K|^{1/2} \\
 &\quad \times \exp\{-(\mathbf{X} - \mathbf{M})^T K (\mathbf{X} - \mathbf{M})/2\} \\
 p(K|Z) &= W(K, \Sigma_N, N_N) \\
 &= c(n, N_N) |N_N \Sigma_N / 2|^{(N_N - 1)/2} |K|^{(N_N - n - 2)/2} \\
 &\quad \times \exp[\text{tr}(-N_N \Sigma_N K) / 2] \\
 c(n, N_N) &= \left\{ \pi^{n(n-1)/4} \prod_{i=1}^n \Gamma[(N_N - i)/2] \right\}^{-1} \\
 \Sigma_N &= \frac{\left\{ (1/N) \sum_{i=1}^N (\mathbf{X}_i - \mathbf{M})(\mathbf{X}_i - \mathbf{M})^T \right\} + (N_0/N) \Sigma_0}{1 + N_0/N} \\
 N_N &= N_0 + N, \tag{9}
 \end{aligned}$$

where  $K$  denotes the inverse of the covariance matrix,  $N$  denotes the number of training samples,  $\Sigma_0$  denotes the initial guess of the true covariance matrix,  $N_0$  denotes the confidence constant about the initial guess  $\Sigma_0$ , and  $\Gamma$  denotes the Gamma function.

The conditional density of the inverse covariance matrix  $p(K|Z)$  is known as the Wishart distribution.<sup>(14)</sup> By performing the integral of (9), the conditional density  $p(\mathbf{X}|Z)$  leads to the multivariate  $t$ -distribution with  $N_N - 1$  degrees of freedom.

$$\begin{aligned}
 p(\mathbf{X}|Z) &= (N_N - 1) \pi^{-n/2} \\
 &\quad \times \frac{\Gamma[(N_N + n - 1)/2]}{\Gamma[(N_N - 1)/2]} |\Sigma_N|^{1/2} [1 + (\mathbf{X} - \mathbf{M})^T \\
 &\quad \times \Sigma_N^{-1} (\mathbf{X} - \mathbf{M}) / (N_N - 1)]^{-(N_N + n - 1)/2}. \tag{10}
 \end{aligned}$$

Taking a negative logarithm and neglecting the constant common to all classes, a discriminant function for unknown covariance case is obtained:

$$\begin{aligned}
 g(\mathbf{X}) &= (N_N + n - 1) \ln[1 + (\mathbf{X} - \mathbf{M})^T \Sigma_N^{-1} (\mathbf{X} - \mathbf{M}) / \\
 &\quad (N_N - 1)] + \ln |\Sigma_N| - 2 \ln [P(\omega)]. \tag{11}
 \end{aligned}$$

The discriminant function (11) is Bayes (optimal) discriminant function for the Gaussian distribution with unknown covariance matrix.

(3) *Modified quadratic discriminant function.* When  $N_N$  is much larger than  $n$ , (11) reduces to

$$\begin{aligned}
 g(\mathbf{X}) &= (\mathbf{X} - \mathbf{M})^T \Sigma_N^{-1} (\mathbf{X} - \mathbf{M}) \\
 &\quad + \ln |\Sigma_N| - 2 \ln [P(\omega)], \tag{12}
 \end{aligned}$$

which has the same form as the quadratic discriminant function in (7) except for the covariance matrix. This result is due to the fact that the  $t$ -distribution approaches the normal distribution when the degree of freedom of the  $t$ -distribution approaches infinity.

The covariance matrix  $\Sigma_N$  given in (9) is a Bayes

estimate of the true covariance matrix. It is a weighted sum of an initial guess and the maximum likelihood estimate of the true covariance matrix:

$$\Sigma_N = (1 - \alpha) \hat{\Sigma} + \alpha \Sigma_0,$$

where

$$\begin{aligned}
 \hat{\Sigma} &= (1/N) \sum_{i=1}^N (\mathbf{X}_i - \mathbf{M})(\mathbf{X}_i - \mathbf{M})^T \\
 \alpha &= N_0 / (N + N_0). \tag{13}
 \end{aligned}$$

Usually we have very little knowledge about the initial guess of the covariance matrix. We can assume

$$\Sigma_0 = \sigma^2 I \tag{14}$$

without much loss of significance. With the assumption (14) and  $N_0 \ll N$ , we get a pseudo Bayes estimate

$$\Sigma_N = \hat{\Sigma} + \alpha \sigma^2 I = \hat{\Sigma} + h^2 I,$$

where

$$h^2 = \alpha \sigma^2 = [N_0 \sigma^2 / (N + N_0)]. \tag{15}$$

From (12) and (15),

$$\begin{aligned}
 g(\mathbf{X}) &= \sum_{i=1}^n \frac{[\Phi_i^T (\mathbf{X} - \mathbf{M})]^2}{\lambda_i + h^2} + \ln |\Sigma_N| - 2 \ln [P(\omega)], \tag{16}
 \end{aligned}$$

where  $\lambda_i$  and  $\Phi_i$  denote the  $i$ th eigenvalue and eigenvectors of the maximum likelihood estimate of the true covariance matrix.

It should be noted that the following equations were used to derive (16):

$$\begin{aligned}
 \Sigma_N \Phi_i &= (\hat{\Sigma} + h^2 I) \Phi_i = \lambda_i \Phi_i + h^2 \Phi_i = (\lambda_i + h^2) \Phi_i \\
 \Sigma_N &= \sum_{i=1}^n (\lambda_i + h^2) \Phi_i \Phi_i^T. \tag{17}
 \end{aligned}$$

Assuming  $h^2 \gg \lambda_i$  for  $i > k$ , (16) is approximated by

$$\begin{aligned}
 g(\mathbf{X}) &= \sum_{i=1}^k \frac{[\Phi_i (\mathbf{X} - \mathbf{M})]^2}{\lambda_i + h^2} + \sum_{i=k+1}^n \frac{[\Phi_i^T (\mathbf{X} - \mathbf{M})]^2}{h^2} \\
 &\quad + \ln |\Sigma_N| - 2 \ln [P(\omega)]. \tag{18}
 \end{aligned}$$

