

Research Article

Facial Affect Recognition Using Regularized Discriminant Analysis-Based Algorithms

Chien-Cheng Lee, Shin-Sheng Huang, and Cheng-Yuan Shih

Department of Communications Engineering, Yuan Ze University, 135 Yuan-Tung Road, Chung-Li, Taoyuan County 320, Taiwan

Correspondence should be addressed to Chien-Cheng Lee, clee@saturn.yzu.edu.tw

Received 3 December 2009; Accepted 17 February 2010

Academic Editor: Jenq-Neng Hwang

Copyright © 2010 Chien-Cheng Lee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a novel and effective method for facial expression recognition including happiness, disgust, fear, anger, sadness, surprise, and neutral state. The proposed method utilizes a regularized discriminant analysis-based boosting algorithm (RDAB) with effective Gabor features to recognize the facial expressions. Entropy criterion is applied to select the effective Gabor feature which is a subset of informative and nonredundant Gabor features. The proposed RDAB algorithm uses RDA as a learner in the boosting algorithm. The RDA combines strengths of linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA). It solves the small sample size and ill-posed problems suffered from QDA and LDA through a regularization technique. Additionally, this study uses the particle swarm optimization (PSO) algorithm to estimate optimal parameters in RDA. Experiment results demonstrate that our approach can accurately and robustly recognize facial expressions.

1. Introduction

Human-computer interaction (HCI) technologies have attracted more and more attention. The traditional interface devices such as the keyboard and mouse are constructed to transmit explicit messages. The implicit information about the user, such as changes in the affective state, is ignored. However, HCI is moving gradually from computer-centered designs toward human-centered designs [1, 2]. Designs for human-centered computing should focus on the human portion of the HCI context, like nonlinguistic conversational signal, emotion, and affective states. Human-centered interfaces must have the ability to detect human affective behavior because it conveys fundamental components of human-human communication. These affective states motivate human actions and enrich the meaning of human communication.

Previous research [3] shows that 55% of face-to-face human communication is relied on facial expressions, indicating that facial expressions play an important role in social interactions between human beings. As a result, facial expressions are also an important part of HCI. Thus,

automatic facial expression recognition in the human-computer environment is an essential and challenging task.

Various techniques have been developed for automatic facial expression recognition. Three recent surveys [4–7] on this topic indicate that facial expression recognition has grown more sophisticated. Facial expression recognition techniques can be categorized based on recognition targets or data sources. With respect to recognition targets, most techniques attempt to recognize a small set of prototypic emotional expressions, that is, happiness, surprise, anger, sadness, fear, and disgust, as well as the neutral state. This practice is based on the work of Darwin [8] and more recently Keltner and Ekman [9] who proposed that basic emotions have corresponding prototypic facial expressions. Ekman and Friesen [10] developed the Facial Action Coding System (FACS) for describing facial expressions in terms of action units (AUs). FACS consists of 46 AUs, which describe basic facial movements based on muscle activities. Various researchers engage in AUs recognition to model facial actions [11].

Facial expression recognition techniques can generally be divided into two categories based on their data sources: static

images and image sequences. In sequence-based methods, an image sequence displays one expression. Thus, a neutral face must be identified first to serve as a baseline face. Then, expression recognition depends on the difference between the baseline face and the following input face image. Optical flow estimation is a typical method of extracting facial features. Yacoub and Davis [12] used the optical flow approach to track the motion of facial features from image sequences and classified the extracted facial features into six basic expressions. Bartlett et al. [13] used a method combining principal component analysis (PCA) with optical flow for facial expression recognition. Essa and Pentland [14] used optical flow in a physical model of the face with a recursive framework to classify facial expressions. Xiang et al. [15] used Fourier transform to extract facial features and represent expressions. These features are then processed using fuzzy C means clustering to generate a spatiotemporal model for each expression type.

If the baseline image in the sequence-based methods is not identified correctly, it is difficult to identify the facial expression for a given image frame. However, facial expression recognition using static images is more difficult than that using image sequences because less information is available. Psychologists often use single images for expression recognition. Therefore, facial expression recognition using static images has attracted a lot of attention. Chen and Huang [16] proposed a clustering-based feature extraction method for facial expression recognition. They used the AR database, created by Aleix Martinez and Robert Benavente, to classify three facial expressions: neutral, smiling, and angry. Zhi and Ruan [17] proposed a method called the two-dimensional discriminant locality preserving projections (2D-DLPPs) algorithm and applied the method to facial expression recognition in a Japanese female facial expression database (JAFFE) and the Cohn-Kanade database. Shin et al. [18] combined the two-dimensional learning discriminant analysis (2D-LDA) and support vector machine (SVM) methods to recognize seven basic expressions. Feng et al. [19] divided face images into small regions and extracted local binary pattern (LBP) histograms as features and then used a linear programming technique to classify seven facial expressions.

This paper proposes a novel regularized discriminant analysis-based boosting algorithm (RDAB) to recognize seven expressions, including happiness, surprise, anger, sadness, fear, disgust, and a neutral state, from static images. The proposed method also employs an entropy criterion to select effective Gabor features for facial image representation which is a subset of informative and non-redundant Gabor features. In RDAB, regularized discriminant analysis (RDA) acts as a learner in the boosting algorithm. RDA combines the strengths of linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) and solves the small sample size and ill-posed problems of QDA and LDA by regularizing parameters. This study also uses a particle swarm optimization (PSO) algorithm to estimate the optimal parameters in RDA. Experiment results demonstrate that the proposed RDAB facial expression method achieves a high

recognition rate and outperforms other facial expression recognition systems.

