FEATURE EXTRACTION FOR INTEGRATED PATTERN RECOGNITION SYSTEMS

X. Wang and K. K. Paliwal

School of Microelectronical Engineering
Griffith University, Brisbane
QLD 4111, Australia

ABSTRACT

Conventional pattern recognition systems have two components: feature analysis and pattern classification. Feature analysis is achieved in two steps: parameter extraction and feature extraction. Feature extraction and pattern classification can be conducted independently or jointly. Two popular independent feature extraction algorithms are Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA). Minimum Classification Error (MCE) algorithm and Support Vector Machine (SVM) are two integrated pattern classification algorithms. This paper investigates the two integrated pattern classification algorithms. A generalized structure is proposed to enhance the performance of MCE and SVM.

1. INTRODUCTION

The goal of pattern recognition systems is to classify input data into given classes. Conventional pattern recognition systems achieve this goal through feature analysis and pattern classification, as shown in Figure 1. Feature analysis includes two steps: parameter extraction and feature extraction. In the parameter extraction step, information relevant for pattern classification is extracted from the input data in the form of a p-dimensional parameter vector x. In the feature extraction step, the parameter vector x is transformed to a m-dimensional feature vector y, $m \leq p$. If the parameter extractor is properly designed so that the parameter vector x is matched to the pattern classifier and/or its dimensionality is low, then there is no necessity for the feature extraction step. However, in practice, parameter vectors are not suitable for pattern classifiers. For example, parameter vectors have to be decorrelated before applying them to a classifier based on Gaussian mixture models (with diagonal variance matrices). Furthermore, the dimensionality of parameter vectors is normally very high and needs to be reduced for the sake of less computational cost and system complexity. Due to these reasons, feature extraction has been an important part in pattern recognition tasks.

Feature extraction can be conducted independently or jointly with either parameter extraction or classification. LDA

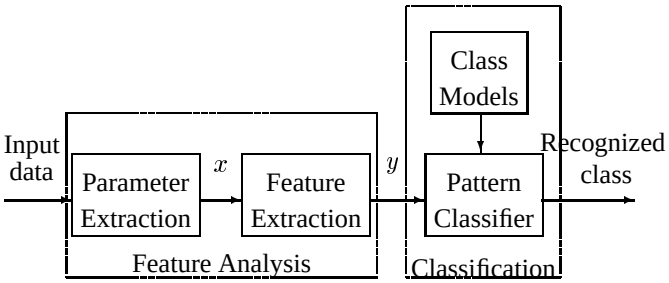

Figure 1: Conventional pattern recognition system.

[1] and PCA [2] are the two popular independent feature extraction methods. However, the drawback of independent feature extraction algorithms is that their optimization criteria are different from the classifier's minimum classification error criterion, which may cause inconsistency between feature extraction and the classification stages of a pattern recognizer and consequently, degrade the performance of classifiers [4]. A direct way to overcome this problem is to conduct feature extraction and classification jointly with a consistent criterion. MCE training algorithm [3, 4, 5] provides such an integrated framework. It is a type of discriminant analysis but achieves minimum classification error directly when extracting features. This direct relationship has made MCE training algorithm widely popular in a number of pattern recognition applications, such as dynamic time-wrapping based speech recognition[6] and Hidden Markov Model (HMM) based speech and speaker recognition[7]. SVM is another recently developed kernel-based integrated pattern classification algorithm. It is based on the idea that the classification that affords dot-products can be computed efficiently in higher dimensional feature spaces [8, 9]. The classes which are not linearly separable in the original parametric space can be linearly separated in the higher dimensional feature space. Because of this, SVM has the advantage that it can handle the classes with complex non-linear decision boundaries. SVM has now evolved into an active area of research [10, 11].