Using the relation

$$\begin{aligned}
 &\sum_{i=k+1}^n [\Phi_i^T (\mathbf{X} - \mathbf{M})]^2 \\
 &= |\mathbf{X} - \mathbf{M}|^2 - \sum_{i=1}^k [\Phi_i^T (\mathbf{X} - \mathbf{M})]^2, \tag{19}
 \end{aligned}$$

(18) is represented by

$$\begin{aligned}
 g(\mathbf{X}) &= \{ |\mathbf{X} - \mathbf{M}|^2 - \sum_{i=1}^k \frac{\lambda_i}{\lambda_i + h^2} [\Phi_i^T (\mathbf{X} - \mathbf{M})]^2 \} / h^2 \\
 &\quad + \ln [h^{2(n-k)} \prod_{i=1}^k (\lambda_i + h^2)] - 2 \ln [P(\omega)]. \tag{20}
 \end{aligned}$$

By assuming equal class a priori probably to neglect the last term of (20), and replacing the mean vector  $\mathbf{M}$  by the maximum likelihood estimate  $\hat{\mathbf{M}}$ , we can get

the discriminant function given in (1). The modified discriminant function has the following advantages when compared with the quadratic discriminant function given by (7):

(1) the MQDF is less sensitive to the estimation error in the covariance matrix than the QDF employing the maximum likelihood estimate and can achieve better performance;

(2) the computation time and storage required by the MQDF are considerably smaller than those of the QDF (about  $k/n = 12/64$  times);

(3) it is reasonably used in the case that a sufficient number of samples is not available (even in the case where the number of the training samples is smaller than the dimensionality); and

(4) the MQDF is suitable for implementation by hardware.

### 3.2. Structural classifier (S-algorithm)

In this section, we provide a brief description of the structural classifier. This classifier utilizes a group made up of Boolean expressions involving the primitives, for each numeral subclass. Groups are then combined to form a chain for each numeral. In the recognition procedure a tree structure is utilized to determine the specific group to which a given test numeral belongs. The algorithm also has provision for rejecting a test numeral or yielding multiple membership to the test numeral (example 1 or 7). The structural classifier is an extension of the procedure developed in an earlier phase by Shridhar.<sup>(6)</sup>

(1) *Determination of groups.* A group,  $Gr_i$ , is defined as a set of primitives characterizing a specific subclass of numeral  $i$ . The subclasses are determined by the similarity of their right and left profiles. As an example, a subclass of the numeral 5 can be represented by the following descriptors:

$$\left. \begin{array}{l} \text{RPEAK} > 10 \\ \text{LPEAK} > \end{array} \right\} \text{ for } 2 \leq R_2 \leq 50$$

$$\text{LMX} > \text{LMIN}; \text{ for } 10 \leq R_1 \leq 40.$$

Thus, the group in this case consists of the above descriptors as its elements.

A group table is constructed for each subclass and stored for use in recognition tests. The Appendix lists the Boolean expressions for all the groups in terms of the logical predicates.

(2) *Derivation of chains.* A chain,  $Ch(i)$ , is either a single group or a union of groups all identifying a specific numeral  $i$ . Thus, the chain for a numeral  $i$  is defined as

$$Ch(i) = \bigcup_{k=1}^K Gr_i^k, \quad (22)$$

where  $K$  is the number of subclasses for numeral  $i$  and  $Gr_i^k$  is the group defined for the  $k$ th subclass of numeral  $i$ .

For example, the chains for the numerals 0, 6 and 8 will be represented as a concatenation of groups as follows:

$$(1) Ch(0) = \bigcup_{k=1}^3 Gr_0^k,$$

where

$$Gr_0^1 = \bar{a}_1 \wedge \bar{b}_1 \wedge (\bar{b}_7 \wedge \bar{a}_{12}) \wedge a_8 \wedge (d_1 \vee d_2) \\ \wedge \bar{a}_4 \wedge \bar{a}_{16} \wedge g_4$$

$$Gr_0^2 = a_1 \wedge \bar{b}_1 \wedge \bar{h} \wedge (g_2 \wedge g_3) \wedge e_3 \wedge f_1$$

$$Gr_0^3 = a_1 \wedge b_1 \wedge \bar{a}_4 \wedge (\bar{a}_{12} \vee \bar{b}_8) \wedge e_3 \wedge f_1.$$

$$(2) Ch(6) = \bigcup_{k=1}^3 Gr_6^k,$$

where

$$Gr_6^1 = a_1 \wedge b_1 \wedge \bar{a}_3 \wedge (a_{13} \vee b_8) \wedge a_{14} \wedge \bar{b}_{10}$$

$$Gr_6^2 = \bar{a}_1 \wedge b_1 \wedge \bar{b}_3 \wedge (\bar{a}_{15} \vee b_9) \wedge \bar{a}_7 \wedge \bar{b}_7 \wedge \bar{a}_5 \\ \wedge b_5 \wedge \bar{b}_2$$

$$Gr_6^3 = a_1 \wedge \bar{b}_1 \wedge \bar{h} \wedge (\bar{g}_2 \vee \bar{g}_3) \wedge f_2 \wedge e_3 \wedge \bar{f}_1 \\ \wedge \bar{b}_4 \wedge \bar{a}_9 \wedge \bar{g}_1.$$

$$(3) Ch(8) = \bigcup_{k=1}^3 Gr_8^k,$$

where

$$Gr_8^1 = \bar{a}_1 \wedge \bar{b}_1 \wedge b_3 \wedge (a_{15} \wedge \bar{b}_9) \wedge a_6 \wedge c_3 \wedge \bar{e}_5$$

$$Gr_8^2 = a_1 \wedge \bar{b}_1 \wedge \bar{h} \wedge i \wedge \bar{e}_3$$

$$Gr_8^3 = a_1 \wedge b_1 \wedge \bar{a}_3 \wedge (\bar{a}_{13} \wedge \bar{b}_8) \wedge \bar{e}_3,$$

where ' $\vee$ ', ' $\wedge$ ', and ' $\bar{\phantom{x}}$ ' are the logical OR, logical AND and negation, respectively, and  $a, b, \dots$  are the primitives defined in the Appendix.

