LDA-Based Compound Distance for Handwritten Chinese Character Recognition

Tian-Fu Gao, Cheng-Lin Liu
National Laboratory of Pattern Recognition (NLPR)
Institute of Automation, Chinese Academy of Sciences
PO Box 2728, Beijing 100080, P.R. China
E-mail: {tfgao, liucl}@nlpr.ia.ac.cn

Abstract

In this paper, we propose a linear discriminant analysis (LDA)-based compound distance measure for discriminating similar characters in handwritten Chinese character recognition. The previous compound Mahalanobis function (CMF) is shown to be a special case of the proposed method. On finding similar character pairs by cross-validation using a baseline classifier, LDA is applied to each similar pair, and the LDA-based distance measure is combined with the discriminant function of the baseline classifier. In our experiments on the ETL9B and CASIA databases using the modified quadratic discriminant function (MQDF) as baseline classifier, the LDA-based compound distance is demonstrated to outperform the previous compound function methods.

1. Introduction

In recent years, great improvements have been achieved in the field of handwritten Chinese character recognition (HCCR), via improving the methods of image preprocessing, feature extraction, classification, or post-processing. Despite the high accuracies achieved so far, some problems still remain unsolved. One of the problems is the discrimination of similar characters. Similar Chinese characters often share common radicals and have only subtle shape difference in local details. The variability of writing styles also adds to the difficulty of recognition.

Many efforts have been devoted to the discrimination of similar characters in HCCR for improving the overall recognition accuracy. This is generally done by training classifiers for small subsets of characters, which are decided by a baseline all-class classifier. In the simplest case, the subset classifier discriminates only two classes. Using pair discriminators to reorder the candidate classes given by a baseline classifier has shown success [1][2]. The problem with this scheme, however, is the huge number of pair discriminators. For an M-class problem, the total number of pairs is $M(M-1)/2$. This is a number of millions for thousands of classes. Even though pairs of characters can be well separated by linear classifiers [3], the storage for all pairs is costly. A heuristic to reduce the number of similar pairs is the co-occurrence frequency in top-rank candidate classes given by the baseline classifier.

The compound Mahalanobis function (CMF) method, proposed by Suzuki et al. [4], combines pair discrimination measures with class-wise Mahalanobis distance. This method is unique in that the pair discriminator has no extra parameters, since it is derived from the Gaussian density parameters of the baseline Mahalanobis distance classifier. This method has been extended to other types of quadratic classifiers (we call this family of methods compound function or compound distance methods) [5][6][7], including the modified projection distance [5] and the modified quadratic discriminant function (MQDF) [8].

Despite that the CMF applies to all pairs of classes without storing extra parameters and has yielded improved accuracies, the class pairs are not optimally separated. In this paper, we propose a new compound function method, which uses linear discriminant analysis (LDA) to discriminate similar character pairs. We show that under restrictive assumptions, the previous CMF is a special case of our LDA-based compound distance method. Our experiments on two large databases, ETL9B and CASIA, using the MQDF as baseline classifier, demonstrate that the proposed LDA-based compound distance method outperforms the previous compound function methods.

In the rest of this paper, we briefly review the MQDF and CMF methods in Section 2. Section 3 describes the proposed LDA-based compound distance method. Section 4 explains the character recognition scheme; Section 5 presents our experimental results and Section 6 offers concluding remarks.

2. MQDF and CMF

The MQDF of Kimura et al. [8] has been widely applied to handwritten Japanese and Chinese character recognition with great success. We take it as a baseline classifier and use compound functions to further improve the accuracy.

Representing the input character pattern by a *d*-dimensional feature vector $\mathbf{x} = (x_{1,} x_2, x_3, \cdots, x_d)^T$, the MQDF for each class ω_i (i=1,2…,M) is computed by

$$f(\mathbf{x},\omega_i)=\sum_{j=1}^{k}\frac{1}{\lambda_{ij}}[\boldsymbol{\varphi}_{ij}^T(\mathbf{x}-\boldsymbol{\mu}_i)]^2+\frac{1}{\delta}\Big(\|\mathbf{x}-\boldsymbol{\mu}_i\|^2-\sum_{j=1}^{k}[(\mathbf{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\varphi}_{ij}]^2\Big)+\sum_{j=1}^{k}\log\lambda_{ij}+(d-k)\log\delta \quad (1)$$