The rest of this paper is organized as follows. Section 2 provides an overview of the proposed method. Section 3 describes each component of the proposed method. Section 4 presents experiment results. Finally, conclusions are summarized in Section 5.

2. Overview of the Proposed Method

The overall process of the proposed facial expression recognition method is displayed in Figure 1. First, the preprocessing step is applied to input images to produce standardized facial images for subsequent processing. Facial images are detected using the Viola and Jones face detector [20]. The original images are cropped into face images and resized to 128×96 . Then, a histogram equalization method is applied to eliminate variations in illumination. Several previous studies [21] show that classification using downsampled images in pattern recognition produces a higher accuracy than using the original images. Using downsampled images also reduces the computation complexity. Thus, the following feature extraction step uses facial images that have been downsampled by a factor of two.

Automated facial expression recognition must solve two basic problems: facial feature extraction and facial expression classification. Facial feature extraction methods can be categorized in terms of image sequences or static images. Motion extraction approaches directly focus on facial changes that occur due to facial expressions, whereas static image-based methods do not rely on neutral face images to extract facial features. Gabor features are widely used in image analysis because they closely model the receptive field properties of cells in the primary visual cortex [22–24]. Therefore, this study uses Gabor features to recognize facial expressions from static images.

In practice, the dimensionality of a Gabor feature vector is so high that the computation and memory requirements are very large. For example, if an image measures 128×96 pixels, the dimensionality of the Gabor feature vector with three frequencies and eight orientations is 294912 ($128 \times 96 \times 3 \times 8$). Some of these features are similar. In other words, using the Gabor features with all frequencies and orientations is redundant. For this reason, several sampling methods have been proposed to determine the “optimal” subset for extracting Gabor features. Liu et al. [25] proposed an optimal sampling of Gabor features using PCA for face recognition. Additionally, AdaBoost has been widely used for feature selection [26–28]. This study proposes an effective Gabor feature selection to extract the informative Gabor features representing the facial characteristics. Entropy is used as a criterion to measure the importance of the feature. This approach reduces the feature dimensionality without losing much information and also decreases computation and storage requirements.

In the facial expression classification step, several classifiers have been proposed to cope with the facial expression classification problem, including neural networks, support

vector machines (SVMs), and boosting algorithms. SVMs and boosting algorithms are both large margin classifiers primarily designed for two-class classification problems. SVMs often adopt one-against-one strategy or one-against-all procedure to deal with multiclass problems. On the other hand, boosting algorithms can solve multiclass problems using a multiclass learner. Thus, several researchers have used boosting algorithms with different learners to process multiclass problems. Yang et al. [29] adopted the AdaBoost algorithm to learn the combination of optimal discriminative features to construct the classifier and classify seven expressions and several AUs. Lu et al. [30] developed a novel boosting algorithm combined with LDA-based learners for face recognition. This paper proposes an RDA-based boosting algorithm to recognize facial expressions. RDA combines the benefits of LDA and QDA to achieve a higher recognition rate.

3. Automatic Facial Expression Recognition System

The block diagram of the automatic facial expression recognition system is shown in Figure 1. The key components of the proposed approach include (1) preprocessing, (2) effective Gabor feature selection, and (3) RDAB classification. The details of proposed method are described as follows.

3.1. Preprocessing. This preprocessing is an important step because the input images usually have some slight differences, such as head tilt and head size. The preprocessing phase takes the segmented face, normalizes the face images, reduce lighting variations, and downsamples the face images. The preprocessing phase contains the following steps.

- (1) It detects faces from input images.
- (2) It normalizes the face regions with respect to a 128×96 face image by using the eyes and nose as reference points.
- (3) It performs histogram equalization to reduce the nonuniformity in the pixel distributions that may occur due to various imaging situations.
- (4) It downsamples the face images to obtain the low-frequency images. In this way, the noise and the computation complexity will be reduced.

3.2. Effective Gabor Feature Extraction. The Gabor filter is a very useful tool in computer vision and image analysis because it has optimal localization properties in both spatial and frequency analysis. A 2D Gabor filter is a complex field sinusoidal grating modulated by a 2D Gaussian function in the spatial domain. The 2D Gabor filter $G(x, y)$ is defined as

$$h(x, y) = g(x', y') \cdot \exp[i2\pi(Ux + Vy)], \quad (1)$$

where $(x', y') = (x \cos \phi + y \sin \phi, -x \sin \phi + y \cos \phi)$ are rotated coordinates, in which the major axis is oriented at

an angle ϕ from the x -axis, (U, V) represents a particular 2D frequency, and

$$g(x, y) = \frac{1}{2\pi\lambda\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left[\left(\frac{x/\eta}{\sigma_x}\right)^2 + \left(\frac{y}{\sigma_y}\right)^2\right]\right\}, \quad (2)$$

where σ_x and σ_y represent the spatial extent and bandwidth of the filter, and η is the aspect ratio between x and y axes. For convenience, we assume that $\sigma_x = \sigma_y = \sigma$, the aspect ratio is $\eta = 1$, and the x -axis of the Gaussian has the same orientation as the frequency; hence, (1) can be simplified to

$$h(x, y) = g(x', y') \cdot \exp[i2\pi Fx'], \quad (3)$$

where $F = \sqrt{U^2 + V^2}$ is called the modulation frequency.