A verbal description of the group  $Gr_8^2$ , as an example, is as follows:

(1) the left and right profiles are relatively smooth, i.e. LPEAK and RPEAK are less than 10 in the range  $2 \leq R_2 \leq 50$ ;

(2) the ratio of height to maximum width is less than 2.5;

(3) the widths at the upper and lower halves are larger than in the middle;

(4) the left profile attains its minimum value before it attains its maximum value in the range 10 to 30, i.e.  $LMX > LMIN$  for  $10 \leq R_1 \leq 30$ .

The recognition algorithm utilizes a tree structure to identify the unknown numeral specimens. A feature is evaluated and compared with the feature description of numerals to determine the appropriate subgroups to which the unknown numeral belongs. Other features are then evaluated and the algorithm proceeds down the tree until an identity is established. The flow chart for identifying the numerals 0, 6 and 8 is shown in Fig. 6.

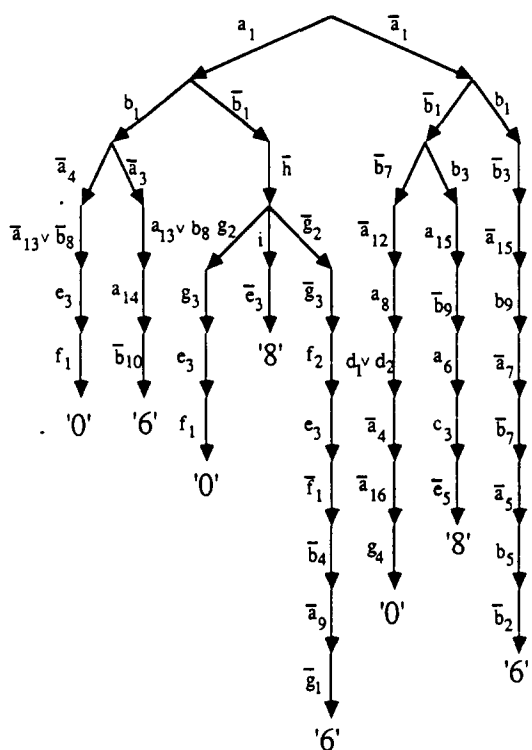


Fig. 6. Flow chart for identifying the numerals 0, 6 and 8.

4. TEST RESULTS

4.1. K-algorithm

The K-algorithm has two thresholds RJ1 and RJ2 to control error-rejection rate. If the minimum value of (1) achieved by a numeral class is greater than the RJ1, the sample is rejected. Also, if the difference between the minimum value and the second minimum value of (1) is less than RJ2, the sample is rejected with a pair of numeral classes (multiple membership) achieving these two minimum values.

The performance of K-algorithm was evaluated for various combinations of the RJ1 and RJ2. To calculate the mean vector and the eigen properties necessary for discriminant function calculation, 300 samples were used as design samples for each numeral class. A total of 17,394 test samples were used to evaluate the performance of the recognition algorithms. The results are given in Table 1. This table shows, for example, the minimum error rate

0.0% with minimum reject rate 34.16% achieved for RJ1 = 0 and RJ2 = 50. The maximum correct recognition rate 97.58% is achieved for RJ1 = 100 and RJ2 = 0 with 2.05% error and 0.37% rejection (not in Table 1).

The total processing time of the K-algorithm was 300 ms/numeral on a 25 MHz 80386 IBM compatible computer. The processing time for the feature extraction (including preprocessing) and the classification were 200 ms/numeral and 100 ms/numeral, respectively.

4.2. S-algorithm

The tests samples were applied to the structural classifier in a two-pass procedure with the peak thresholds adjusted for each pass. The results obtained in each pass were combined logically as follows, to determine the final decision:

- (1) accept identity if both passes yield the same result;
- (2) reject sample if both passes reject the sample;
- (3) determine a subclass (multiple membership) if the two passes yield different results (e.g. 1 or 7);
- (4) determine a fuzzy class if one pass yields a recognition while the other pass rejects the sample (e.g. "may be 1").

The results obtained with the structural classifier are shown in Table 2. The error rate was 0.17% with 3.22% rejection and 19.21% multiple membership (e.g. 7 or 2). The total processing time of the S-algorithm was 125 ms/numeral.

5. IMPROVEMENT IN PERFORMANCE

We consider two parallel and four sequential combinations of the two recognition algorithms. In the parallel combinations, both recognition algorithms are applied to the input numeral and the results are combined in the following way to decide the numeral class of the input numeral.

Table 2. Error and rejection rates (%) of S-algorithm

Error	Rejection	Multiple membership	Correct
0.17	3.22	19.21	77.40

Table 1. Error-rejection rates (%) of K-algorithm

RJ1\RJ2	10	20	30	40	50
0	0.09/18.55	0.03/19.66	0.01/22.28	0.01/26.78	0.00/34.16
20	0.30/8.29	0.14/10.16	0.07/13.77	0.04/19.33	0.02/27.92
40	0.51/4.81	0.24/7.15	0.12/11.27	0.06/17.25	0.02/26.30
60	0.72/3.60	0.34/6.16	0.18/10.44	0.09/16.62	0.04/25.85
80	0.89/3.09	0.45/5.78	0.25/10.16	0.11/16.44	0.05/25.72



Table 3. Combination matrix for Parallel-1 combination

S\K	Rejection	Error	Correct
Rejection	R	E	C
Error	E	R or E	R
Correct	C	R	C

R: rejection; E: error; C: correct; S: S-algorithm; K: K-algorithm

(1) *Parallel-1*

If the input numeral is rejected by both algorithms, the numeral is rejected. If both algorithms accept the input numeral with the same result, or if only one algorithm yields a rejection, the numeral class is accepted. If both algorithms accept the input numeral with different results, the numeral is rejected. The method of Parallel-1 combination is represented in a matrix form as shown in Table 3. In this table, E and C correspond to the acceptance.

(2) *Parallel-2*

When one algorithm yields multiple membership, it is exploited in the following way. If the other algorithm accepts the input numeral with an identity, the identity is accepted if it is included in the multiple membership, and rejected otherwise. If the other algorithm yields multiple membership or rejects the input numeral, the numeral is rejected. In all other cases irrelevant to the multiple membership, the results are combined in the same way as in the Parallel-1. Table 4 shows the combination matrix for the Parallel-2 combination. In sequential combinations, the second recognition algorithm is applied only when the first algorithm rejects an input numeral. There are four sequential combinations.