where $\boldsymbol{\mu}_i$ denotes the mean vector of class ω_i, λ_{ij} and $\boldsymbol{\varphi}_{ij}$ denote the *j*-th largest eigenvalue and the corresponding eigenvector of the covariance matrix of class ω_i, respectively. k ($k<d$) is the number of dominant principal components. To compensate for the estimation error of parameters on limited training samples, the minor eigenvalues are replaced with a constant δ. The discriminant function in (1) is a distance metric, i.e., the input pattern is classified to the class of minimum distance, and multiple candidate classes are ordered in ascending order of distances.

In MQDF, the *d-k* non-dominant eigenvectors, which contain useful discriminating information for similar characters, are not used effectively. The compound Mahalanobis function (CMF) [5][6][7] makes use of this information to further discriminate the top two candidate classes given by MQDF.

For an input pattern **x**, the top two candidate classes given by MQDF are ω_j and ω_i, which are further discriminated by CMF. With respect to class ω_j, the CMF is formulated as

$$f_{\mathrm{CMF}}(\mathbf{x},\omega_j)=(1-\beta)\,f_{\mathrm{mqdf}}(\mathbf{x},\omega_j)+\beta\,f_{\mathrm{md}}(\mathbf{x}',\omega_j;\omega_i) \quad (0\le\beta\le 1) \quad (2)$$

where $f_{\mathrm{mqdf}}(\mathbf{x},\omega_j)$ is the MQDF and $f_{\mathrm{md}}(\mathbf{x}',\omega_j;\omega_i)$ is an added Mahalanobis distance to ω_j, β is a weight parameter.

The vector $\mathbf{x}'$ is the projection of **x** on an axis in the minor subspace of ω_j:

$$\mathbf{x}'=\boldsymbol{\mu}_j+[(\mathbf{x}-\boldsymbol{\mu}_j)^{\mathrm{T}}\boldsymbol{\psi}]\boldsymbol{\psi}, \quad (3)$$

where

$$\boldsymbol{\psi}=\frac{\sum_{m=k+1}^{d}[(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T\boldsymbol{\varphi}_{jm}]\boldsymbol{\varphi}_{jm}}{\sqrt{\sum_{m=k+1}^{d}[(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T\boldsymbol{\varphi}_{jm}]^2}} \quad (4)$$

is the projected vector of $\boldsymbol{\mu}_i-\boldsymbol{\mu}_j$ on the minor subspace of ω_j. This vector is proposed to carry useful information for discriminating two classes. The added distance $\mathrm{f}_{\mathrm{md}}(\mathbf{x}',\omega_j;\omega_i)$ is then computed by

$$f_{\mathrm{md}}(\mathbf{x}',\omega_j;\omega_i)=\|\mathbf{x}'-\boldsymbol{\mu}_j\|^2/\delta^2=\frac{\Big\{(\mathbf{x}-\boldsymbol{\mu}_j)^T(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)-\sum_{m=1}^{k}[(\mathbf{x}-\boldsymbol{\mu}_j)^T\boldsymbol{\varphi}_{jm}][(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T\boldsymbol{\varphi}_{jm}]\Big\}^2}{\delta^2\Big\{\|\boldsymbol{\mu}_i-\boldsymbol{\mu}_j\|^2-\sum_{m=1}^{k}[(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T\boldsymbol{\varphi}_{jm}]^2\Big\}} \quad (5)$$

In the above formula, the non-dominant eigenvectors of class ω_j, $\phi_{jm}, m=k+1,\ldots,d$, are eliminated according to $\boldsymbol{\mu}_i-\boldsymbol{\mu}_j=\sum_{m=1}^{d}[(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T\boldsymbol{\varphi}_{jm}]\boldsymbol{\varphi}_{jm}$. Thus, the non-dominant eigenvectors are not necessarily stored, and the CMF has not extra parameters compared to the baseline MQDF classifier.