This study considers a class of self-similar functions called Gabor wavelets. Using (3) as the mother Gabor wavelet, the self-similar filter bank can be derived by dilations and rotations of $h(x, y)$ through the generating function:

$$h_{mn}(x, y) = \beta^{-m}h(x'', y''), \quad (4)$$

$$\beta > 1, \quad m = 1, 2, \dots, S, \quad n = 1, 2, \dots, T,$$

where $x'' = \beta^{-m}(x \cos \phi_n + y \sin \phi_n)$, $y'' = \beta^{-m}(-x \sin \phi_n + y \cos \phi_n)$, and $\phi_n = \pi(n - 1)/T$. The subscripts m and n represent the index for scale (dilation) and orientation (rotation). S is the total number of scales and T is the total number of orientations.

For a given input image $I(x, y)$, the magnitude of a filtered image $F_{mn}(x, y)$ can be obtained as

$$F_{mn}(x, y) = \sqrt{[hR_{mn}(x, y) * I(x, y)]^2 + [hI_{mn}(x, y) * I(x, y)]^2}, \quad (5)$$

where $*$ indicates 2-D convolution, and $hR_{mn}(x, y)$ and $hI_{mn}(x, y)$ represent the real and imaginary parts of the Gabor filters from (4). The real part of Gabor filters with three scales and eight orientations is shown in Figure 2.

Entropy is a measure of the uncertainty associated with a random variable X in information theory, defined as

$$H(X) = - \sum_x p(X = x) \log(p(X = x)). \quad (6)$$

The less uncertainty there is, the less entropy there is. Conversely, more uncertainty produces more entropy. The objective of feature selection is to select a subset of features that gives as much information as possible. Thus, this study formulates an effective feature selection scheme based on the feature position probability distribution to select the informative Gabor features. Let $O_{m,n,x,y}(r)$ denote the occurrence of Gabor magnitude response r in $F_{mn}(x, y)$ for all training images. The feature position probability is defined as

$$p_{m,n,x,y}(r) = \frac{O_{m,n,x,y}(r)}{n_L}, \quad (7)$$

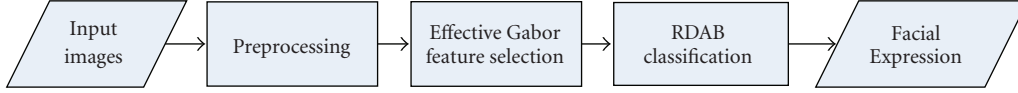


FIGURE 1: Overall process of the proposed facial expression recognition method.

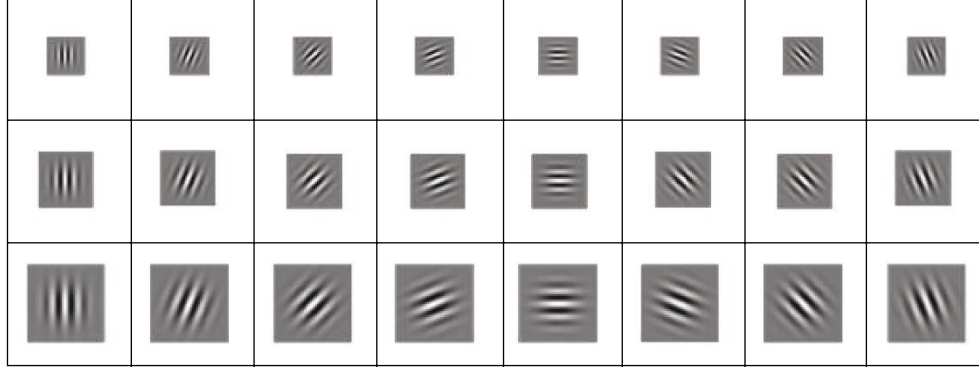


FIGURE 2: The real part of the Gabor filters with three scales and eight orientations.

where

$$n_L = \sum_{r=0}^L O_{m,n,x,y}(r), \quad \text{for } 0 \leq r \leq L. \quad (8)$$

The entropy of the feature position probability distribution is defined as

$$H_{m,n,x,y}(R) = - \sum_r p_{m,n,x,y}(R=r) \log(p_{m,n,x,y}(R=r)), \quad (9)$$

where R is a random variable of the occurrence of Gabor magnitude response. The value of the entropy $H_{m,n,x,y}(R)$ indicates that the uncertainty of the feature at the pixel position (x, y) of the Gabor features with m th scale and n th orientation for all training images. A larger value of $H_{m,n,x,y}(R)$ means that the feature magnitudes vary from different images. Thus, features along this range of the feature space can improve the discriminating power between different expression classes. On the other hand, a smaller value of the entropy $H_{m,n,x,y}(R)$ indicates the corresponding features tend toward the same magnitude. That is, features along this range contribute less to discrimination. To reduce the feature space, these features should not be considered in the classification phase. Accordingly, sort $H_{m,n,x,y}(R)$ in descending order and use the first $M \leq N$ Gabor features as the feature vector to form a lower M -dimensional subspace, where N is the dimensionality of the original feature space and M is that of the effective feature space. Figure 3 shows an example of effective Gabor filters of a face image with three scales and eight orientations. Figure 3(a) shows the magnitudes of the Gabor features. Figure 3(b) shows the points of the top 10 percent of the Gabor features.

3.3. RDA-Based Boosting Algorithm. RDA combines the strengths of LDA and QDA, offering several advantages

compared to the conventional LDA and QDA. LDA and QDA are well known and popular methods in classification and recognition. However, these approaches often suffer from the small sample size problem (SSS) that exists in high-dimensional pattern recognition tasks. To solve the SSS problem, the traditional solution adopts a two-phase framework PCA plus LDA for feature compression and selection. However, the PCA may discard dimensions that contain important discriminative information [31]. Thus RDA, an intermediate method between LDA and QDA, is proposed to deal with this problem [32].

