Emotion Recognition from Arbitrary View Facial Images

Wenming Zheng[1], Hao Tang[2], Zhouchen Lin[3], and Thomas S. Huang[2]

[1]Research Center for Learning Science, Southeast University, Nanjing 210096, China.
wenming_zheng@seu.edu.cn
[2]Beckman Institute, University of Illinois at Urbana-Champaign, USA
haotang2@uiuc.edu, huang@ifp.uiuc.edu
[3]Visual Computing Group, Microsoft Research Asia, China
zhoulin@microsoft.com

Abstract. Emotion recognition from facial images is a very active research topic in human computer interaction (HCI). However, most of the previous approaches only focus on the frontal or nearly frontal view facial images. In contrast to the frontal/nearly-frontal view images, emotion recognition from non-frontal view or even arbitrary view facial images is much more difficult yet of more practical utility. To handle the emotion recognition problem from arbitrary view facial images, in this paper we propose a novel method based on the regional covariance matrix (RCM) representation of facial images. We also develop a new discriminant analysis theory, aiming at reducing the dimensionality of the facial feature vectors while preserving the most discriminative information, by minimizing an estimated multiclass Bayes error derived under the Gaussian mixture model (GMM). We further propose an efficient algorithm to solve the optimal discriminant vectors of the proposed discriminant analysis method. We render thousands of multi-view 2D facial images from the BU-3DFE database and conduct extensive experiments on the generated database to demonstrate the effectiveness of the proposed method. It is worth noting that our method does not require face alignment or facial landmark points localization, making it very attractive.

1 Introduction

The research on human's emotion can be traced back to the Darwin's pioneer work in [1] and since then has attracted a lot of researchers to this area. According to Ekman et al. [2], there are six basic emotions that are universal to human beings, namely, angry (AN), disgust (DI), fear (FE), happy (HA), sad (SA), and surprise (SU), and these basic emotions can be recognized from human's facial expression. Nowadays, the recognition of these six basic emotions from human's facial expressions has become a very active research topic in human computer interaction (HCI). During the past decades, various methods have been proposed for emotion recognition. One may refer to [3][4][5][6] for a survey.

Although emotion recognition has been extensively explored in the past decades, most of the previous approaches focus on the frontal or nearly frontal view facial images. But actually emotion recognition from non-frontal view or even arbitrary view facial images is of more practical utility. However, recognizing the non-frontal view emotions is very difficult. To the best of our knowledge, only a few papers address this issue [7][8][9][10][11][12][14]. In [12], Hu et al. investigated the facial expression recognition problem on a set of images with five yaw views, i.e., 0^o, 30^o, 45^o, 60^o, and 90^o, which are generated from the BU-3DFE database [13]. They used the geometric features defined on the landmark points around the eyes, eye-brow and mouth to represent the face images and then conducted the emotion recognition with various classifiers. Instead of using geometric features, Zheng et al. [14] used sparse SIFT features [15] extracted at 83 landmark points to represent the facial images. They also proposed a novel feature extraction method, based on an upper bound of the multi-class Bayes error under the Gaussian assumption, to reduce the dimensionality of the feature vectors. However, a common limitation of both methods is that the landmark points are known apriori from the original 3D face models. This may severely limit their practical applications, where no 3D face model is available. Moreover, the effectiveness of both methods is only evaluated using facial images in limited views, i.e., five yaw views. In practice, one may encounter much more different views in emotion recognition. In addition, the assumption of Gaussian distribution for each emotion category in [14] may not suffice for the true distributions of the data.

In this paper, we address the emotion recognition problem from arbitrary view facial images. To this end, we propose a novel facial image representation method, which enables us to avoid the face alignment or facial feature localization. The basic idea of the proposed image representation method is to use the region covariance matrix (RCM) [16][17] of the facial region. More specifically, we first detect the facial region from a given facial image [18], then extract a set of dense SIFT feature vectors from each facial image. The concept of dense SIFT feature vectors is illustrated in Fig.1, where the whole facial region is divided into some patches, and at the center of each patch we extract a 128-dimensional SIFT feature vector. The RCM of the facial region is then obtained by computing the covariance of the SIFT vectors. However, it should be noted that, as the dimensionality of the SIFT vectors is 128, the number of entries to be estimated in RCM may be much larger than the number of SIFT feature vectors extracted from each facial image. On the other hand, since the SIFT features are extracted from arbitrary view facial images, they may carry much information that are irrelevant to the emotion recognition. Therefore, extracting the most discriminative features from the raw SIFT feature vectors is advantageous and necessary for improving the recognition performance.

Recall that in [14], Zheng et al. propose a discriminative feature extraction method based on an estimated Bayes error using the Gaussian distributions. However, when the samples, i.e., the SIFT feature vectors, are extracted from arbitrary view facial images, only a single Gaussian may not be enough to accurately model the distribution of the samples. To accurately model the distribution of each basic emotion class, in this paper we instead use mixtures of

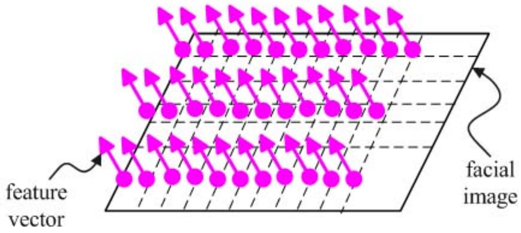

Fig. 1. The whole facial region is divided into some patches, and each patch produces a SIFT feature vector.

Gaussians, rather than a single Gaussian. The Gaussian mixture model (GMM) can be obtained via the expectation-maximization (EM) algorithm [19]. Under the GMM model, we derive a new upper bound of the multi-class Bayes error. Based on this upper bound, we develop a new discriminant analysis method, hereafter called the Bayes discriminant analysis via GMM (BDA/GMM), to reduce the dimensionality of the SIFT feature vectors while preserving the most discriminative information. Moreover, we also propose an efficient algorithm to solve for the optimal discriminant vectors of BDA/GMM.

The rest of this paper is organized as follows. In section 2, we describe the feature representation method. In section 3, we propose our BDA/GMM method. In section 4, we present an efficient algorithm for BDA/GMM. In section 5, we show the emotion classification. The experiments are presented in section 6. Finally section 7 concludes our paper.