In the same way, we can get the CMF $f_{CMF}(\mathbf{x},\omega_i)$ for class ω_i. Finally, the top two candidates given by MQDF is discriminated by comparing $f_{CMF}(\mathbf{x},\omega_j)$ and $f_{CMF}(\mathbf{x},\omega_i)$, and the input pattern is classified to the class of smaller CMF. If we want to reorder more than two candidate classes given by MQDF, we need to compare each pair of them by CMF, and the class with maximum votes is accepted as the final result.

3. Pair Discrimination Using LDA

The CMF method projects the feature vector onto an axis on the minor subspace of one class, which is based on the covariance of one class only. To improve the separability of two classes, we turn to estimate the discriminant axis using linear discriminant analysis (LDA), which considers the covariance matrices of both classes. In LDA, the projection axis (discriminant vector) **w** for discriminating two classes is estimated to maximize the Fisher ratio:

$$J(\mathbf{w})=\frac{\mathbf{w}^{\mathbf{t}}\mathbf{S}_B\mathbf{w}}{\mathbf{w}^{\mathbf{t}}\mathbf{S}_w\mathbf{w}}, \quad (6)$$

where $\mathbf{S}_B$ and $\mathbf{S}_w$ denote the between-class scatter matrix and within-class scatter matrix, respectively. For two classes ω_i and ω_j, with mean vectors $\boldsymbol{\mu}_i$ and

$\boldsymbol{\mu}_j$, covariance matrices Σ_i and Σ_j, the within-class and between-class scatter matrices can be written as:

$$\mathbf{S}_w = \left(\Sigma_i + \Sigma_j\right)/2 \quad (7)$$

$$\mathbf{S}_B = (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \quad (8)$$

By LDA, the optimal discriminant vector is obtained as:

$$\begin{aligned} \mathbf{w} &= \mathbf{S}_w^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \\ &= \left(\left(\Sigma_i + \Sigma_j\right)/2\right)^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \\ &= \Sigma^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \end{aligned} \quad (9)$$

Σ is a symmetric matrix and can be re-written as $\boldsymbol{\Sigma} = \boldsymbol{\Psi\Lambda\Psi}^{\mathbf{T}}$, where $\Psi = [\psi_1, \psi_2, \ldots, \psi_d]$ and $\Lambda = diag[\lambda_1, \lambda_2, \ldots, \lambda_d]$ (the eigenvalues are ordered in non-ascending order). So,

$$\begin{aligned} \mathbf{w} &= \Sigma^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \\ &= \boldsymbol{\Psi\Lambda}^{-1}\boldsymbol{\Psi}^T(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \\ &= \sum_{n=1}^{d}(1/\lambda_n)\boldsymbol{\psi}_n\boldsymbol{\psi}_n^T(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \end{aligned} \quad (10)$$

In calculating **w**, we encounter the same problem of estimation error of Σ on limited samples: when estimated by maximum likelihood (ML), the minor eigenvalues tend to be under-estimated. For this reason, we set a threshold $b = a$*trace $(\Sigma)/d$ (parameter a is a constant, d is the dimensionality of feature vector): whenever an eigenvalue is smaller than b, it is set equal to b. The parameter a is set empirically.

The vector **w** calculated by (10) is normalized to unit norm: $\tilde{\mathbf{w}} = \mathbf{w}/\|\mathbf{w}\|$. The feature vector of input pattern is then projected onto this axis for discriminating two classes.

In the following, we show that the discriminant vector calculated by LDA covers that of CMF as a special case. When $\Sigma_i \approx \Sigma_j$, we can set $\Sigma = \Sigma_j$ in (9) and the vector **w** is changed to:

$$\mathbf{w} = \sum_{m=0}^{d}(1/\lambda_{jm})\boldsymbol{\varphi}_{jm}\boldsymbol{\varphi}_{jm}^T(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \quad \left(\lambda_{j1} > \lambda_{j2}, \cdots, > \lambda_{jd}\right). \quad (11)$$

When λ_{jm} is large, we can set $1/\lambda_{jm} \approx 0 (m = 1, 2, \cdots, k)$, then, the leading k parts of (11) can be neglected. Further, if we reasonably set the minor eigenvalues to be a small constant: $\lambda_{jm} = \varepsilon\ (m = k+1, k+2, \cdots, d)$, the vector **w** becomes:

$$\begin{aligned} \mathbf{w} &= \sum_{m=k+1}^{d}(1/\lambda_{jm})\boldsymbol{\varphi}_{jm}\boldsymbol{\varphi}_{jm}^T(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \\ &= \sum_{m=k+1}^{d}(1/\varepsilon)[(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T\boldsymbol{\varphi}_{jm}]\boldsymbol{\varphi}_{jm} \end{aligned} \quad (12)$$

The normalized vector $\tilde{\mathbf{w}} = \mathbf{w}/\|\mathbf{w}\|$ is then the same as $\boldsymbol{\psi}$ in (4).