This paper investigates the performance of both MCE

training algorithm and SVM in vowel recognition. A generalized structure is proposed to enhance the performances of both MCE and SVM algorithms. The rest of this paper is organized as follows: Section 2 introduces the framework of MCE training algorithm and SVM. Section 3 proposes a generalized structure to enhance the performance of MCE and SVM. Section 4 shows the results of recognition experiments on TIMIT database.

2. INTEGRATED PATTERN CLASSIFICATION SYSTEMS

2.1. MCE Training Algorithm

Consider an input vector x and a transformation T, let $y = Tx$ be the feature vector in the feature space $\mathcal{F}$. The classifier makes its decision in $\mathcal{F}$ by the following decision rule:

$$x \in Class\ k \quad if \quad g_k(y,\Lambda) = \max_{for\ all\ i \in K} g_i(y,\Lambda) \qquad (1)$$

where $g_i(y,\Lambda)$ is discriminant function of y to class i, Λ is the parameter set and K is the number of classes. A deferentiable misclassification measure can be formulated from Eq. (1) as:

$$d_k(y,\Lambda) = \frac{[\frac{1}{N-1}\sum_{for\ all\ i \neq k}(g_i(y,\Lambda))^\eta]^{1/\eta}}{g_k(y,\Lambda)} \qquad (2)$$

where η is a positive number and $g_k(x,\Lambda)$ is the discriminant of observation x to its known class k. When η approaches ∞, misclassification measure reduces to:

$$d_k(y,\Lambda) = \frac{g_j(y,\Lambda)}{g_k(y,\Lambda)} \qquad (3)$$

where class j has the largest discriminant value among all the classes other than class k. Eq. (3) is not suitable for direct minimization yet. A loss function, which employs the sigmoid function, is defined to smooth the misclassification measure. Expressing explicitly in x and T, loss function is given as:

$$l_k(Tx,\Lambda) = f(d_k(Tx,\Lambda)) = \frac{1}{1+e^{-\xi d_k(Tx,\Lambda)}} \qquad (4)$$

where $\xi > 0$. For a training set $\mathcal{X}$, the empirical loss is:

$$L(\Lambda) = E\{l_k(Tx,\Lambda)\} = \sum_{k=1}^{K}\sum_{i=1}^{N_k} l_k(Tx^{(i)},\Lambda) \qquad (5)$$

where N_k is the number of samples in class k. The class parameter set Λ and transformation matrix T is optimized through the steepest gradient descent method. This is an iterative algorithm and the iteration rules are:

$$\begin{aligned} \Lambda_{t+1} &= \Lambda_t - \epsilon \nabla L(\Lambda)|_{\Lambda=\Lambda_t} \\ T_{sq}(t+1) &= T_{sq}(t) - \epsilon \frac{\partial L}{\partial T_{sq}}\Big|_{T_{sq}=T_{sq}(t)} \\ \nabla L(\Lambda) &= \begin{bmatrix} \partial L/\partial \lambda_1 \\ \vdots \\ \partial L/\partial \lambda_m \end{bmatrix} \end{aligned} \qquad (6)$$

where t denotes tth iteration, $\lambda_1, \cdots, \lambda_m \in \Lambda$ are class parameters, $\epsilon > 0$ is the adaption constant and s and q are the row and column indicators of T.

2.2. Support Vector Machine

Considering a two-class case, suppose the two classes are ω_1 and ω_2. We have training data $\mathcal{X} = \{x_1, \cdots, x_N\} \subset \mathcal{R}^p$. The training data are labelled by the following rule:

$$y_i = \begin{cases} +1 & x_i \in \omega_1 \\ -1 & x_i \in \omega_2 \end{cases} \qquad (7)$$