2 Feature Representation

2.1 SIFT Feature Descriptor

In [14], Zheng et al. extracted a set of SIFT features at 83 pre-defined landmark points to describe a facial image. Then they concatenated the SIFT features to represent the image and perform classification. Their experiments demonstrated the effectiveness of SIFT features for emotion recognition. In practical applications, however, automatically locating the landmark points from arbitrary view facial image is very challenging. To overcome this problem, we use the so-called dense SIFT features description method illustrated in Fig.1 to describe the facial image, which does not need the face alignment and facial landmark points localization. More specifically, we divide the whole facial region into a set of patches. Then, we extract 128-dimensional SIFT features at the center of each patch. These features are finally used for the calculation of RCM.

2.2 RCM for Facial Image Representation

RCM was originally proposed for image representation and had been successfully applied to face detection, texture recognition, and pedestrian detection [16][17]. RCM can not only capture the statistical properties of the samples, but also be invariant to the image translation, scale and rotation changes. On the other hand, for emotion recognition we may need to integrate the SIFT feature vectors of each image to form a data point and then conduct the classification. Based on the above analysis, we use RCM to represent each facial image in this paper.

However, it should be noted that the entry number of RCM is proportional to the squared dimensionality of the SIFT feature vectors. For example, in this paper the dimensionality of the raw SIFT feature vectors is 128, resulting in (128×128+128)/2=8256 entries to be estimated in RCM. However, the number of SIFT vectors we extract from each facial image is about 450, which is much less than the number of parameters to be estimated in RCM. On the other hand, considering that the SIFT features are extracted from arbitrary view facial images, they may contain much information irrelevant to the emotion recognition. So it will be advantageous and necessary to reduce the dimensionality of the SIFT feature vectors before using the RCM representation. In the next section, we will propose a novel discriminant analysis theory aiming at reducing the dimensionality of the facial feature vectors while preserving the most discriminative information.

3 BDA/GMM: Bayes Discriminant Analysis via Gaussian Mixture Model

In this section, we propose the BDA/GMM method for dimensionality reduction. Let $\mathbf{X}_i = \{\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \cdots, \mathbf{x}_{i,N_i}\} \in \mathbb{R}^d$ $(i = 1, 2, \cdots, c)$ denote the ith class data set, where $\mathbf{x}_{i,j}$ represents the j-th sample of the i-th class, N_i is the number of samples in the i-th class, and c denotes the number of classes.

3.1 Gaussian Mixture Model

Let $p_i(\mathbf{x}|\mathbf{x} \in \mathbf{X}_i)$ denote the class distribution function of $\mathbf{X}_i$. Then the GMM of $p_i(\mathbf{x}|\mathbf{x} \in \mathbf{X}_i)$ can be expressed as follows:

$$p_i(\mathbf{x}|\mathbf{x} \in \mathbf{X}_i) = \sum_{r=1}^{K_i} \pi_{i,r} \mathcal{N}(\mathbf{x}|\mathbf{m}_{i,r}, \mathbf{\Sigma}_{i,r}), \tag{1}$$

where each Gaussian density

$$\mathcal{N}(\mathbf{x}|\mathbf{m}_{i,r}, \mathbf{\Sigma}_{i,r}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\mathbf{\Sigma}_{i,r}|^{\frac{1}{2}}} \exp\left\{ -\frac{1}{2}(\mathbf{x} - \mathbf{m}_{i,r})^T \mathbf{\Sigma}_{i,r}^{-1} (\mathbf{x} - \mathbf{m}_{i,r}) \right\},$$

is called a Gaussian mixture component, the parameters $\pi_{i,r}$ $(0 \leq \pi_{i,r} \leq 1$ and $\sum_{r=1}^{K_i} \pi_{i,r} = 1)$ are called the mixing coefficients, and K_i is the number of Gaussian mixture components. The parameters $\pi_{i,r}$, $\mathbf{m}_{ir}$, and $\mathbf{\Sigma}_{i,r}$ of the GMM in (1) can be estimated via the EM algorithm [19].

3.2 An Upper Bound of Two-class Bayes Error

Let $p_k(\mathbf{x}|\mathbf{x} \in \mathbf{X}_k)$ and P_k be the class distribution density function and the prior probability of the k-th class, respectively. Then the Bayes error between the i-th class and the j-th class can be expressed as [21]:

$$\varepsilon = \int \min \{P_i p_i(\mathbf{x}|\mathbf{x} \in \mathbf{X}_i), P_j p_j(\mathbf{x}|\mathbf{x} \in \mathbf{X}_j)\} \, d\mathbf{x}. \tag{2}$$

Let $\hat{\pi}_{k,q} = P_k \pi_{k,q}$ and $\mathcal{N}_{k,q} = \mathcal{N}(\mathbf{x}|\mathbf{m}_{k,q}, \mathbf{\Sigma}_{k,q})$. Then from (1) we have

$$\begin{aligned} &\min \{P_i p_i(\mathbf{x}|\mathbf{x} \in \mathbf{X}_i), P_j p_j(\mathbf{x}|\mathbf{x} \in \mathbf{X}_j)\} \\ &= \min \left\{ \sum_{r=1}^{K_i} \hat{\pi}_{i,r} \mathcal{N}_{i,r}, \sum_{l=1}^{K_j} \hat{\pi}_{j,l} \mathcal{N}_{j,l} \right\} \le \sum_r \min \left\{ \hat{\pi}_{i,r} \mathcal{N}_{i,r}, \sum_{l=1}^{K_j} \hat{\pi}_{j,l} \mathcal{N}_{j,l} \right\} \\ &\le \sum_r \sum_l \min \{\hat{\pi}_{i,r} \mathcal{N}_{i,r}, \hat{\pi}_{j,l} \mathcal{N}_{j,l}\} \le \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l} \mathcal{N}_{i,r} \mathcal{N}_{j,l}}, \end{aligned} \tag{3}$$

where we have used the inequality $\min(a, b) \le \sqrt{ab}$, $\forall a, b \ge 0$ in the last inequality of (3). By substituting (3) into (2), we have the following upper bound of the Bayes error [21]:

$$\varepsilon \le \varepsilon_{ij} = \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l}} \exp \left(-D_{i,j}^{r,l} \right), \tag{4}$$

where

$$D_{i,j}^{r,l} = \frac{1}{8} (\mathbf{m}_{i,r} - \mathbf{m}_{j,l})^T (\bar{\mathbf{\Sigma}}_{i,j}^{r,l})^{-1} (\mathbf{m}_{i,r} - \mathbf{m}_{j,l}) + \frac{1}{2} \ln \frac{|\bar{\mathbf{\Sigma}}_{i,j}^{r,l}|}{\sqrt{|\mathbf{\Sigma}_{i,r}||\mathbf{\Sigma}_{j,l}|}}, \tag{5}$$

in which $\bar{\mathbf{\Sigma}}_{i,j}^{r,l} = \frac{1}{2}(\mathbf{\Sigma}_{i,r} + \mathbf{\Sigma}_{j,l})$.