From (10), the discriminant vector is computed from the covariance matrices of two classes. Since the diagonalization of the average covariance matrix is computationally expensive, it must be computed offline and the discriminant vectors of selected class pairs should be stored. On the other hand, the discriminant vectors of the CMF method can be computed online by Eq. (4) and applies to all pairs of classes without storing extra parameters. However, the CMF method implicitly assumes that the covariance matrices of each pair of classes are identical. This results in degraded classification performance.

4. Similar Character Recognition

We use the MQDF as the baseline classifier, and for each input pattern **x**, the MQDF classifier gives two top-rank candidate classes ω_i and ω_j. The compound function is used to further discriminate to which class the input pattern belongs. The compound functions for two classes are computed by

$$\begin{cases} f(\mathbf{x}, \omega_i) = (1-\beta) * f_{\text{mqdf}}(\mathbf{x}, \omega_i) + \beta * f_{\text{LDA}}(\mathbf{x}, \omega_i) \\ f(\mathbf{x}, \omega_j) = (1-\beta) * f_{\text{mqdf}}(\mathbf{x}, \omega_j) + \beta * f_{\text{LDA}}(\mathbf{x}, \omega_j) \end{cases} \quad (13)$$

We can then decide that:

$$\begin{cases} \mathbf{x} \in \omega_i & \text{if } f(\mathbf{x}, \omega_i) \le f(\mathbf{x}, \omega_j) \\ \mathbf{x} \in \omega_j & \text{otherwise} \end{cases} \quad (14)$$

The functions $f_{\text{LDA}}(\mathbf{x}, \omega_i)$ and $f_{\text{LDA}}(\mathbf{x}, \omega_j)$ are the distance functions projected on the discriminant vector $\tilde{\mathbf{w}}$ of class ω_i and ω_j.

Projecting **x** and $\boldsymbol{\mu}_i$ on the vector $\tilde{\mathbf{w}}$, we get:

$$\begin{aligned} \hat{x} &= \mathbf{x}^{\mathbf{T}}\tilde{\mathbf{w}} \\ \hat{\mu}_i &= \boldsymbol{\mu}_i^T\tilde{\mathbf{w}} \\ f_{\text{LDA}}(\mathbf{x}, \omega_i) &= f_{\text{ED}}(\mathbf{x}, \omega_i) = (\hat{x} - \hat{\mu}_i)^2 / C \end{aligned} \quad (15)$$

C is a constant in case that the output of $f_{\text{ED}}(\mathbf{x}, \omega_i)$ is far larger than $f_{\text{LDA}}(\mathbf{x}, \omega_i)$.

Considering that the projections on the discriminant vector are dispersive, there are regions of intersection for two similar classes, so we replace Euclidean distance with Mahalanobis distance:

$$f_{\text{LDA}}(\mathbf{x}, \omega_i) = f_{\text{MD}}(\mathbf{x}, \omega_i) = (\hat{x} - \hat{\mu}_i)^2 / \hat{\delta}_i^2 \quad (16)$$

where $\hat{\delta}_i^2$ is the variance in one-dimensional projected subspace for class ω_i.

5. Experiment results

We evaluated our methods on the ETL9B database and CASIA database. The ETL9B database, collected by the Electro-Technical Laboratory of Japan, contains handwritten samples of 3,036 characters, including 2,965 Kanji characters and 71 hiragana, 200 samples per class. We choose the first 20 and last 20 samples from each class for testing, and the remaining 160 samples from each class for training. The CASIA database, collected by the institute of automation, Chinese academy of sciences, contains 3,755 Chinese characters of the level-1 set of the standard GB2312-80, 300 samples per class. We choose 250 samples per class for training and the remaining 50 samples per class for testing.