(3) *Sequential-1*

The S-algorithm is applied first. The K-algorithm is applied only to those numerals rejected by the S-algorithm. The multiple membership yielded by either algorithm is not exploited. The combination matrix for Sequential-1 combination is given in Table 5.

Table 5. Combination matrix for Sequential-1 combination

S\K	Rejection	Error	Correct
Rejection	R	E	C
Error	E	E	E
Correct	C	C	C

R: rejection; E: error; C: correct; S: S-algorithm; K: K-algorithm

(4) *Sequential-2*

The S-algorithm is applied first. The K-algorithm is applied only to those numerals rejected by the S-algorithm. The multiple membership yielded by the S-algorithm is exploited in the same way as in the Parallel-2 combination. The combination matrix for the Sequential-2 combination is given in the Table 6.

(5) *Sequential-3*

The K-algorithm is applied first. The S-algorithm is applied only to those numerals rejected by the K-algorithm. The multiple membership yielded by either algorithm is not exploited. The combination matrix for the Sequential-3 combination is easily obtained from the matrix for Sequential-1 combination by changing the role of the rows and columns.

(6) *Sequential-4*

The K-algorithm is applied first. The S-algorithm is applied only to those numerals rejected by the K-algorithm. The multiple membership yielded by the K-algorithm is exploited in the same way as in the Parallel-2 combination. The combination matrix for the Sequential-4 combination is easily obtained from the matrix for the Sequential-2 combination by changing the role of the rows and columns.

The performance of combined algorithms was evaluated for various rejection thresholds RJ1 and RJ2. The results are given in Tables 7–12. Figure 7 shows the error–rejection relationships obtained by changing the rejection thresholds for the Parallel-1 combination. Each curve corresponds to each row

Table 4. Combination matrix for Parallel-2 combination

S\K	Rejection	Multiple mem.	Error	Correct
Rejection	R	R	E	C
Multiple mem.	R	R	R or E	R or C
Error	E	R or E	R or E	R
Correct	C	R or C	R	C

R: rejection; E: error; C: correct; S: S-algorithm; K: K-algorithm

Table 6. Combination matrix for Sequential-2 combination

<i>SVK</i>	Rejection	Multiple Mem.	Error	Correct
Rejection	R	R	E	C
Multiple Mem.	R	R	R or E	R or C
Error	E	E	E	E
Correct	C	C	C	C

R: rejection; E: error; C: correct; S: S-algorithm; K: K-algorithm

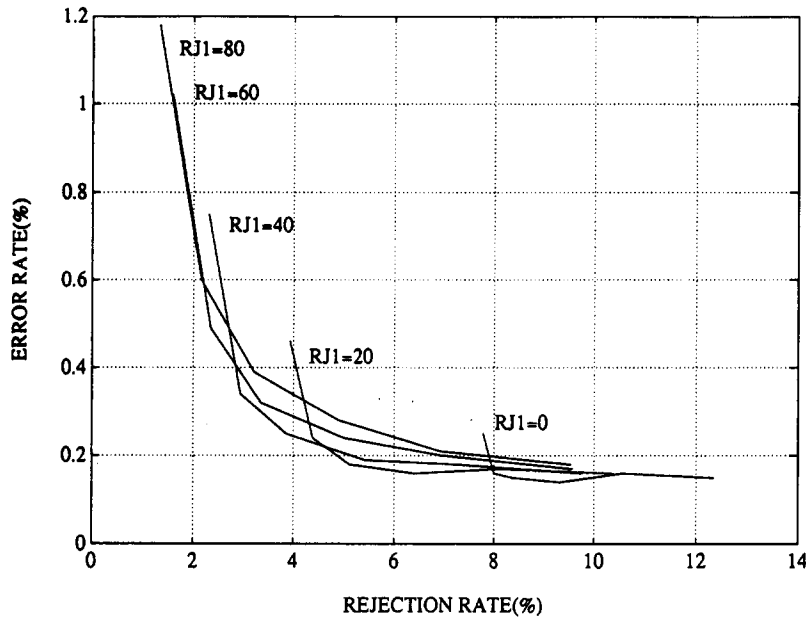


Fig. 7. Error-rejection relationships for the Parallel-1 combination.

of Table 7. The overall error-rejection relation for the Parallel-1 combination is obtained from these curves as the envelope connecting the lower convex points. These convex points are underscored in Table 7. Figure 8 shows the error-rejection relationship thus obtained for each combined algorithm and the K-algorithm.

In order to reduce the error rate the Parallel-2 combination is the best because it always gives the least error rate in the low error region, which is defined to be the region with the error rate less than 0.3%. Typical error and rejection rates of this combination are 0.07% error with 10.38% rejection, 0.13% error with 4.79% rejection, and 0.25% error with 3.52% rejection. In the region with the higher error rate (greater than 0.3%), the Sequential-1 or the Parallel-1 combination achieve less error rate than the Parallel-2 combination; however, the following considerations are limited to the low error region since our main concern is in the low error region.

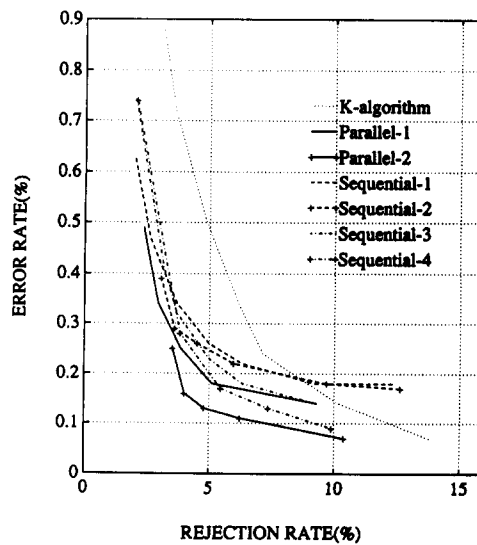


Fig. 8. Error-rejection relationships for each combined algorithm and the K-algorithm.