Project $\mathbf{x}$ onto a line in direction $\omega \in \mathbb{R}^d$, then the following theorem holds: **Theorem 1.** Let $p_i(\mathbf{x}|\mathbf{x} \in \mathbf{X}_i)$ expressed in (1) denote the distribution function of the i-th class. Then the class distribution function $\tilde{p}_i(\omega^T \mathbf{x}|\mathbf{x} \in \mathbf{X}_i)$ of the projected samples $\omega^T \mathbf{x}$ is also a mixture of Gaussians:

$$\tilde{p}_i(\omega^T \mathbf{x}|\mathbf{x} \in \mathbf{X}_i) = \sum_{r=1}^{K_i} \pi_{i,r} \mathcal{N}(\omega^T \mathbf{x}|\omega^T \mathbf{m}_{i,r}, \omega^T \mathbf{\Sigma}_{i,r} \omega). \tag{6}$$

Proof: See supplementary materials. □
From Theorem 1, equation (5) becomes

$$\tilde{D}_{i,j}^{r,l} = \frac{1}{8} \frac{\left[\omega^T (\mathbf{m}_{i,r} - \mathbf{m}_{j,l})\right]^2}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} + \frac{1}{2} \ln \frac{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega}{\sqrt{(\omega^T \mathbf{\Sigma}_{i,r} \omega)(\omega^T \mathbf{\Sigma}_{j,l} \omega)}}, \tag{7}$$

and the upper bound of the Bayes error in (4) becomes

$$\varepsilon_{ij} = \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l}} \left(\frac{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega}{\sqrt{(\omega^T \mathbf{\Sigma}_{i,r} \omega)(\omega^T \mathbf{\Sigma}_{j,l} \omega)}} \right)^{-\frac{1}{2}} \exp \left\{ -\frac{1}{8} \frac{[\omega^T (\mathbf{m}_{i,r} - \mathbf{m}_{j,l})]^2}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} \right\}. \tag{8}$$

To find a useful upper bound of ε_{ij}, we introduce the following two lemmas:

Lemma 1. Let $f(x) = (1 - x^2)^{\frac{1}{4}}$ $(0 \le x \le 1)$. Then $\hat{f}(x) = \left(\frac{3}{4}\right)^{\frac{1}{4}} \left(\frac{7}{6} - \frac{1}{3}x\right)$ $(0 \le x \le 1)$ is the tightest *linear* upper bound of $f(x)$ in the sense that the total gap $\int_0^1 [\hat{f}(x) - f(x)]\mathrm{d}x$ between them is minimum.

Proof: See supplementary materials. □

Lemma 2. Let $h(x) = \exp(-x)$ $(0 \le x \le a)$. Then $\hat{h}(x) = 1 - \dfrac{1 - \exp(-a)}{a}x$ $(0 \le x \le a)$ is the tightest *linear* upper bound of $h(x)$.

Proof: $h(x)$ is a convex function on the interval $[0, a]$. So the linear function passing through its two ends, $(0, h(0))$ and $(a, h(a))$, is the tightest linear upper bound of $h(x)$. This function is $\hat{h}(x)$. □

From Lemmas 1 and 2, we have:

$$\left(\frac{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega}{\sqrt{(\omega^T \mathbf{\Sigma}_{i,r} \omega)(\omega^T \mathbf{\Sigma}_{j,l} \omega)}} \right)^{-\frac{1}{2}} \le A_0 - A_1 \frac{|\omega^T \mathit{\Delta} \mathbf{\Sigma}_{i,j}^{r,l} \omega|}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega}, \tag{9}$$

$$\exp\left\{ -\frac{1}{8} \frac{[\omega^T(\mathbf{m}_{i,r} - \mathbf{m}_{j,l})]^2}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} \right\} \le 1 - B_{ij} \frac{[\omega^T(\mathbf{m}_{i,r} - \mathbf{m}_{j,l})]^2}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega}, \tag{10}$$

where $A_0 = \left(\frac{3}{4}\right)^{\frac{1}{4}} \frac{7}{6}$, $A_1 = \left(\frac{3}{4}\right)^{\frac{1}{4}} \frac{1}{3}$, $\mathit{\Delta} \mathbf{\Sigma}_{i,j}^{r,l} = \frac{\mathbf{\Sigma}_{i,r} - \mathbf{\Sigma}_{j,l}}{2}$, and $B_{ij} = \frac{1 - e^{-\lambda_{ij}}}{8\lambda_{ij}}$, in which $\lambda_{ij} = \max_\omega \frac{1}{8} \frac{\omega^T \mathbf{B}_{i,j}^{r,l} \omega}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega}$ and $\mathbf{B}_{i,j}^{r,l} = (\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^T$. Applying (9) and (10) to (8), we have

$$\begin{aligned} \varepsilon_{ij} \le & \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l}} \left\{ \left(A_0 - A_1 \frac{|\omega^T \mathit{\Delta} \mathbf{\Sigma}_{i,j}^{r,l} \omega|}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} \right) \right. \\ & \left. - B_{ij} \left[\min_\omega \left(A_0 - A_1 \frac{|\omega^T \mathit{\Delta} \mathbf{\Sigma}_{i,j}^{r,l} \omega|}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} \right) \right] \frac{[\omega^T(\mathbf{m}_{i,r} - \mathbf{m}_{j,l})]^2}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} \right\} \\ = & \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l}} \left(A_0 - A_1 \frac{|\omega^T \mathit{\Delta} \mathbf{\Sigma}_{i,j}^{r,l} \omega|}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} - B_{ij}(A_0 - A_1) \frac{[\omega^T(\mathbf{m}_{i,r} - \mathbf{m}_{j,l})]^2}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} \right), \end{aligned} \tag{11}$$

where we have used the fact that $0 \le \frac{|\omega^T \mathit{\Delta} \mathbf{\Sigma}_{ij} \omega|}{\omega^T \bar{\mathbf{\Sigma}}_{ij} \omega} \le 1$.