SVM first maps the training data into a high dimensional feature space $\mathcal{F}$ through a non-linear mapping $\Phi : \mathcal{R}^p \rightarrow \mathcal{F}$, where $\mathcal{R}^p$ is the parameter space. Then a linear discriminant function in $\mathcal{F}$ is computed as:

$$f(x) = (w \cdot \Phi(x)) + b \qquad (8)$$

where $(\cdot)$ denotes the dot product. Ideally, the discriminants of all data satisfy the following constraints [10]:

$$y_i[(w \cdot \Phi(x_i)) + b] - 1 \geq 0 \quad \forall i \qquad (9)$$

Considering the points $\Phi(x_i)$ in $\mathcal{F}$ for which the equality in (9) holds, these points lie on two hyperplanes $H_1 : (w \cdot \Phi(x_i)) + b = +1$ and $H_2 : (w \cdot \Phi(x_i)) + b = -1$. These two hyperplanes are parallel and no training points fall between them. The margin between them is $\frac{2}{||w||}$. Therefore we can find a pair of hyperplanes with maximum margin by minimizing $||w||^2$ subject to Eq.(9)[10]. This problem can be written as a convex optimization problem:

$$\begin{aligned} &minimize \quad \tfrac{1}{2}||w||^2 \\ &subject\ to \quad y_i(w \cdot \Phi(x_i)) + b - 1 \geq 0 \quad \forall i \end{aligned} \qquad (10)$$

Eq. (10) can be solved by constructing a Lagrange function from both the objective function and the corresponding constraints. We introduce positive Lagrange multipliers $\alpha_i, i = 1, \cdots, N$, one for each constraint in Eq.(10). The Lagrange function is given by:

$$L_P = \frac{1}{2}||w||^2 - \sum_{i=1}^{N} \alpha_i y_i (w \cdot \Phi(x_i) + b) + \sum_{i=1}^{N} \alpha_i \qquad (11)$$

L_P must be minimized with respect to w and b, which requires the gradient of L_P to vanish with respect to w and

b. Thus we obtain the $Karush-Kuhn-Tucker$ (KKT) conditions:

$$\begin{array}{l} w_s - \sum_{i=1}^{N} \alpha_i y_i \Phi(x_{is}) = 0 \quad s = 1, \cdots, p \\ \sum_{i=1}^{N} \alpha_i y_i = 0 \\ y_i(w \cdot \Phi(x_i)) + b - 1 \geq 0 \quad \forall i \\ \alpha_i \geq 0 \quad \forall i \\ \alpha_i(y_i(w \cdot \Phi(x_i)) + b - 1) = 0 \quad \forall i \end{array} \tag{12}$$

where w, b and α are the variables to be solved. From KKT condition (12) we obtain:

$$\begin{array}{l} w = \sum_{i=1}^{N} \alpha_i y_i \Phi(x_i) \\ \sum_{i=1}^{N} \alpha_i y_i = 0 \end{array} \tag{13}$$

Define kernel function $k(x_i, x) = (\Phi(x_i) \cdot \Phi(x_j))$ in $\mathcal{F}$ and substitute Eq. (13) into Eq. (11), we obtain the dual optimization problem on α:

$$\begin{array}{ll} maximize & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j k_{ij} + \sum_{i=1}^{N} \alpha_i \\ subject\,to & \sum_{i=1}^{N} \alpha_i y_i = 0 \\ & \alpha_i \geq 0 \quad \forall i \end{array} \tag{14}$$

The solution of Eq. (14), however, is beyond the scope of this paper. A detailed discussion of this problem can be found in [9].

3. A GENERALIZED STRUCTURE FOR MCE AND SVM

Both MCE and SVM, however, have limitations. The major limitation of MCE is that it is sensitive to the initial value of transformation matrix T due to the gradient descent optimization method used in MCE. Conventionally, T is initialized as an identity matrix [5]. Such initialization, however, cannot provide good starting feature space for gradient descent optimization in many cases. The problem with SVM is that classification has to be conducted in the parameteric space. However, the parameteric space normally includes large amount of information irrelevant for classification and has high dimensionality. Thus SVM classifiers are complex and inefficient. The problems with both MCE and SVM are caused by the initial feature space used. These problems can be solved by a two-layer structured pattern classification system, as shown in Fig. 2. The first layer uses linear discriminant analysis to provide a discriminated feature space for the next integrated classification layer. The formulation of LDA is given as follows:

Suppose we have K classes, $\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_K$. Let the ith observation vector from the $\mathcal{X}_j$ be x_{ji}, where $j = 1, \ldots, J$ and $i = 1, \ldots, N_j$. J is the number of classes and N_j is the number of observations from $class\ j$. The *within-class* covariance matrix S_W and *between-class* covariance matrix

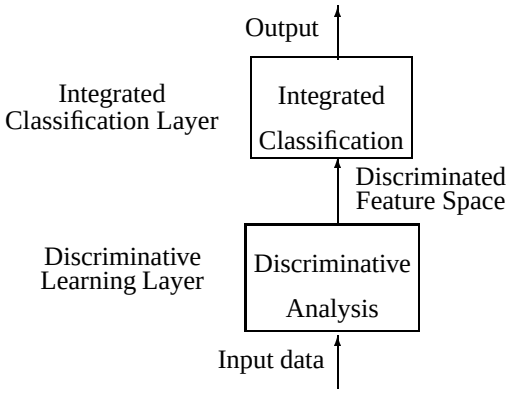

Figure 2: Two-layer integrated pattern classification system.

are defined by:

$$\begin{array}{l} S_W = \sum_{j=1}^{K} \sum_{i=1}^{N_j} (x_{ij} - \mu_j)(x_{ij} - \mu_j)^T \\ S_B = \sum_{j=1}^{K} N_j (\mu_j - \mu)(\mu_j - \mu)^T \end{array} \tag{15}$$

where μ_j is the mean of class j and μ is the overall mean. LDA chooses a linear transformation matrix T that maximizes the objective function

$$J(T) = \frac{|T^T S_B T|}{|T^T S_W T|} \tag{16}$$

The solution of Eq. (16) is that the ith column of an optimal W is the eigenvector corresponding to the ith largest eigenvalue of matrix $S_W^{-1} S_B$.

For simplicity reasons, MCE and SVM learning algorithms conducted under this generalized structure are denoted as Generalized MCE (GMCE) and Generalized SVM (GSVM), respectively.

4. CLASSIFICATION EXPERIMENTS

Our experiments focus on vowel recognition tasks. The classifier used in MCE and GMCE is Mahalanobis distance measure based minimum distance classifier:

$$d_i(y) = (y - \mu_i)^T \Sigma_i^{-1} (y - \mu_i) \quad i = 1, \cdots, K \tag{17}$$

where μ_i is the mean vector of class i and Σ_i is the covariance matrix. The kernel function employed in SVM is polynomial kernel, which has the formulation as:

$$k(x, y) = (x \cdot y + 1)^p \tag{18}$$

where p is the order of the kernel.

TIMIT database is used for the experiments. The vowels used are listed in Table 1. The center 20 msec segments of selected vowels are excised from each sentence. Spectral analysis is performed on these segments and each segment

is represented by a 21-dimension Mel-Frequency Cepstral Coefficients (MFCCs) feature vectors. Each vector contains 1 energy coefficient and 20 MFCCs. The results of MCE, GMCE, SVM and GSVM are given in Figure 3.

Phonemes	aa	ae	ah	ao	aw	ax	ay	eh	oy	uh
Training	541	665	313	445	126	207	395	591	118	57
Testing	176	214	136	168	40	89	131	225	49	21

Phonemes	el	en	er	ey	ih	ix	iy	ow	uw	Total
Training	145	97	384	346	697	583	1089	336	106	7241
Testing	42	34	135	116	239	201	381	116	37	2550

Table 1: Number of selected phonemes in training dataset.