3.3.1. Regularized Discriminant Analysis. Given a set of objects Z , the purpose of classification or discriminant is to assign objects to one of several K classes. The classification rule is based on a quantity called the discriminant score for the k th class, defined as

$$d_k = \min_{1 \leq k \leq K} d_k(Z), \quad (10)$$

with

$$d_k(Z) = (Z - \mu_k)^T \Sigma_k^{-1} (Z - \mu_k) + \ln |\Sigma_k| - 2 \ln \pi_k, \quad (11)$$

where k denotes the k th class, and μ_k and Σ_k are mean and covariance matrix of the k th class, respectively. In the case of LDA, variables are normally distributed in each class with different mean vectors and a common covariance matrix. On the other hand, the variables in QDA are assumed to be normally distributed in each class with different mean vectors and different covariance matrices.

Compared with LDA, which shares the same covariance structure between different classes, QDA better fits the data distribution because it allows for different covariance matrices. However, QDA needs more parameters to estimate

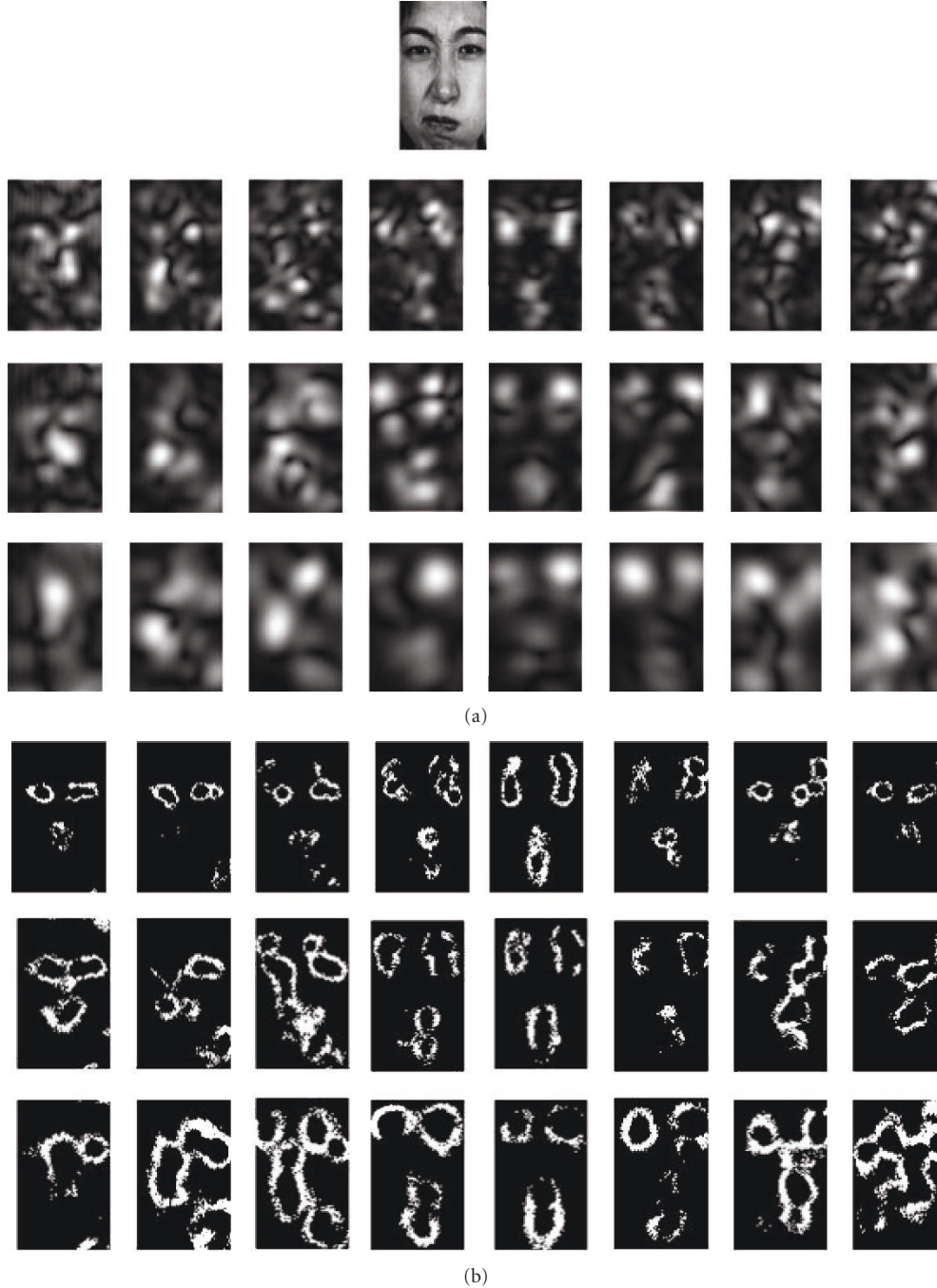


FIGURE 3: Example of the effective Gabor filter of a face image with three scales and eight orientations. (a) Original Gabor features. (b) Points of the top ten percent of the Gabor features (white points).

covariance matrices. Hence, classification based on QDA requires larger samples than those based on LDA. Additionally, in small-sample classification, reducing the number of estimated parameters by using the pooled covariance estimate may lead to superior performance, even if the class covariance matrices are substantially different. That is, LDA generally outperforms QDA in SSS problems. For these reasons, RDA provides a regularization mechanism to shrink the separate covariance of QDA toward a common

covariance as in LDA. The regularized covariance matrix of k th class is

$$\Sigma_k(\lambda) = \lambda S_w + (1 - \lambda)\Sigma_k, \quad (12)$$

where Σ_k is the covariance of the k th class and S_w is the pooled covariance matrix used in LDA, which is also known as the within-class scatter matrix. However, the regularization in (12) is not enough. If the total sample size,