3.3 An Upper Bound of Multiclass Bayes Error

For the c classes problem, the Bayes error can be upper bounded as $\varepsilon \le \frac{1}{2} \sum\limits_i \sum_{j \ne i} \varepsilon_{ij}$ [20]. Then, from (11) we obtain that

$$\begin{aligned} \varepsilon \le & \frac{A_0}{2} \sum_i \sum_{j \ne i} \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l}} - \frac{A_1}{2} \sum_i \sum_{j \ne i} \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l}} \frac{\left| \omega^T (\mathit{\Delta} \mathbf{\Sigma}_{i,j}^{r,l}) \omega \right|}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega} \\ & - \frac{B_{\min}(A_0 - A_1)}{2} \sum_i \sum_{j \ne i} \sum_r \sum_l \sqrt{\hat{\pi}_{i,r} \hat{\pi}_{j,l}} \frac{[\omega^T(\mathbf{m}_{i,r} - \mathbf{m}_{j,l})]^2}{\omega^T \bar{\mathbf{\Sigma}}_{i,j}^{r,l} \omega}, \end{aligned} \tag{12}$$

where $B_{\min} = \min_{i,j}\{B_{ij}\} = \frac{1-e^{-\lambda_{\max}}}{8\lambda_{\max}}$ and $\lambda_{\max} = \max_{i,j}\{\lambda_{ij}\}$. Recursively applying the following inequality

$$\frac{a}{b} + \frac{c}{d} \geq \frac{a+c}{b+d},\ \forall a, c \geq 0; b, d > 0 \tag{13}$$

to the error bound in (12), we have the following upper bound of the Bayes error:

$$\begin{aligned}\varepsilon \leq & \frac{A_0}{2}\sum_i\sum_{j\neq i}\sum_r\sum_l\sqrt{\hat{\pi}_{i,r}\hat{\pi}_{j,l}} - \frac{A_1}{2}\frac{\sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}|\omega^T\Delta\mathbf{\Sigma}_{i,j}^{r,l}\omega|}{\sum_i\sum_{j\neq i}\sum_r\sum_l\hat{\pi}_{i,r}\hat{\pi}_{j,l}\omega^T\bar{\mathbf{\Sigma}}_{i,j}^{r,l}\omega} \\ & -\frac{B_{\min}(A_0-A_1)}{2}\frac{\sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}[\omega^T(\mathbf{m}_{i,r}-\mathbf{m}_{j,l})]^2}{\sum_i\sum_{j\neq i}\sum_r\sum_l\hat{\pi}_{i,r}\hat{\pi}_{j,l}\omega^T\bar{\mathbf{\Sigma}}_{i,j}^{r,l}\omega}.\end{aligned} \tag{14}$$

3.4 Our BDA/GMM Method

As the exact value of the Bayes error is hard to evaluate, to minimize the Bayes error, we may minimize its upper bound instead. From (14) we may maximize the following function

$$\begin{aligned}J(\omega) = & \frac{\sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}[\omega^T(\mathbf{m}_{i,r}-\mathbf{m}_{j,l})]^2}{\sum_i\sum_{j\neq i}\sum_r\sum_l\hat{\pi}_{i,r}\hat{\pi}_{j,l}\omega^T\bar{\mathbf{\Sigma}}_{i,j}^{r,l}\omega} \\ & +\frac{A_1}{B_{\min}(A_0-A_1)}\frac{\sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}|\omega^T\Delta\mathbf{\Sigma}_{i,j}^{r,l}\omega|}{\sum_i\sum_{j\neq i}\sum_r\sum_l\hat{\pi}_{i,r}\hat{\pi}_{j,l}\omega^T\bar{\mathbf{\Sigma}}_{i,j}^{r,l}\omega}.\end{aligned} \tag{15}$$

Let

$$\mathbf{B} = \sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}(\mathbf{m}_{i,r}-\mathbf{m}_{j,l})(\mathbf{m}_{i,r}-\mathbf{m}_{j,l})^T$$

and

$$\bar{\mathbf{\Sigma}} = \sum_i\sum_{j\neq i}\sum_r\sum_l\hat{\pi}_{i,r}\hat{\pi}_{j,l}\bar{\mathbf{\Sigma}}_{i,j}^{r,l}.$$

Then we have the following discriminant criterion

$$J(\omega,\mu) = \frac{\omega^T\mathbf{B}\omega}{\omega^T\bar{\mathbf{\Sigma}}\omega} + \mu\frac{\sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}|\omega^T(\mathbf{\Sigma}_{i,r}-\mathbf{\Sigma}_{j,l})\omega|}{\omega^T\bar{\mathbf{\Sigma}}\omega}, \tag{16}$$

where $0 \leq \mu \leq \frac{A_1}{B_{\min}(A_0-A_1)}$ is a parameter to make the upper bound tighter, whose optimal value can be found by cross validation. Based on the above discriminant criterion $J(\omega,\mu)$, we define the optimal discriminant vectors of BDA/GMM as follows [14]:

$$\omega_1 = \arg\max_{\omega} J(\omega,\mu), \quad \text{and} \quad \omega_k = \arg\max_{\substack{\omega^T\bar{\mathbf{\Sigma}}\omega_j=0,\\ j=1,\cdots,k-1}} J(\omega,\mu), \quad (k>1). \tag{17}$$