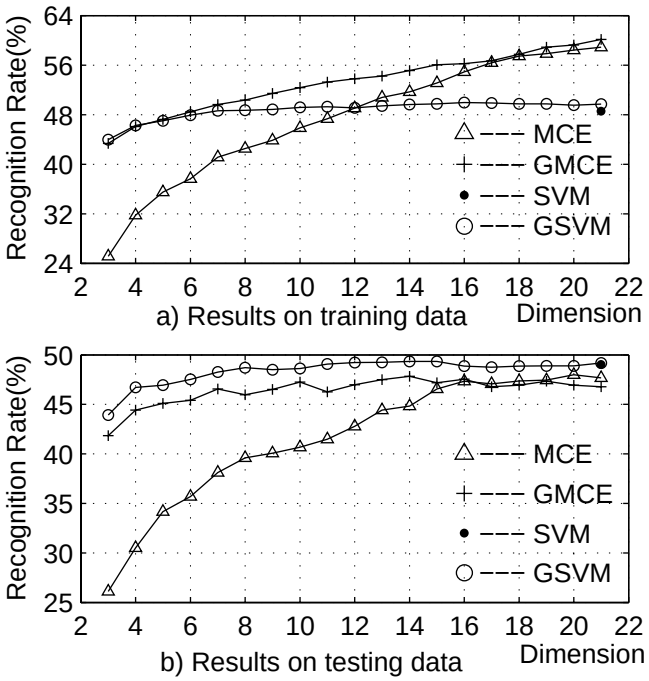

Figure 3: Results of MCE and GMCE training algorithms and SVM on TIMIT database

5. CONCLUSIONS

The following conclusions can be drawn from the results of experiments on TIMIT database.

- MCE has a better fitness to the training data, while SVM has better generalization properties.
- The performances of MCE and SVM under the generalized two-layer pattern classification structure are improved.
- The performances of both GMCE and GSVM are significantly improved on low dimensional feature spaces.

6. REFERENCES

[1] Bocchieri E.L. and Wilpon J.G., "Discriminative analysis for feature reduction in automatic speech recognition", *Proceedings of IEEE International Conference of Acoustics, Speech and Signal Processing*, vol.1, pp. 501-504, 1992.

[2] Jolliffe I.T., *Principal component analysis*, Springer-Verlag, New York, 1986.

[3] Katagiri S., Lee C.H. and Juang B.H., "A Generalized Probabilistic Descent Method", *Proceedings of the Acoustic Society of Japan, Fall Meeting*, pp. 141-142, 1990.

[4] Juang B.H. and Katagiri S., "Discriminative Learning for Minimum Error Classification",*IEEE Transactions on Signal Processing*, Vol. 40, No. 12, 1992.

[5] Paliwal K.K., Bacchiani M. and Sagisaka Y., "Simultaneous Design of Feature Extractor and Pattern Classifier Using the Minimum Classification Error Training Algorithm", *Proceedings of IEEE Workshop on Neural Networks for Signal Processing*, Boston, USA, pp. 67-76, September, 1995.

[6] P.C. Chang, S.H. Chen and B.H. Juang, Discriminative Analysis of Distortion Sequences in Speech Recognition, Proceedings of IEEE International Conference of Acoustics, Speech and Signal Processing, vol. 1, 1991, pp. 549-552.

[7] C.S. Liu, C.H. Lee, W. Chou and B.H. Juang, A Study on Minimum Error Discriminative Training for Speaker Recognition, Journal of Acoustical Society of America, 97(1)(1995) 637-648.

[8] Roth V. and Steinhage V., "Nonlinear discriminant analysis using kernel functions", Technical Report IAI-TR-99-7, University Bonn, 1999.

[9] Smola A.J. and Scholkopf B., "A tutorial on support vector regression", *NeuroCOLT2 Technical Report Series NC2-TR-1998-030*, ESPRIT working group on Neural and Computational Learning Theory "NeuroCOLT 2", 1998.

[10] Burges C.J.C., "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, 2(2), pp.955-974, 1998.

[11] Clarkson P. and Moreno P.J., "On the use of Support Vector Machines for Phonetic Classification". *proceedings of ICCASP '99*, 1999