Table 7. Error-rejection rates (%) of Parallel-1 combination

RJ1\RJ2	10	20	30	40	50
0	0.16/7.99	0.15/8.37	<u>0.14/9.31</u>	0.16/10.55	0.15/12.34
20	0.24/4.38	<u>0.18/5.11</u>	<u>0.16/6.39</u>	0.17/8.08	0.16/10.37
40	<u>0.34/2.94</u>	<u>0.25/3.85</u>	0.19/5.41	0.18/7.24	0.16/9.75
60	<u>0.49/2.35</u>	0.32/3.35	0.24/5.00	0.20/6.96	0.17/9.56
80	<u>0.60/2.16</u>	0.39/3.21	0.28/4.90	0.21/6.92	0.18/9.53

Table 8. Error-rejection rates (%) of Parallel-2 combination

RJ1\RJ2	10	20	30	40	50
0	0.15/8.51	0.14/8.86	0.13/9.75	0.13/10.95	0.13/12.67
20	0.14/5.19	0.13/5.83	0.12/7.05	0.12/8.66	0.10/10.87
40	0.16/4.00	0.13/4.79	0.11/6.23	0.10/7.97	0.07/10.38
60	0.25/3.52	0.18/4.39	0.13/5.89	0.10/7.76	0.08/10.24
80	0.33/3.38	0.22/4.30	0.16/5.85	0.10/7.77	0.08/10.27

Table 9. Error-rejection rates (%) of Sequential-1 combination

RJ1\RJ2	10	20	30	40	50
0	0.22/7.89	0.20/8.31	0.18/9.26	0.18/10.52	0.18/12.31
20	0.34/4.15	0.26/4.98	0.22/6.32	0.21/8.03	0.19/10.34
40	0.47/2.61	0.34/3.69	0.25/5.31	0.22/7.19	0.20/9.71
60	0.63/1.97	0.42/3.17	0.30/4.89	0.25/6.90	0.21/9.52
80	0.74/1.72	0.49/2.99	0.34/4.77	0.25/6.85	0.22/9.48

Table 10. Error-rejection rates (%) of Sequential-2 combination

RJ1\RJ2	10	20	30	40	50
0	0.21/8.41	0.20/8.80	0.18/9.69	0.18/10.89	0.17/12.62
20	0.25/4.93	0.23/5.65	0.21/6.90	0.20/8.53	0.18/10.75
40	0.29/3.60	0.26/4.51	0.22/5.98	0.21/7.74	0.18/10.16
60	0.39/3.05	0.31/4.05	0.25/5.61	0.21/7.49	0.19/9.99
80	0.47/2.84	0.36/3.90	0.28/5.50	0.21/7.45	0.19/9.96

Table 11. Error-rejection rates (%) of Sequential-3 combination

RJ1\RJ2	10	20	30	40	50
0	0.21/7.89	0.16/8.31	0.14/9.26	0.16/10.52	0.15/12.31
20	0.36/4.15	0.23/4.98	0.18/6.32	0.17/8.03	0.16/10.34
40	0.55/2.61	0.32/3.69	0.22/5.31	0.19/7.19	0.16/9.71
60	0.75/1.97	0.41/3.17	0.28/4.89	0.21/6.90	0.17/9.52
80	0.92/1.72	0.52/2.99	0.35/4.77	0.24/6.85	0.18/9.48

Table 12. Error-rejection rates (%) of Sequential-4 combination

RJ1\RJ2	10	20	30	40	50
0	0.21/7.89	0.16/8.32	0.13/9.27	0.14/10.54	0.13/12.34
20	0.36/4.17	0.21/5.03	0.15/6.39	0.13/8.11	0.11/10.42
40	0.54/2.67	0.28/3.82	0.17/5.46	0.13/7.36	0.09/9.89
60	0.74/2.05	0.37/3.33	0.23/5.08	0.15/7.11	0.10/9.73
80	0.91/1.82	0.49/3.17	0.30/4.98	0.17/7.09	0.11/9.74

The exploitation of the multiple membership is effective in reducing the error rate in the low error region except for the Sequential-2 combination. The Parallel-2 and the Sequential-4 combinations exploiting the multiple membership give better results than corresponding Parallel-1 and Sequential-3 combinations, respectively, which do not exploit the multiple membership. The error rate of the Sequential-1 combination is not improved noticeably by the exploitation of the multiple membership because of the 0.17% fixed error rate of the first algorithm (*S*-algorithm), which cannot be reduced by the multiple membership exploitation.

For the sequential combinations, applying the recognition algorithm with lower error rate first and the higher one later is better than applying them in the reverse order. In the low error region, the Sequential-3 and the Sequential-4 combinations applying the recognition algorithm with lower error rate first achieve lower error rates than the Sequential-1 and the Sequential-2 combinations. In the higher error region, the Sequential-1 combination, for example, achieves lower error rate than the Sequential-3 and the Sequential-4 combinations because in this region the first algorithm (*K*-algorithm) yields more error than the second algorithm.

The performance of the sequential combinations approaches the performance of the parallel combinations when (and only when) the error rate of the first algorithm approaches zero. The performance of the Sequential-3 and the Sequential-4 combinations approaches the performance of the Parallel-1 and the Parallel-2 combinations, respectively, when the rejection thresholds of the *K*-algorithm are set so that the error rate of the first algorithm is nearly equal to zero (0.02% or less). In this case, there is no need to rely on the second algorithm to verify the result once the samples are accepted by the first algorithm. This fact suggests the effectiveness of the sequential combination of multiple (more than two) recognition algorithms each of which except the final one is tuned to achieve almost no error. The total error-rejection relationship for this sequential combination is controlled by adjusting the final algorithm.

The Parallel-2 and Sequential-4 combinations give less error than *K*-algorithm for any rejection rate (their error-rejection curves are always under the curve for the *K*-algorithm). This result shows the effectiveness of the exploitation of the multiple membership, parallel combination strategy, and the importance of the first recognition algorithm with lower error rate in the sequential combination strategy.

The best performance (minimum error) for each combination is achieved when the rejection parameters of the *K*-algorithm are selected so that the classifier achieves a very low error rate. This result shows that the flexible error-rejection adjustability is very important in deriving an effective combined

recognition algorithm. The primary requirement for each classifier is not a reduction in both error and rejection rates but a reduction in the error rate by adjusting the rejection rate.

## 6. DISCUSSION AND CONCLUSION

In this paper, the authors developed two algorithms for application to recognition of unconstrained isolated handwritten numerals. While both algorithms yielded very low error rates, the authors combined the two algorithms in different ways to study the best polling strategy and realized significant improvement in performance. The results of the experiments are summarized as follows.