4 An Efficient Algorithm for BDA/GMM

Let $\omega = \bar{\mathbf{\Sigma}}^{-\frac{1}{2}}\alpha$, $\hat{\mathbf{\Sigma}}_{i,r} = \bar{\mathbf{\Sigma}}^{-\frac{1}{2}}\mathbf{\Sigma}_{i,r}\bar{\mathbf{\Sigma}}^{-\frac{1}{2}}$, $\hat{\mathbf{\Sigma}}_{j,l} = \bar{\mathbf{\Sigma}}^{-\frac{1}{2}}\mathbf{\Sigma}_{j,l}\bar{\mathbf{\Sigma}}^{-\frac{1}{2}}$, and $\hat{\mathbf{B}} = \bar{\mathbf{\Sigma}}^{-\frac{1}{2}}\mathbf{B}\bar{\mathbf{\Sigma}}^{-\frac{1}{2}}$. Then the optimization problem (17) becomes:

$$\alpha_1 = \arg\max_{\alpha} \hat{J}(\alpha,\mu), \quad \text{and} \quad \alpha_k = \arg\max_{\alpha^T\mathbf{U}_{k-1}=\mathbf{0}} \hat{J}(\alpha,\mu), \tag{18}$$

where

$$\mathbf{U}_{k-1} = [\bar{\mathbf{\Sigma}}^{-1}\alpha_1, \bar{\mathbf{\Sigma}}^{-1}\alpha_2, \cdots, \bar{\mathbf{\Sigma}}^{-1}\alpha_{k-1}] \quad \text{and}$$

$$\hat{J}(\alpha,\mu) = \frac{\alpha^T\hat{\mathbf{B}}\alpha}{\alpha^T\alpha} + \mu\frac{\sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}|\alpha^T(\hat{\mathbf{\Sigma}}_{i,r}-\hat{\mathbf{\Sigma}}_{j,l})\alpha|}{\alpha^T\alpha}.$$

Let $K = \max\{K_i | i = 1,2,\cdots,c\}$, $\mathbf{S} = (\mathbf{S})_{c\times c\times K\times K}$ be a $c\times c\times K\times K$ sign tensor whose elements $(\mathbf{S})_{ijrl} = s_{ijrl} \in \{+1,-1\}$, and $\mathbf{\Omega} = \{\mathbf{S}|(\mathbf{S})_{ijrl} \in \{+1,-1\}\}$ denote the set of sign tensors. Further define

$$\mathbf{T}(\mathbf{S},\mu) = \hat{\mathbf{B}} + \mu\sum_i\sum_{j\neq i}\sum_r\sum_l(\hat{\pi}_{i,r}\hat{\pi}_{j,l})^{\frac{3}{2}}s_{ijrl}(\hat{\mathbf{\Sigma}}_{i,r}-\hat{\mathbf{\Sigma}}_{j,l}).$$

Then we have

$$\hat{J}(\alpha,\mu) = \max_{\mathbf{S}\in\mathbf{\Omega}}\frac{\alpha^T\mathbf{T}(\mathbf{S},\mu)\alpha}{\alpha^T\alpha}. \tag{19}$$

From (18) and (19), the optimal vectors α_i in (18) can be expressed as

$$\begin{aligned}\alpha_1 &= \arg\max_{\mathbf{S}\in\mathbf{\Omega}}\max_{\alpha}\frac{\alpha^T\mathbf{T}(\mathbf{S},\mu)\alpha}{\alpha^T\alpha},\\ &\cdots\\ \alpha_k &= \arg\max_{\mathbf{S}\in\mathbf{\Omega}}\max_{\alpha^T\mathbf{U}_{k-1}=\mathbf{0}}\frac{\alpha^T\mathbf{T}(\mathbf{S},\mu)\alpha}{\alpha^T\alpha}.\end{aligned} \tag{20}$$

Suppose that the sign tensor $\mathbf{S}$ is fixed, then the first vector α_1 in (20) is the eigenvector associated with the largest eigenvalue of $\mathbf{T}(\mathbf{S},\mu)$. The principal eigenvector of a matrix can be efficiently computed via the power iteration approach [22]. Suppose that we have obtained the first k vectors $\alpha_1,\cdots,\alpha_k$. Then the $(k+1)$-th vector α_{k+1} can be solved thanks to the following theorem [14]:
Theorem 2. Let $\mathbf{Q}_r\mathbf{R}_r$ be the QR decomposition of $\mathbf{U}_r$, where $\mathbf{R}$ is an $r\times r$ upper triangular matrix. Then α_{r+1} defined in (20) is the principal eigenvector corresponding to the largest eigenvalue of the following matrix $(\mathbf{I}_d - \mathbf{Q}_r\mathbf{Q}_r^T)\mathbf{T}(\mathbf{S},\mu)(\mathbf{I}_d - \mathbf{Q}_r\mathbf{Q}_r^T)$.