Each binary character image is normalized to gray-scale image of 64*64 pixels by the named bi-moment normalization method [9] and 8-direction gradient direction features [10] are extracted. The resulting 512-dimensional feature vector is projected onto a 160-dimensional subspace learned by LDA. The 160-dimensional projected vector is then fed to classification.

In the experiments, the δ in Eq. (1) and (5) is set equal to the average of all eigenvalues, so is C in (15). The parameter a in the equation $b = a$*trace $(\sum)/d$ is set to be 0.3, which performs best among the values $0.1, 0.2, \cdots, 1$. Similar character pairs are selected on the training dataset through 5-fold cross validation using MQDF. When the first candidate is not the genuine class, the genuine class is paired with up to three preceding top-rank classes. Each similar pair is estimated a discriminant vector by LDA on the training set.

Table1. Numbers of similar pairs for MQDF with different number k

k	10	20	30	40	50
ETL9B	7904	7221	7005	6830	6994
CASIA	17191	15570	14937	14664	14960

The number k of principal eigenvectors of MQDF was chosen as 10, 20, 30, 40, and 50. Correspondingly, the numbers of similar class pairs are as in Table 1.

We compare our LDA-based compound distance method with the CMF method in [6] [7] (denoted by MQDF+CMF). Our LDA-based has two versions depending on the distance metric in pair discrimination: Euclidean distance (ED) and Mahalanobis distance (MD). The two methods are denoted as MQDF+ED and MQDF+MD, respectively.

The test accuracies on ETL9B and CASIA using MQDF with fixed number of eigenvectors k=40 and different compound function weight β are shown in Fig. 1 and Fig. 2, respectively ($\beta = 0$ corresponds to MQDF). In Fig. 1, we can see that the best accuracy on ETL9B reaches 99.31% by MQDF+MD ($\beta = 0.6$), 99.30% by MQDF+ED ($\beta = 0.8$), and 99.28% by MQDF+CMF ($\beta = 0.8$). In Fig. 2, the best accuracies of MQDF+MD, MQDF+ED, and MQDF+CMF are 98.35%, 98.31%, and 98.28%, respectively. From Fig. 1 and Fig. 2, we can also see that using only ED or MD is better than using only CMF in discriminating similar characters ($\beta = 1$).

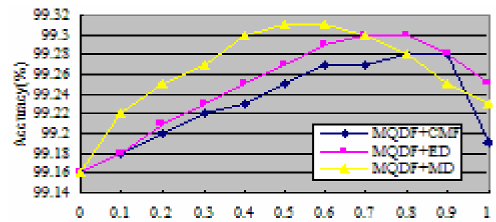

Fig. 1. Test accuracies on ETL9B using MQDF with eigenvector k=40 as baseline classifier.

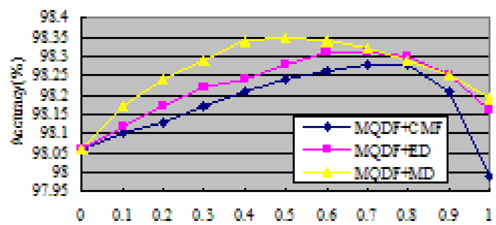

Fig. 2. Test accuracies on CASIA using MQDF with eigenvector k=40 as baseline classifier.

Fig. 3 and Fig. 4 show the test accuracies using MQDF as baseline classifier with k=10,20,30,40,50 and with $\beta = 0.6$ for all compound distances. Again, it is shown that the MQDF+MD and MQDF+ED outperform the MQDF+CMF.

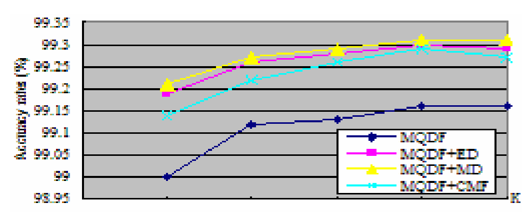

Fig. 3. Test accuracies on ETL9B using MQDF as baseline classifier with different k.