(1) The best performance was achieved by the Parallel-2 combination, which realized very low error rates (0.2% or less) and rejection rates below 4%.

(2) The parallel combinations usually achieve less error than the corresponding sequential combination.

(3) The exploitation of the multiple membership is effective in reducing the error rate.

(4) For the sequential combinations, applying the recognition algorithm with lower error rate first and the higher one later is better than applying them in the reverse order.

(5) The performance of the sequential combinations approaches the performance of the parallel combinations when the error rate of the first algorithm approaches zero.

Although the recognition rates of handwritten numeral recognition are highly dependent on the test data, and it is difficult to compare the results with other works, the authors believe that the result obtained in this paper is one of the best results reported so far. The authors used a fairly large test database to obtain a statistically meaningful performance evaluation. For example, one of the authors also developed a numeral recognition technique based on multiple recognition algorithms including the relaxation matching technique<sup>(9)</sup> instead of the *K*-algorithm. The result for the same test data was 0.6% error with 3.1% rejection, which is worse than the result for any combination strategy discussed in this paper. Also, the authors have applied their algorithm to recognition of digit samples obtained from USPS Zip codes. The Zip codes contained both isolated and connected digits. The accuracies obtained with isolated Zip code digits were comparable to those reported here. Segmentation and recognition of connected numerals is currently under investigation. The results of this study will be reported at a future time.

From the standpoint of computation time, the Sequential-1 and the Sequential-2 combinations are better than other sequential and parallel algorithms. However, the increased error rate that results is not acceptable for many practical applications. By a

proper choice of hardware and a careful design of the software for implementing the recognition algorithms, an accepted speed can be realized.

Further improvements on the performance can be achieved by reducing the error rate of the S-algorithm. The authors are currently studying the improvement of the S-algorithm, including the readjustment of the error-rejection rate of the algorithm. Combinations of more than two algorithms also appear to be effective. Combinations of the statistical classifiers utilizing different features are also easily implemented and are promising due to the error-rejection adjustability. All these are topics of further study.

REFERENCES

1. P. G. Guebey, Handwritten address recognition, *Proc. USPS advd Technol. Conf.*, pp. 371-375 (October 1986).
2. J. S. Huang and K. Chuang, Heuristic approach to handwritten numeral recognition, *Pattern Recognition* **19**, 15-19 (1986).
3. B. T. Mitchell and A. M. Gillies, Advanced research in recognizing handwritten ZIP codes, *Proc. USPS advd Technol. Conf.*, pp. 813-827 (May 1988).
4. P. Ahmed and C. Y. Suen, Computer recognition of totally unconstrained handwritten Zip codes, *Int. J. Pattern Recognition Artif. Intell.* **1**, 1-15 (1987).
5. M. Shridhar and A. Badreldin, Recognition of isolated and simply connected handwritten numerals, *Pattern Recognition* **19**, 1-12 (1986).
6. M. Shridhar and A. Badreldin, A high-accuracy syntactic recognition algorithm for handwritten numerals, *IEEE Trans. Syst. Man Cybern.* **15**, 152-158 (1985).
7. F. Kimura, K. Takashina, S. Tsuruoka and Y. Miyake, Modified quadratic discriminant functions and the application to Chinese character recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* **9**, 149-153 (1987).
8. F. Kimura, S. Tsuruoka and Y. Miyake, On avoiding peaking phenomenon of the quadratic discriminant function, *8th Int. Conf. Pattern Recognition*, Paris, pp. 275-278 (1986).
9. L. Lam and C. Y. Suen, Structural classification and relaxation matching of totally unconstrained handwritten Zip-code numbers, *Pattern Recognition* **21**, 19-31 (1988).
10. C. Y. Suen, P. Ahmed and L. Lam, Recognition of totally unconstrained handwritten Zip codes, *Proc. USPS advd Technol. Conf.*, pp. 855-866 (May 1988).
11. S. N. Srihari, E. Cohen, J. J. Hull and L. Kuan, A system to locate and recognize ZIP codes in handwritten addresses, *IJRE* **1**, 37-45 (1989).
12. J. J. Hull, S. N. Srihari, E. Cohen, C. L. Kuan, P. Cullen and P. Palumbo, A blackboard-based approach to handwritten ZIP code recognition, *Proc. USPS advd Technol. Conf.*, pp. 1018-1032 (May 1988).
13. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, p. 51. Wesley, New York (1973).
14. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, p. 219. Academic Press, New York (1972).