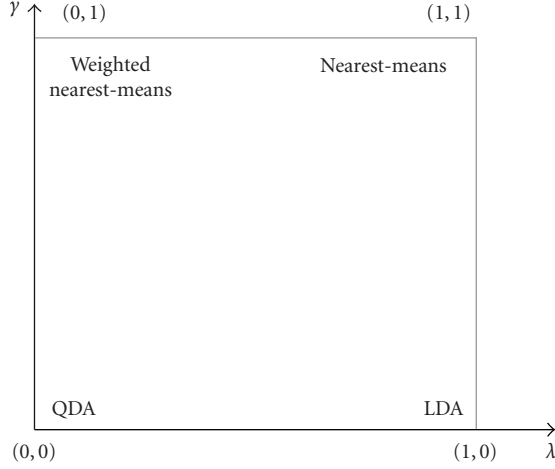


FIGURE 4: RDA model selection. $(\lambda = 0, \gamma = 0)$ represents QDA, $(\lambda = 1, \gamma = 0)$ represents LDA, $(\lambda = 1, \gamma = 1)$ corresponds to the nearest-means classifier, and $(\lambda = 0, \gamma = 1)$ represents a weighted nearest-means classifier.

n , is less than the data dimensionality, QDA and LDA are ill-posed. Additionally, biasing the class covariance matrices toward commonality may not be an effective shrinkage way. According to [33], ridge regression regularizes ordinary linear least squares regression by shrinking toward a multiple of the identity matrix. Therefore, the regularization should be

$$\Sigma_k(\lambda, \gamma) = (1 - \gamma)\Sigma_k(\lambda) + \frac{\gamma}{d}\text{trace}[\Sigma_k(\lambda)]I_d, \quad (13)$$

where I_d is an identity matrix of size d by d and d is the dimensionality of the data. The terms λ and γ represent two parameters that range from 0 to 1.

RDA provides a fairly rich class of regularization alternatives. Figure 4 shows four corners in the (λ, γ) plane representing well-known classification procedures. The lower left corner $(\lambda = 0, \gamma = 0)$ represents QDA. The lower right $(\lambda = 1, \gamma = 0)$ represents LDA. The upper right corner $(\lambda = 1, \gamma = 1)$ corresponds to the nearest-means classifier. The upper left corner of the plane represents a weighted nearest-means classifier.

3.3.2. Model Selection. A good pair of values for λ and γ is not likely to be known in advance. Selecting an optimal value for a parameter pair such as (λ, γ) is called *model selection*. Because model selection is a type of optimization problem, this study uses a PSO algorithm [34] to obtain the optimal parameters. The basic concept of PSO algorithm is described as follows.

Suppose that the i th particle of the swarm is denoted by $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$, and the velocity vector of the i th particle is denoted by $V_i = (v_{i1}, v_{i2}, \dots, v_{id})$. Equations (14) shows the particle position and the velocity vector updating:

$$\begin{aligned} V_i &= V_i + c_1 \text{rand}() (P_i - X_i) + c_2 \text{rand}() (P_g - X_i), \\ X_i &= X_i + V_i, \end{aligned} \quad (14)$$

where $\text{rand}()$ is a random number generated within $(0,1)$. The terms c_1 and c_2 are positive constant parameters which control the maximum step size. $P_i = (p_{i1}, p_{i2}, \dots, p_{id})$ is the best solution achieved so far by X_i particle, and $P_g = (p_{g1}, p_{g2}, \dots, p_{gd})$ is the best solution achieved so far for the whole particles. The velocity vector V_i is confined within $[V_{\min}, V_{\max}]$, and it is set equal to the corresponding threshold if the velocity vector exceeds the threshold V_{\min} or V_{\max} . The PSO procedure is given as follows.

- (1) Randomly initialize X_i and V_i of all particles.
- (2) Evaluate the fitness values of all particles, and update P_i and P_g .
- (3) Update X_i and V_i according to (14).
- (4) Evaluate the fitness values of all particles, and update P_i and P_g .
- (5) If the convergence condition is not reached, go back to step (3).

3.3.3. Boosting Procedure. The ability of a boosting algorithm to reduce the overfitting and generalization errors of classification problems is quite interesting. In the traditional AdaBoost algorithm, the learner is weak and just slightly better than random guessing. In contrast, the proposed RDA-based boosting algorithm uses Direct LDA (DLDA) [35] to reduce dimensionality and extract discriminative features. RDA then performs the classification tasks.

Algorithm 1 illustrates the proposed RDA-based boosting algorithm. Given a training set $X = \{X_i\}_{i=1}^C$, containing C classes, each class $X_i = \{(x_{ij}, y_{ij})\}_{j=1}^{L_i}$ consists of a number of examples x_{ij} and their corresponding class labels y_{ij} . $N = \sum_{i=1}^C L_i$ is the total number of examples in the set. Let \mathbf{X} be the sample space: $x_{ij} \in X$ and at $\mathbf{Y} = \{1, \dots, C\}$ be the label set: $y_{ij} \in Y$. The goal of learning is to estimate a classifier $h(\mathbf{x}) : \mathbf{X} \rightarrow \mathbf{Y}$, which will correctly classify unseen examples. It works by repeatedly applying a given learner to a weighted version of the training set in T iterations and combining these learners $\{h_t\}_{t=1, \dots, T}$ at each iteration into a single strong classifier.