In [14], Zheng et al. proposed a greedy search approach to solve the suboptimal solution to a similar optimization problem as (20), where each element of $\mathbf{S}$ should be checked at least once in each iteration of finding the suboptimal vectors. Consequently, the computation cost would increase drastically when the number of Gaussian mixture components grows. To reduce the computational

```
Algorithm 1: Solution method for $\omega_i$ $(i = 1, 2, \cdots, k)$
Input:
  – GMM parameters $\mathbf{m}_{i,r}$ $(i = 1, \cdots, c)$ and $\mathbf{\Sigma}_{i,r}$, $\hat{\pi}_{i,r}$, and $K_i$. Parameter $\mu$.
Initialization:
  1. Compute matrices $\bar{\mathbf{\Sigma}}$ and $\mathbf{B}$; Perform SVD of $\bar{\mathbf{\Sigma}}$: $\bar{\mathbf{\Sigma}} = \mathbf{U}\Lambda\mathbf{U}^T$, compute $\bar{\mathbf{\Sigma}}^{-\frac{1}{2}} = \mathbf{U}\Lambda^{-\frac{1}{2}}\mathbf{U}^T$ and $\bar{\mathbf{\Sigma}}^{-1} = \mathbf{U}\Lambda^{-1}\mathbf{U}^T$, $\hat{\mathbf{\Sigma}}_{i,r} = \bar{\mathbf{\Sigma}}^{-\frac{1}{2}}\mathbf{\Sigma}_{i,r}\bar{\mathbf{\Sigma}}^{-\frac{1}{2}}$, $\hat{\mathbf{B}} = \bar{\mathbf{\Sigma}}^{-\frac{1}{2}}\mathbf{B}\bar{\mathbf{\Sigma}}^{-\frac{1}{2}}$;
For $i = 1, 2, \cdots, k$, Do
  1. Set $\mathbf{S} \leftarrow \text{ones}(c, c, K, K)$, where $K = \max\{K_i | i = 1, \cdots, c\}$, $\mathbf{S}_1 \leftarrow \mathbf{S}$;
  2. Solve the principal eigenvector of $\hat{\mathbf{B}}\alpha_i = \lambda\alpha_i$ via the power method;
  3. Set $(\mathbf{S}_1)_{ijlr} \leftarrow \text{sign}({\alpha_i}^T(\hat{\mathbf{\Sigma}}_{i,r} - \hat{\mathbf{\Sigma}}_{j,l})\alpha_i)$;
  4. While $\mathbf{S} \neq \mathbf{S}_1$, Do
     (a) Set $\mathbf{S} \leftarrow \mathbf{S}_1$;
     (b) Compute $\mathbf{T}(\mathbf{S}, \mu) = \hat{\mathbf{B}} + \mu\sum_i\sum_{j\neq i}\sum_r\sum_l s_{ijrl}(\pi_{i,r}\pi_{j,l})^{\frac{3}{2}}(\hat{\mathbf{\Sigma}}_{i,r} - \hat{\mathbf{\Sigma}}_{j,l})$ and solve the principal eigenvector of $\mathbf{T}(\mathbf{S}, \mu)\alpha_i = \lambda\alpha_i$ via the power method;
     (c) Set $(\mathbf{S}_1)_{ijlr} \leftarrow \text{sign}({\alpha_i}^T(\hat{\mathbf{\Sigma}}_{i,r} - \hat{\mathbf{\Sigma}}_{j,l})\alpha_i)$;
  5. If $i = 1$, $\mathbf{q}_i \leftarrow \alpha_i$, $\mathbf{q}_i \leftarrow \mathbf{q}_i/\|\mathbf{q}_i\|$, and $\mathbf{Q}_1 \leftarrow \mathbf{q}_i$;
     else $\mathbf{q}_i \leftarrow \alpha_i - \mathbf{Q}_{i-1}(\mathbf{Q}_{i-1}^T\alpha_i)$, $\mathbf{q}_i \leftarrow \mathbf{q}_i/\|\mathbf{q}_i\|$, and $\mathbf{Q}_i \leftarrow (\mathbf{Q}_{i-1} \quad \mathbf{q}_i)$;
  6. Compute $\hat{\mathbf{\Sigma}}_{p,q} \leftarrow \hat{\mathbf{\Sigma}}_{p,q} - (\hat{\mathbf{\Sigma}}_{p,q}\mathbf{q}_i)\mathbf{q}_i^T - \mathbf{q}_i(\mathbf{q}_i^T\hat{\mathbf{\Sigma}}_{p,q}) + \mathbf{q}_i(\mathbf{q}_i^T\hat{\mathbf{\Sigma}}_{p,q}\mathbf{q}_i)\mathbf{q}_i^T$ $(p = 1, \cdots, c;\ q = 1, \cdots, K_p\ )$;
  7. Compute $\hat{\mathbf{B}} \leftarrow \hat{\mathbf{B}} - \hat{\mathbf{B}}\mathbf{q}_i\mathbf{q}_i^T - \mathbf{q}_i(\mathbf{q}_i^T\hat{\mathbf{B}}) + \mathbf{q}_i(\mathbf{q}_i^T\hat{\mathbf{B}}\mathbf{q}_i)\mathbf{q}_i^T$
Output:
  – $\omega_i = \frac{1}{\sqrt{\alpha_i^T\bar{\mathbf{\Sigma}}^{-1}\alpha_i}}\bar{\mathbf{\Sigma}}^{-\frac{1}{2}}\alpha_i$, $i = 1, 2, \cdots, k$.
```

cost, here we propose a much more efficient algorithm to find the suboptimal solutions to (20). To this end, we introduce the following definition:

Definition 1: Let $\mathbf{S}_1$ and $\mathbf{S}_2$ be two sign tensors and α_1 and α_2 be the principal eigenvectors of $\mathbf{T}(\mathbf{S}_1, \mu)$ and $\mathbf{T}(\mathbf{S}_2, \mu)$, respectively. If $\alpha_2^T\mathbf{T}(\mathbf{S}_2, \mu)\alpha_2 > \alpha_1^T\mathbf{T}(\mathbf{S}_1, \mu)\alpha_1$, then we say that $\mathbf{S}_2$ is better than $\mathbf{S}_1$.

According to Definition 1, solving the optimal solution in (20) boils down to finding the best sign tensor $\mathbf{S}$. Then we have the following theorem:

Theorem 3. Suppose that $\alpha^{(1)}$ is the principal eigenvector of $\mathbf{T}(\mathbf{S}_1, \mu)$ and $\mathbf{S}_2$ is defined as $(\mathbf{S}_2)_{ijrl} = \text{sign}({\alpha^{(1)}}^T(\hat{\mathbf{\Sigma}}_{i,r} - \hat{\mathbf{\Sigma}}_{j,l})\alpha^{(1)})$. Then $\mathbf{S}_2$ is better than $\mathbf{S}_1$.

Proof: See supplementary materials. □

Thanks to Theorem 3, we are able to improve the sign tensor step by step. We give the pseudo-code of solving k most discriminant vectors of our BDA/GMM method in Algorithm 1.

5 Classification

Suppose that $\mathbf{f}_p$ $(p \in I)$ are the raw SIFT feature vectors extracted from an image $\mathcal{F}$ using the method described in section 2, where I denotes the center positions of the patches in $\mathcal{F}$. Let $\mathbf{W} = [\omega_1, \omega_2, \cdots, \omega_k]$ and $\mathbf{g}_p = \mathbf{W}^T \mathbf{f}_p \in \mathbb{R}^k$ be the projected feature vectors of $\mathbf{f}_p$ onto $\mathbf{W}$. Let $\mathbf{M}_{\mathrm{COV}}$ denote the covariance matrix of the feature vectors $\{\mathbf{g}_p | p \in I\}$. Since $\mathbf{M}_{\mathrm{COV}}$ is a symmetric matrix, we concatenate the elements in the upper triangular part of $\mathbf{M}_{\mathrm{COV}}$ into a vector $\mathbf{v}_{\mathrm{COV}}$. Then we have the final feature vector $\mathbf{v} = \mathbf{v}_{\mathrm{COV}} / \|\mathbf{v}_{\mathrm{COV}}\|$ after normalizing $\mathbf{v}_{\mathrm{COV}}$. Now we can train a classifier, e.g., the support vector machine (SVM) [19], Adaboost [23], or simply the linear classifier [21], using all the vectors $\mathbf{v}$. For a test facial image, we use the same method to obtain the corresponding vector $\mathbf{v}_{\mathrm{test}}$, and then classify it using the trained classifier. In this paper, we choose the linear classifier for our emotion recognition task.