APPENDIX  
GROUP TABLE FOR EACH SUBCLASS

Table A1. Groups of the numerals 0 to 9

Subclass	Boolean Expression
Gr <sub>0</sub> <sup>1</sup>	$\bar{a}_1 \wedge \bar{b}_1 \wedge (\bar{b}_7 \wedge \bar{a}_{12}) \wedge a_8 \wedge (d_1 \vee d_2) \wedge \bar{a}_4 \wedge \bar{a}_{16} \wedge g_4$
Gr <sub>0</sub> <sup>2</sup>	$a_1 \wedge b_1 \wedge h \wedge (g_2 \wedge g_3) \wedge e_3 \wedge f_1$
Gr <sub>0</sub> <sup>3</sup>	$a_1 \wedge b_1 \wedge \bar{a}_4 \wedge (\bar{a}_{13} \vee b_8) \wedge e_3 \wedge f_1$
Gr <sub>1</sub> <sup>1</sup>	$a_1 \wedge b_1 \wedge h$
Gr <sub>1</sub> <sup>2</sup>	$\bar{a}_1 \wedge \bar{b}_1 \wedge (\bar{b}_7 \wedge \bar{a}_{12}) \wedge a_8 \wedge (d_1 \vee d_2) \wedge \bar{a}_4 \wedge \bar{a}_{16} \wedge \bar{a}_{11} \wedge \bar{g}_4$
Gr <sub>1</sub> <sup>3</sup>	$a_1 \wedge b_1 \wedge h \wedge (\bar{g}_2 \wedge \bar{g}_3) \wedge \bar{f}_2 \wedge a_2 \wedge (a_{17} \wedge b_{11}) \wedge \bar{a}_8$
Gr <sub>1</sub> <sup>4</sup>	$\bar{a}_1 \wedge b_1 \wedge b_3 \wedge a_5 \wedge (b_9 \vee \bar{a}_{15}) \wedge f_3$
Gr <sub>1</sub> <sup>5</sup>	$a_1 \wedge b_1 \wedge \bar{a}_7 \wedge (a_{13} \vee b_8) \wedge a_{14} \wedge b_{10} \wedge \bar{e}_2$
Gr <sub>1</sub> <sup>6</sup>	$\bar{a}_1 \wedge b_1 \wedge b_3 \wedge \bar{a}_5 \wedge \bar{a}_{10} \wedge (a_{13} \vee b_8) \wedge \bar{e}_2$
Gr <sub>1</sub> <sup>7</sup>	$\bar{a}_1 \wedge b_1 \wedge (\bar{b}_7 \wedge \bar{a}_{12}) \wedge a_8 \wedge (d_1 \vee d_2) \wedge a_{13} \wedge f_3$
Gr <sub>1</sub> <sup>8</sup>	$a_1 \wedge b_1 \wedge (\bar{b}_7 \vee \bar{a}_{12}) \wedge a_8 \wedge (d_1 \vee d_2) \wedge a_4 \wedge a_{16} \vee c_2 \vee f_3$
Gr <sub>1</sub> <sup>9</sup>	$\bar{a}_1 \wedge b_1 \wedge (\bar{b}_7 \vee \bar{a}_{12}) \wedge a_8 \wedge (d_1 \vee d_2) \wedge a_4 \wedge a_{13} \wedge \bar{f}_3$
Gr <sub>1</sub> <sup>10</sup>	$\bar{a}_1 \wedge b_1 \wedge (\bar{a}_{12} \vee b_7) \wedge \bar{a}_8$
Gr <sub>2</sub> <sup>1</sup>	$\bar{a}_1 \wedge b_1 \wedge b_3 \wedge (b_9 \vee \bar{a}_{15}) \wedge \bar{a}_7 \wedge \bar{b}_7 \wedge \bar{a}_5 \wedge \bar{b}_5$
Gr <sub>2</sub> <sup>2</sup>	$\bar{a}_1 \wedge b_1 \wedge b_3 \wedge (b_9 \vee \bar{a}_{15}) \wedge \bar{a}_7 \wedge b_7$
Gr <sub>2</sub> <sup>3</sup>	$\bar{a}_1 \wedge b_1 \wedge b_3 \wedge (b_9 \vee \bar{a}_{15}) \wedge a_7 \wedge a_{13} \wedge e_1 \wedge e_6$
Gr <sub>2</sub> <sup>4</sup>	$\bar{a}_1 \wedge b_1 \wedge e_4 \wedge b_9 \wedge f_3 \wedge b_6 \wedge c_1$
Gr <sub>2</sub> <sup>5</sup>	$a_1 \wedge b_1 \wedge \bar{a}_3 \wedge (a_{13} \vee b_8) \wedge a_{14} \wedge \bar{b}_{10}$
Gr <sub>2</sub> <sup>6</sup>	$\bar{a}_1 \wedge b_1 \wedge b_3 \wedge (\bar{a}_{15} \vee b_9) \wedge \bar{a}_7 \wedge b_7 \wedge \bar{a}_5 \wedge b_5 \wedge \bar{b}_2$
Gr <sub>2</sub> <sup>7</sup>	$a_1 \wedge b_1 \wedge h \wedge (\bar{g}_2 \vee \bar{g}_3) \wedge \bar{f}_2 \wedge e_3 \wedge f_1 \wedge b_4 \wedge \bar{a}_9 \wedge g_1$
Gr <sub>2</sub> <sup>8</sup>	$\bar{a}_1 \wedge b_1 \wedge (a_{12} \wedge b_7) \wedge \bar{a}_4$
Gr <sub>2</sub> <sup>9</sup>	$\bar{a} \wedge b_1 \wedge b_3 \wedge (a_{15} \wedge b_9) \wedge a_6 \wedge \bar{c}_3$
Gr <sub>2</sub> <sup>10</sup>	$\bar{a}_1 \wedge b_1 \wedge b_3 \wedge (a_{15} \wedge b_9) \wedge a_6 \wedge c_3 \wedge \bar{e}_5$
Gr <sub>2</sub> <sup>11</sup>	$a_1 \wedge b_1 \wedge h \wedge i \wedge \bar{e}_3$
Gr <sub>2</sub> <sup>12</sup>	$a_1 \wedge b_1 \wedge \bar{a}_3 \wedge (\bar{a}_{13} \wedge \bar{b}_8) \wedge \bar{e}_3$
Gr <sub>2</sub> <sup>13</sup>	$\bar{a}_1 \wedge b_1 \wedge (a_{12} \wedge b_7) \wedge \bar{a}_4$
Gr <sub>2</sub> <sup>14</sup>	$\bar{a}_1 \wedge b_1 \wedge (\bar{a}_{12} \vee b_7) \wedge a_8 \wedge (d_1 \vee d_2) \wedge \bar{a}_4 \wedge a_{16} \wedge \bar{c}_2$