4. Experiment Results

The current study evaluates the proposed algorithm using the JAFFE database (Figure 5) which is commonly used for facial expression recognition tasks. The database includes 210 facial expression images of ten people. Each person has seven expressions (anger, disgust, fear, happiness, sadness, surprise, and the neutral state) and there are three examples of each expression. These images are grayscale and have a resolution of 256×256 . In the preprocessing step, face images are detected by the Viola and Jones face detector, cropped, and resized to 128×96 . Histogram equalization is then applied to eliminate the illumination variation. Afterwards, the resolution of the face images is downsampled to 64×48 .

The current study uses the leave-one-out strategy in the training procedure, as in [18]. The database is divided randomly into thirty segments for each expression. Then,

- (1) Given set of training images $X = \{(x_{ij}, y_{ij})_{j=1}^{L_i}\}_{i=1}^C$ with labels $y_{ij} = i \in Y$, where $Y = \{1, \dots, C\}$; a DLDA feature extractor and an RDA-based learner; and the iteration number, T . Let $B = \{(x_{ij}, y) : x_{ij} \in X, y \in Y, y \neq y_{ij}\}$
- (2) Initialize $d_1(x_{ij}, y) = 1/|B| = 1/N(C-1)$, the mislabel distribution over B .
- (3) For $t = 1, \dots, T$, repeat the following steps:
 - (a) Update the pseudo sample distribution: $\hat{D}_t(x_{ij}) = \sum_{y \neq y_{ij}} d_t(x_{ij}, y)$
 - (b) **If** $t = 1$: randomly choose n samples per class to form a learning set $\mathfrak{R}_1 \subset X$
 - (c) **else**: choose n hardest samples per class based on \hat{D}_t to form $\mathfrak{R}_t \subset X$
 - (d) Train a DLDA feature extractor P , which is a projection matrix, to obtain discriminative feature set $(\psi_t, \{\bar{x}_{i,t}\}_{i=1}^C)$
 - (e) Use PSO to find the optimal parameters (λ, γ) of covariance matrix in RDA.
 - (f) Build an RDA learner $h_t = r(\psi_t, \{\bar{x}_{i,t}\}_{i=1}^C)$, apply it into the entire training set \mathbf{X} , and normalize the classified result from 0 to 1 by $(r_{\max} - r_{x,i})/(r_{\max} - r_{\min})$.
 - (g) Calculate the pseudo-loss produced by h_t : $\hat{\epsilon}_t = (1/2) \sum_{(x_{ij}, y) \in B} d_t(x_{ij}, y)(1 - h_t(x_{ij}, y_{ij}) + h_t(x_{ij}, y))$.
 - (h) Set $\beta_t = \hat{\epsilon}_t/(1 - \hat{\epsilon}_t)$. If $\beta_t = 0$, then $T = t - 1$ and abort loop.
 - (i) Update the mislabel distribution d_t : $d_{t+1}(x_{ij}, y) = d_t(x_{ij}, y) \cdot \beta_t^{(1+h_t(x_{ij}, y_{ij})+h_t(x_{ij}, y))/2}$
 - (j) Normalize γ_{t+1} so that it is a distribution, $d_{t+1}(x_{ij}, y) \leftarrow d_{t+1}(x_{ij}, y) / \sum_{(x_{ij}, y) \in B} d_{t+1}(x_{ij}, y)$
- (4) Output the final composite classifier $h_f(x) = \arg \max_{y \in Y} \sum (\log(1/\beta_t)) h_t(x, y)$

ALGORITHM 1: RDA-based boosting algorithm.

twenty-nine segments per class are used to train and the remaining segment is used to test. The procedure of training and testing is repeated thirty times until each segment has been used in test. Finally, all the recognition rates are averaged to obtain an overall recognition rate for the proposed method.

The performance of the proposed system can be affected by two factors. One is the effective Gabor feature selection and the other is the number of hardest-to-classify samples in RDAB algorithm. Table 1 compares different levels of the effective Gabor features. Results show that the top ten percent of the selected Gabor features with three scales and eight orientations achieve the highest recognition rate of 96.67%. This means that there was a 10-fold reduction in the number of Gabor filters used. The results also clearly show that the Gabor features with three scales and eight orientations outperform those with five scales and eight orientations. In this experiment, the number of iterations in RDAB algorithm is 25. The number of hardest-to-classify images per class is 12.

The RDAB algorithm selects many hardest-to-classify samples to train the learner in each iteration. The number of hardest-to-classify samples affects the learner's capacity. Consequently, the optimal number of hardest-to-classify samples can improve the classifier efficiency. Figure 6 compares the recognition rates for different numbers of hardest-to-classify samples. This figure shows that the highest recognition rate, 96.67%, is achieved when the number of hardest-to-classify is samples 12 for the top ten percent of the selected Gabor features with three scales and eight orientations. As the number of hardest-to-classify samples increases, the recognition rate decreases due to the overfitting problem. These results also indicate that Gabor features with three scales and eight orientations perform better than those with five scales and eight orientations. The number of iterations in RDAB algorithm in this experiment is 25, and the top ten percent of the selected Gabor features are used.

TABLE 1: Recognition rates for different levels of the effective Gabor features.

Percentage of Selected Features	Recognition Rates	
	3 scales and 8 orientations	5 scales and 8 orientations
5	93.81%	90.95%
10	96.67%	92.38%
20	94.76%	93.81%
30	93.81%	94.76%
40	92.86%	93.81%
50	94.29%	92.86%
60	94.76%	92.38%
70	94.29%	92.38%
80	95.24%	92.38%
90	95.71%	92.38%
100	94.29%	92.38%

TABLE 2: Comparison of facial expression recognition using JAFFE database.