6 Experiments

In this section, we conduct experiments to demonstrate the effectiveness of the proposed method. Since no facial expression database with arbitrary view facial images is available, we conduct our experiments on the facial images generated from the BU-3DFE database [13]. More specifically, by projecting the 3D facial expression models in the BU-3DFE database in various directions, we can generate a set of 2D facial images with various facial views. The BU-3DFE database consists of 3D facial expression models of 100 subjects (56 female and 44 male). For each subject, there are 6 basic emotions with 4 levels of intensities. In our experiments, we only choose the 3D models with the highest level of intensity to generate 35 facial images corresponding to 35 projection directions, i.e., seven yaw angles (-45^o, -30^o, -15^o, 0^o, $+15^o$, $+30^o$, and $+45^o$) and five pitch angles (-30^o, -15^o, 0^o, $+15^o$, and $+30^o$). Consequently, we have $100 \times 6 \times 5 \times 7 = 21000$ facial images in total for our experiments. Fig. 2 shows some examples of the generated face images.

We adopt a five-fold cross validation strategy [21] to conduct the experiments. More specifically, we randomly divide the 100 subjects into five groups, each one having 20 subjects. In each trail of the experiment, we choose one group as test set and the other ones as training set. We conduct five trials of the experiment in total such that each subject is used as test data once. For all the experiments, we fit the GMM with 5 different numbers, i.e., 16, 32, 64, 128, and 256, of Gaussian mixture components, and for each choice of the number of Gaussian mixture components, we apply our BDA/GMM algorithm to reduce the dimensionality of the SIFT feature vectors from 128 to 30. The parameter μ in the discriminant criterion (16) is simply fixed at $\mu = 0.5$ in all the experiments. Note that a better choice of its value may result in better performance.

Table 1 summarizes the experimental results of the overall error rates as well as the error rates of each emotion with different numbers of Gaussian mixture components. Fig.3 shows the overall confusion matrix of recognizing the six basic

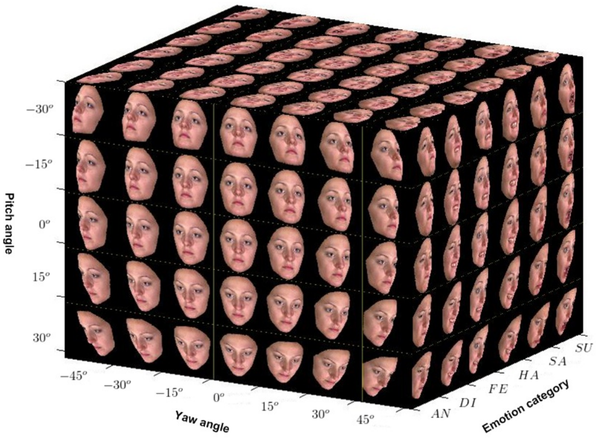

Fig. 2. Some facial images rendered from the BU-3DFE database, covering the facial images of six basic emotions, seven yaw angles, and five pitch angles.

Table 1. The overall error rates (%) of the proposed method under different numbers of Gaussian mixture components.

mixture #	16	32	64	128	256
AN	43.51	43.46	42.37	43.23	42.60
DI	31.71	32.20	32.60	32.00	31.89
FE	45.06	44.60	46.49	45.17	44.89
HA	16.74	16.20	17.34	15.60	16.57
SA	44.31	43.80	41.11	41.03	42.09
SU	14.57	14.29	13.26	13.31	12.57
Ave	32.65	32.42	32.20	**31.72**	31.77

emotions, in which 256 Gaussian mixture components are used. From Table 1, one can see that the lowest error rate is 31.72%, achieved when 128 Gaussian mixture components are used. We can also see from Table 1 and Fig.3 that the emotions easiest to be recognized are happy and surprise, and the remaining emotions are more difficult.

Table 2 shows the overall error rates of the proposed method across various facial views when 256 Gaussian mixture components are used. In Table 2, each row of the table represents the overall error rates of different pitch angles (from -30^o to $+30^o$), while each column represents the overall error rates of different yaw angles (from -45^o to $+45^o$). From Table 2, one can clearly see that both yaw angles and pitch angles can affect the emotion recognition performance, where the best results are achieved when the facial images are frontal or near frontal.

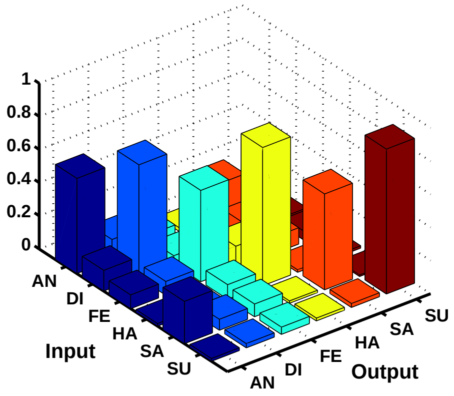

Fig. 3. The overall confusion matrix of the proposed method, where 256 Gaussian mixture components are used.

Table 2. Average error rates (%) of different emotions versus different views using our method, where 256 Gaussian mixture components are used.