Table A2. Predicates used in the recognition of the numerals 0,1, . . . ,9

Number	Primitive	Semantic rule (predicate)
1	$a_1$	True if $LPEAK < 10; 2 \leq R_2 \leq 50$
2	$a_2$	True if $LPEAK < 5; 2 \leq R_2 \leq 10$
3	$a_3$	True if $LPEAK > 5; 2 \leq R_2 \leq 15$
4	$a_4$	True if $LPEAK > 10; 2 \leq R_2 \leq 15$
5	$a_5$	True if $LPEAK > 10; 2 \leq R_2 \leq 20$
6	$a_6$	True if $LPEAK > 5; 2 \leq R_2 \leq 25$
7	$a_7$	True if $LPEAK > 5; 5 \leq R_2 \leq 15$
8	$a_8$	True if $LPEAK > 5; 5 \leq R_2 \leq 35$
9	$a_9$	True if $LPEAK > 10; 5 \leq R_2 \leq 40$
10	$a_{10}$	True if $LPEAK > 10; 10 \leq R_2 \leq 30$
11	$a_{11}$	True if $LPEAK > 10; 15 \leq R_2 \leq 40$
12	$a_{12}$	True if $LPEAK < 5; 25 \leq R_2 \leq 50$
13	$a_{13}$	True if $LPEAK > 10; 30 \leq R_2 \leq 50$
14	$a_{14}$	True if $LPEAK < 5; 30 \leq R_2 \leq 50$
15	$a_{15}$	True if $LPEAK < 5; 35 \leq R_2 \leq 50$
16	$a_{16}$	True if $LPEAK > 10; 35 \leq R_2 \leq 50$
17	$a_{17}$	True if $LPEAK > 5; 40 \leq R_2 \leq 50$
18	$b_1$	True if $RPEAK > 10; 2 \leq R_2 \leq 50$
19	$b_2$	True if $RPEAK > 10; 2 \leq R_2 \leq 15$
20	$b_3$	True if $RPEAK < 10; 2 \leq R_2 \leq 30$
21	$b_4$	True if $RPEAK < 5; 2 \leq R_2 \leq 45$
22	$b_5$	True if $RPEAK < 10; 25 \leq R_2 \leq 45$
23	$b_6$	True if $RPEAK > 10; 25 \leq R_2 \leq 50$
24	$b_7$	True if $RPEAK < 5; 25 \leq R_2 \leq 50$
25	$b_8$	True if $RPEAK > 10; 30 \leq R_2 \leq 50$
26	$b_9$	True if $RPEAK > 5; 35 \leq R_2 \leq 50$
27	$b_{10}$	True if $RPEAK > 10; 35 \leq R_2 \leq 50$
28	$b_{11}$	True if $RPEAK > 5; 40 \leq R_2 \leq 50$
29	$c_1$	True if $RMIN(1 \leq R_1 \leq 30)$ is less than $RMX2(RMIN \leq R_1 \leq 30)$ and greater than $RMX1(1 \leq R_1 \leq RMIN)$
30	$c_2$	True if $RMIN(10 \leq R_1 \leq 40)$ is less than $RMX2(RMIN \leq R_1 \leq 40)$ and greater than $RMX1(1 \leq R_1 \leq RMIN)$
31	$c_3$	True if $RMIN(10 \leq R_1 \leq 45)$ is less than $RMX2(RMIN \leq R_1 \leq 45)$ and greater than $RMX1(1 \leq R_1 \leq RMIN)$
32	$d_1$	True if $RP(RMIN) = RP(RMX)$ where $RMIN$ is in the range $5 \leq R_1 \leq 25$ , and $RMX$ in the range $1 \leq R_1 \leq RMIN$
33	$d_2$	True if $RP(RMIN) = RP(RMX)$ where $RMIN$ is in the range $5 \leq R_1 \leq 25$ , and $RMX$ is in the range $RMIN \leq R_1 \leq 40$
34	$e_1$	True if $LMX < LMIN$ ; where $LMX$ and $LMIN$ are in the range $1 \leq R_1 \leq 10$
35	$e_2$	True if $LMX < LMIN$ ; where $LMX$ in the range $1 \leq R_1 \leq 30$ , and $LMIN$ in the range $1 \leq R_1 \leq LMX$
36	$e_3$	True if $LMX < LMIN$ ; where $LMX$ in the range $10 \leq R_1 \leq 40$ , and $LMIN$ in the range $10 \leq R_1 \leq LMX$
37	$e_4$	True if $LMX < LMIN$ ; where $LMX$ and $LMIN$ are in the range $15 \leq R_1 \leq 45$
38	$e_5$	True if $LMX < LMIN$ ; where $LMX$ and $LMIN$ are in the range $20 \leq R_1 \leq 50$
39	$e_6$	True if $LMX < LMIN$ ; where $LMX$ and $LMIN$ are in the range $40 \leq R_1 \leq 50$
40	$f_1$	True if $RMIN < RMX$ ; where $RMIN$ is in the range $1 \leq R_1 \leq 30$ , and $RMX$ in the range $1 \leq R_1 \leq RMIN$
41	$f_2$	True if $RMIN < RMX$ ; where $RMIN$ and $RMX$ are in the range $20 \leq R_1 \leq 35$
42	$f_3$	True if $RMIN < RMX$ ; where $RMIN$ and $RMX$ are in the range $35 \leq R_1 \leq 50$
43	$g_1$	True if $W(20) \geq W(40)$
44	$g_2$	True if $W(25) \geq W(10)$
45	$g_3$	True if $W(25) \geq W(40)$
46	$g_4$	True if $W(25) \geq W(45)$
47	$h$	True if $Ratio > 2.5$
48	$i$	True if $WMIN < WMX1$ and $WMIN < WMX2$ ; where $WMIN = \min_{10 < J < 40} \{W(J)\} = \{W(LMN)\}$ $WMX1 = \max_{1 < J < LMN} \{W(J)\}$ $WMX2 = \max_{LMN < J < 50} \{W(J)\}$

**About the Author**—Dr F. KIMURA obtained his M.S. in Electrical Engineering from Nagoya University, Japan, in 1975 and Ph.D. in Information Engineering from Nagoya University in 1981. From 1978 to 1982 he was a faculty member in the Information Engineering Department of Nagoya University. During this period, his research interests were in the areas of character recognition and image processing. Since 1982, he has been with Mie University, where he is an associate professor in the department of Electrical Engineering. He is currently with the Department of Electrical and Computer Engineering at the University of Michigan–Dearborn, on leave from Mie University. He has performed several research projects on character recognition and object recognition supported by the Ministry of Education, Science and Culture of Japan. In addition he has consulted for several manufacturers developing OCRs.

**About the Author**—Dr M. SHRIDHAR obtained his M.S. in Electrical Engineering from Polytechnic Institute of Brooklyn, New York, in 1968 and Ph.D. in Electrical Engineering from the University of Aston in Birmingham, U.K., in 1970. From 1969 to 1985 he was a faculty member in the Electrical Engineering Department of the University of Windsor, Canada. During this period, his research interests were in the areas of speech and image processing and pattern recognition. Since 1986, he has been with the University of Michigan–Dearborn, where he is currently a professor and chairman of the department of Electrical and Computer Engineering. Dr Shridhar has been very active in the research area of speech processing, speaker and isolated word recognition. In the 1980s he focused on research in the area of image processing, vision and pattern recognition. He has been involved in contract research with local industries in addition to funded research from NSERC, Canada.