Methods	Recognition rates (%)
Zhi and Ruan [17]	95.91
Shin et al. [18]	95.71
Feng et al. [19]	93.8
Zhao et al. [36]	93.72
Qi and Jiang [37]	94.64
Liejun et al. [38]	95.7
Our proposed method	96.67

It is not easy to directly compare our results with others because different research groups conducted different types of tests using different data sets. As mentioned in the introduction, previous experiments can be classified into two main categories: those using static images or



FIGURE 5: Seven basic emotions (from left to right): anger, disgust, fear, happiness, sadness, surprise, and neutral state (images taken from JAFFE database).

TABLE 3: Confusion matrix of facial expression recognition.

	Anger	Disgust	Fear	Happiness	Sadness	Surprise	Neutral	Total
Anger	30							
Disgust		29			1			
Fear		1	27		1	1		
Happiness				30				
Sadness		2			28			
Surprise						29	1	
Neutral							30	
								203/210

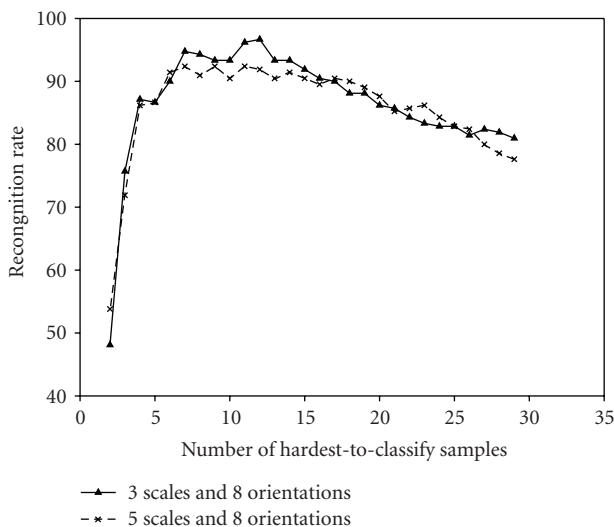


FIGURE 6: A comparison of different numbers of hardest-to-classify samples.

image sequences. This study only compares experiments using JAFFE static images, as Table 2 shows. In [17], facial expression classification performance was tested on feature vectors derived from two-dimensional discriminant locality preserving projections, producing a 95.91% recognition rate. Shin et al. [18] investigated various feature representation and expression classification schemes to recognize seven different facial expressions. Their experiment results show that the method of combining 2D-LDA (Linear Discriminant

Analysis) and SVM outperforms other methods, producing a 95.71% recognition rate using the leave-one-out strategy. Feng et al. [19] reported an accuracy of 93.8% based on local binary pattern histograms and linear programming techniques. In [36], a method for facial expression recognition with Non-negative Matrix Factorization (NMF) and PCA-NMF is presented, and their best recognition rate is 93.72%. The work reported in [37, 38] produced recognition rates of 94.64% and 95.7%, respectively. In this study, we use a leave-one-out strategy similar to that in [18] to verify our algorithm. Our recognition rate of 96.67% outperforms other methods tested on the same database. Table 3 shows the confusion matrix, indicating that anger, happiness, and the neutral state are recognized with very high accuracy, but other expressions (disgust, fear, sadness, surprise) are sometimes confused.

5. Conclusion

This paper proposes a novel facial expression recognition method based on RDA and a boosting algorithm. Since full facial images provide confusing and redundant information for identifying facial expressions, this study proposes an effective Gabor feature selection based on an entropy criterion. This effective Gabor feature is a subset of informative and nonredundant Gabor features. This approach reduces the feature dimensionality without losing much information and decreases computation and storage requirements. This study adopts RDA as a learner in the boosting algorithm. RDA provides a regularization technique to combine the strengths of QDA and LDA. Meanwhile, a PSO algorithm is

adopted to cope with the modal selection problem in RDA. The results of this study show that the proposed method has a high recognition rate of 96.67%, which is better than other reported results. The confusion matrix also shows that anger, happiness, and the neutral state are recognized with very high accuracy.

Acknowledgment

The authors would like to thank the National Science Council (Grant no. NSC 98-2221-E-155-050) for supporting this work.