	-30^o	-15^o	0^o	$+15^o$	$+30^o$	Ave
-45^o	39.67	35.67	31.00	33.00	43.00	36.47
-30^o	30.67	28.33	27.67	28.50	38.50	30.73
-15^o	28.33	29.17	25.83	25.83	33.17	28.47
0^o	30.83	27.83	**25.17**	25.67	31.83	**28.27**
$+15^o$	32.33	29.33	26.33	28.50	32.00	29.70
$+30^o$	32.33	29.33	29.33	32.67	35.50	31.83
$+45^o$	40.17	33.50	31.33	35.83	43.67	36.90
Ave	33.48	30.45	**28.10**	30.00	36.81	31.77

As there are no other methods proposed for *arbitrary view* emotion recognition, we can only provide our own experimental results. Nevertheless, for comparison we also provide the results of two approaches. One is to use the linear discriminant analysis (LDA) to replace our BDA/GMM method for reducing the dimensionality of the SIFT feature vectors, and the other one is to replace the Gaussian mixtures in our BDA/GMM method with single Gaussian, denoted by BDA/Gaussian, to model each class (i.e., a view is a class) and then repeat the rest procedures in our paper, where the remaining experimental settings in both approaches are the same as those for our BDA/GMM. Fig.4 presents the overall error rates of the three methods. From Fig.4, one can clearly see that our BDA/GMM method achieves much better results than the LDA.

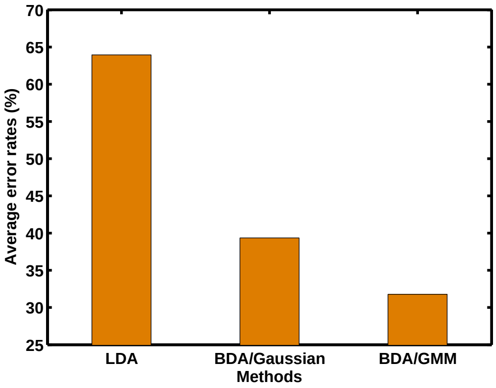

Fig. 4. Average error rate comparisons among LDA, BDA/Gaussian, and BDA/GMM.

7 Conclusions

In this paper we have proposed a new method to address the emotion recognition problem from arbitrary view facial images. A major advantage of this method is that it does not need face alignment or facial landmark points localization from arbitrary view facial images, both of which are very challenging. As an important part of our emotion recognition system, a novel discriminant analysis theory, called the BDA/GMM, is also developed. This new discriminant analysis theory is derived by minimizing a new upper bound of the Bayes error which is derived using the Gaussian mixture model. The proposed method is tested on a lot of facial images with various views, generated from 3D facial expression models in the BU-3DFE database. The experimental results show that our method can achieve a satisfactory recognition performance.

It is worth noting that, although having been proven to be an effective image representation method, the RCM representation may also discard some useful discriminant information, e.g., the class means of samples. Therefore, finding a better image representation method may help to improve the performance of emotion recognition. This will be one of our future work. We will also investigate whether a more advanced classifier, e.g., SVM [19] and Adaboost [23], can greatly improve the recognition performance.

Acknowledgment

This work was partly supported by NSF of China under Grants 60503023 and 60872160, and partly by the science technology foundation of Southeast University of China under the grant XJ2008320.

References

1. Darwin C.: The expression of the emotions in man and animals. London: John Murray (1872)
2. Ekman P., Friesen W.V.: Pictures of facial affect. in Human Interaction Laboratory, San Francisco, CA: Univ. California Medical Center (1976)
3. Fasel B., Luettin J.: Automatic facial expression analysis: a survey. Pattern Recognition, 36, 259-275 (2008)
4. Tian Y.L., Kanade T., Cohn J.F.: Facial expression analysis. In: S.Z. Li, A.K. Jain (eds.), Handbook of Facial Recognition, Springer-verlag (2004)
5. Pantic M., Rothkrantz L.J.M.: Automatic analysis of facial expressions: the state of the art. IEEE Trans. on PAMI, 22(12), 1424-1445 (2000)
6. Zeng Z., Pantic M., Roisman G.I., Huang T.S.: A survey of affect recognition methods: audio, visual, and spontaneous expressions. IEEE Trans. on PAMI, 31(1), 39-58 (2009)
7. Pantic M., Patras I.: Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences. IEEE Trans. on SMC - Part B, 36(2), 433-449 (2006)
8. Hu Y., Zeng Z., Yin L., Wei X., Zhou X.,Huang T.S.: Multi-view facial expression recognition. Int. Conf. on Automatic Face and Gesture Recognition (2008)
9. Moore S., Bowden R.: The effect of pose on facial expresssion recognition. British Machine Vision Conference (2009)
10. Kumano S., Otsuka K., Yamato J., Maeda E.,Sato Y.: Pose-invariant facial expression recognition using variable-intensity templates. Int. J. of Comput. Vision (2009)
11. Sajama, Orlitsky A.: Supervised dimensionality reduction using mixture models. ICML (2005)
12. Hu Y., Zeng Z., Yin L., Wei X., Tu J., Huang T.S.: A study of non-frontal-view facial expressions recognition. Proceedings of ICPR, pp. 1-4 (2008)
13. Yin L., Wei X., Sun Y., Wang J., Rosato M.J.: A 3D facial expression database for facial behavior research. Proceedings of 7th Int. Conf. on Automatic Face and Gesture Recognition, 211-216 (2006)
14. Zheng W., Tang H., Lin Z., Huang T.S.: A novel approach to expression recognition from non-frontal face images. Proceedings of IEEE ICCV, 1901-1908 (2009)
15. Lowe D.G.: Distinctive image features from scale-invariant keypoints. Int. J. of Comput. Vision, 60(2), 91-110 (2004)
16. Tuzel O., Porikli F., Meer P.: Region covariance: a fast descriptor for detection and classification. Proc. of ECCV, 589-600 (2006)
17. Tuzel O., Porikli F., Meer P.: Pedestrian detection via classification on Riemannian manifolds," IEEE Transs on PAMI. 30(10), 1713-1727 (2008)
18. Viola P., Jones M.J.: Robust real-time face detection. Int. J. of Comput. Vision, 57(2), 137-154 (2004)
19. Bishop C. M.: Pattern Recognition and Machine Learning. Springer (2006)
20. Chu, J.T., Chuen J.C.: Error probability in decision functions for character recognition. J. of the Association for Computing Machinery, 14(2), 273-280 (1967)
21. Fukunaga K.: Introduction to Statistical Pattern Recognition (Second Edition). New York: Academic Press (1990)
22. GolubG., Van C.: Matrix Computations. The Johns Hopkins University Press (1996)
23. Rätsch G., Onoda T., Müller K.-R.: Soft margins for Adaboost. Machine Learning, 1-35 (2000)