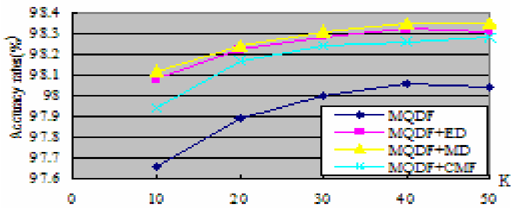

Fig. 4. Test accuracies on CASIA using MQDF as baseline classifier with different k.

From Table 1, we can see that when k=40, the numbers of extra discriminant vectors are 6,830 and 14,664 for ETL9B and CASIA, respectively. On average, each class has extra 2.45 and 3.91 vectors, respectively, compared to 41, the number of stored class mean and eigenvectors. So, the addition of parameter storage is moderate.

Increasing the number of similar pairs could further improve the recognition accuracy, though accompanied with increased storage.

Table 2 gives the average CPU times of classifying a test sample using MQDF (k=40), MQDF+CMF, and MQDF+MD on a computer with AMD 64 Athlon-X2 2GHz processor. We can see that MQDF+CMF and MQDF+MD add only a little computation overhead to MQDF. MQDF+MD is less computationally intensive than MQDF+CMF because in MQDF+MD, the discriminant vectors are pre-computed in training, while the discriminant vectors of MQDF+CMF are computed real time in classification.

Table2. CPU times of classification by MQDF (k=40), MQDF+CMF, and MQDF+MD

	MQDF	MQDF+CMF	MQDF+MD
ETL9B	9.31ms	11.09ms	10.74ms
CASIA	10.35ms	12.04ms	11.83ms

6. Conclusion

In this paper, we propose an LDA-based compound distances method for discriminating similar handwritten Chinese characters. The LDA-based method takes into account the difference of covariance matrices of two classes, thus yield better discrimination performance than the previous compound function methods. Though the LDA-based method needs to store extra discriminant vectors for selected similar pairs, it gives higher recognition accuracies than the previous methods.

The proposed method can be combined with discriminative feature extraction and discriminative learning of quadratic classifier parameters [11] for further improving the performance.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC) under grants no.60543004 and no.60121302.

Reference

[1] T. Ishii, Y. Waizumi, N. Kato, and Y. Nemoto, Recognition system for handwritten characters by alternative method using neural network, *Trans. IEICE Japan*, J83-D-II(3): 988-995, 2000.

[2] Y. Jin and S. Ma, Pairwise classifier combination and its application on Chinese character recognition, *Proc. 5th World Congress on Intelligent Control and Automation*, Hangzhou, 2004, Vol.5, pp.4075-4078.

[3] Y. Jin and S. Ma, Experiment on linear classification of Chinese character, *Journal of Chinese information processing* (in Chinese), 14(2): 55-60, 2002.

[4] M. Suzuki, S. Ohmachi, N. Kato, H. Aso, and Y. Nemoto, A discrimination method of similar characters using compound Mahalanobis function, *Trans IEICE Japan*, J80-D-II(10): 2752-2760, 1997.

[5] T. Nakajima, T. Wakabayashi, F. Kimura, and Y. Miyake, Accuracy improvement by compound discriminant functions for resembling character recognition, *Trans IEICE Japan*, J83-D-II(2): 623-633, 2000.

[6] H. Liu and X. Ding. Handwritten character recognition using gradient feature and quadratic classifier with multiple discrimination schemes, *Proc.8th ICDAR*, Seoul, Korea, 2005, pp.19-23.

[7] H. Liu and X. Ding, Handwritten Chinese character recognition based on mirror image learning and the compound Mahalanobis function, *Journal of Tsinghua Univ.* (in Chinese), 46(7): 1239-1242, 2006.

[8] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake, Modified quadratic discriminate functions and its application to Chinese character recognition, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 9(1): 149-153, 1987.

[9] C.-L. Liu, H. Sako, and H. Fujisawa, Handwritten Chinese character recognition: alternatives to nonlinear normalization, *Proc. 7th ICDAR*, Edinburgh, Scotland, 2003, pp. 524-528.

[10] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, Handwritten digit recognition: investigation of normalization and feature extraction techniques, *Pattern Recognition*, 37(2): 265-279, 2004.

[11] C.-L. Liu, High accuracy handwritten Chinese character recognition using quadratic classifiers with discriminative feature extraction, in *Proc. 18th ICPR*, Hong Kong, 2006, Vol.2, pp.942-946