References

- [1] M. Pantic, A. Pentland, A. Nijholt, and T. Huang, "Human computing and machine understanding of human behavior: a survey," in *Proceedings of the 8th International Conference on Multimodal Interfaces (ICMI '06)*, pp. 239–248, 2006.
- [2] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang, "A survey of affect recognition methods: audio, visual, and spontaneous expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 39–58, 2009.
- [3] A. Mehrabian, "Communication without words," *Psychology Today*, vol. 2, no. 4, pp. 53–56, 1968.
- [4] A. Samal and P. A. Iyengar, "Automatic recognition and analysis of human faces and facial expressions: a survey," *Pattern Recognition*, vol. 25, no. 1, pp. 65–77, 1992.
- [5] M. Pantic and L. J. M. Rothkrantz, "Automatic analysis of facial expressions: the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424–1445, 2000.
- [6] B. Fasel and J. Luetttin, "Automatic facial expression analysis: a survey," *Pattern Recognition*, vol. 36, no. 1, pp. 259–275, 2003.
- [7] M. Pantic and M. S. Bartlett, *Machine Analysis of Facial Expressions, Face Recognition*, I-Tech Education and Publishing, 2007.
- [8] C. Darwin, *The Expression of Emotions in Man and Animals*, John Murray, London, UK, 1965, reprinted by University of Chicago Press.
- [9] D. Keltner and P. Ekman, "Facial expression of emotion," in *Handbook of Emotions*, M. Lewis and J. M. Haviland-Jones, Eds., pp. 236–249, Guilford, New York, NY, USA, 2000.
- [10] P. Ekman and W. V. Friesen, *The Facial Action Coding System: A Technique for the Measurement of Facial Movement*, Consulting Psychologists Press, 1978.
- [11] G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Classifying facial actions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 974–989, 1999.
- [12] Y. Yacoob and L. Davis, "Recognizing faces showing expressions," in *Proceedings of the International Workshop on Automatic Face and Gesture Recognition*, pp. 278–283, 1995.
- [13] M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Measuring facial expressions by computer image analysis," *Psychophysiology*, vol. 36, no. 2, pp. 253–263, 1999.
- [14] I. A. Essa and A. P. Pentland, "Coding, analysis, interpretation, and recognition of facial expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 757–763, 1997.
- [15] T. Xiang, M. K. H. Leung, and S. Y. Cho, "Expression recognition using fuzzy spatio-temporal modeling," *Pattern Recognition*, vol. 41, no. 1, pp. 204–216, 2008.
- [16] X. W. Chen and T. Huang, "Facial expression recognition: a clustering-based approach," *Pattern Recognition Letters*, vol. 24, no. 9–10, pp. 1295–1302, 2003.
- [17] R. Zhi and Q. Ruan, "Facial expression recognition based on two-dimensional discriminant locality preserving projections," *Neurocomputing*, vol. 71, no. 7–9, pp. 1730–1734, 2008.
- [18] F. Y. Shin, C. F. Chuang, and P. S. P. Wang, "Performance comparisons of facial expression recognition in JAFFE database," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 22, no. 3, pp. 445–459, 2008.
- [19] X. Feng, M. Pietikäinen, and A. Hadid, "Facial expression recognition based on local binary patterns," *Pattern Recognition and Image Analysis*, vol. 17, no. 4, pp. 592–598, 2007.
- [20] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 511–518, 2001.
- [21] Y. Xu and Z. Jin, "Down-sampling face images and low-resolution face recognition," in *Proceedings of the 3rd International Conference on Innovative Computing Information and Control (ICICIC '08)*, p. 392, 2008.
- [22] J. G. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 7, pp. 1169–1179, 1988.
- [23] C. C. Tsai, J. Taur, and C. W. Tao, "Iris recognition based on relative variation analysis with feature selection," *Optical Engineering*, vol. 47, no. 9, Article ID 097202, 2008.
- [24] B. Zhang, Z. Wang, and B. Zhong, "Kernel learning of histogram of local Gabor phase patterns for face recognition," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, Article ID 469109, 8 pages, 2008.
- [25] D. H. Liu, K. M. Lam, and L. S. Shen, "Optimal sampling of Gabor features for face recognition," *Pattern Recognition Letters*, vol. 25, no. 2, pp. 267–276, 2004.
- [26] G. Littlewort, M. S. Bartlett, I. Fasel, J. Susskind, and J. Movellan, "Dynamics of facial expression extracted automatically from video," *Image and Vision Computing*, vol. 24, no. 6, pp. 615–625, 2006.
- [27] L. Shen and L. Bai, "Information theory for Gabor feature selection for face recognition," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 30274, 11 pages, 2006.
- [28] L. Shen, L. Bai, D. Bardsley, and Y. Wang, "Gabor feature selection for face recognition using improved AdaBoost learning," in *Proceedings of the International Workshop on Biometric Recognition Systems (IWBR '05)*, vol. 3781 of *Lecture Notes in Computer Science*, pp. 39–49, Beijing, China, October 2005.
- [29] P. Yang, Q. Liu, and D. N. Metaxas, "Boosting encoded dynamic features for facial expression recognition," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 132–139, 2009.
- [30] J. Lu, K. N. Plataniotis, A. N. Venetsanopoulos, and S. Z. Li, "Ensemble-based discriminant learning with boosting for face recognition," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 166–178, 2006.
- [31] L.-F. Chen, H.-Y. M. Liao, M.-T. Ko, J.-C. Lin, and G.-J. Yu, "New LDA-based face recognition system which can solve the small sample size problem," *Pattern Recognition*, vol. 33, no. 10, pp. 1713–1726, 2000.

- [32] J. H. Friedman, "Regularized discriminant analysis," *Journal of the American Statistical Association*, vol. 84, no. 405, pp. 165–175, 1989.
- [33] A. Hoerl and R. Kennard, "Ridge regression: biased estimation for non-orthogonal problems," *Technometrics*, vol. 12, no. 3, pp. 55–67, 1970.
- [34] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.
- [35] H. Yu and J. Yang, "A direct LDA algorithm for high-dimensional data with application to face recognition," *Pattern Recognition*, vol. 34, pp. 2067–2070, 2001.
- [36] L. Zhao, G. Zhuang, and X. Xu, "Facial expression recognition based on PCA and NMF," in *Proceedings of the World Congress on Intelligent Control and Automation (WCICA '08)*, pp. 6822–6825, 2008.
- [37] X. X. Qi and W. Jiang, "Application of wavelet energy feature in facial expression recognition," in *Proceedings of the IEEE International Workshop on Anti-Counterfeiting, Security, Identification (ASID '07)*, pp. 169–174, 2007.
- [38] W. Liejun, Q. Xizhong, and Z. Taiyi, "Facial expression recognition using improved support vector machine by modifying kernels," *Information Technology Journal*, vol. 8, no. 4, pp. 595–599, 2009.