Pattern Recognition Letters 29 (2008) 1797–1805

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Expression-invariant face recognition by facial expression transformations

Hyung-Soo Lee, Daijin Kim [*]

Biometrics Engineering Research Center (BERC), Pohang University of Science and Technology, San 31, Hyoja-Dong, Nam-Gu, Pohang 790-784, Republic of Korea

ARTICLE INFO

Article history:
Received 8 December 2006
Received in revised form
17 September 2007
Available online 2 July 2008

Communicated by T. Tan

Keywords:
Active appearance model
Multi-linear model
Facial expression recognition
Facial expression transformation
Relative expression parameters
Expression-invariant face recognition

ABSTRACT

In this paper, we present a method of expression-invariant face recognition that transforms input face image with an arbitrary expression into its corresponding neutral facial expression image. When a new face image with an arbitrary expression is queried, it is represented by a feature vector using the active appearance model (AAM). Then, the facial expression state of the queried feature vector is identified by the facial expression recognizer. Next, the queried feature vector is transformed into the facial expression vector using the identified expression state via *direct* or *indirect* facial expression transformation, where former uses model translation directly to transform the expression, but the latter uses model translation to obtain relative expression parameters: shape difference and appearance ratio and transforms the expression indirectly by the obtained relative expression parameters. Then, the face recognition is performed by the distance-based matching technique, which matches the transformed neutral expression feature vector with the vectors in the gallery, which have only neutral expression. Experimental results show that the proposed expression-invariant face recognition method is very robust for a variety of expressions.

1. Introduction

In modern life, the need for personal security and access control becomes important issue. Biometrics is a technology which is expected to replace traditional authentication methods which is easy to be stolen, forgotten and duplicated. Fingerprints, face, iris, and voiceprints are commonly used biometric features. Among these features, face provides more direct, friendly and convenient identification way and is more acceptable compared with individual identification ways of other biometrics features. Thus, face recognition is one of the most important parts in biometrics.

Since Kanade (1973) attempted automatical face recognition 30 years ago, many researchers have investigated face recognition. Among various face recognition methods, holistic appearance-based approach, which started from the work of Turk and Pentland (1991), seems to have prevailed up to now. However, face appearance varies drastically with changes of pose, illumination, facial expression, and so forth. Such variations make appearance-based face recognition difficult. This paper attempts to overcome the variations of facial expression.

Many previous studies have attempted to recognize individuals across different expressions. Liu et al. (2003) measured two types of the asymmetric facial information, density difference and edge orientation. They showed that this information could obtain individual differences which are stable to the changes of facial expressions. Elad and Kimmel (2001) proposed an efficient isometric transformation framework of a non-rigid object on a manifold. This framework overcame the disadvantage of taking a rigid transformation of an existing isometric transformation by using multidimensional scaling (MDS). Bronstein et al. (2003) proposed a 3D face recognition which was invariant to the facial expression by introducing the isometric-invariant representation of the facial surface. Wang and Ahuja (2003) decomposed facial expression features using a higher-order singular value decomposition (HOSVD) on the expression subspace and performed face recognition and facial expression recognition at the same time in the subspace. On the other hand, some researchers treated face geometry and texture information separately. The separate modelling of geometry and texture information was based on active appearance model (AAM) (Cootes et al., 1998). Li et al. (2006) used AAM to recognize individuals with varying face expression. They fit the AAM to the input image and warped to the reference image frame to remove the geometry information. However, the texture information still included expression features, even though their approach was better than previous ones.

In this paper, we transform arbitrary facial expression images into corresponding neutral facial expression images for expression-invariant face recognition, assuming that gallery images consist of neutral facial expression images. The facial expression decomposition is an appropriate technique for synthesizing new expression images from input expression images with unchanged identity. Abboud and Davoine (2004) used AAM and the bilinear model to synthesize expression images. However, the identity of

[*] Corresponding author. Tel.: +82 54 279 2249; fax: +82 54 279 2299.
E-mail addresses: sooz@postech.ac.kr (H.-S. Lee), dkim@postech.ac.kr (D. Kim).

0167-8655/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2008.05.012

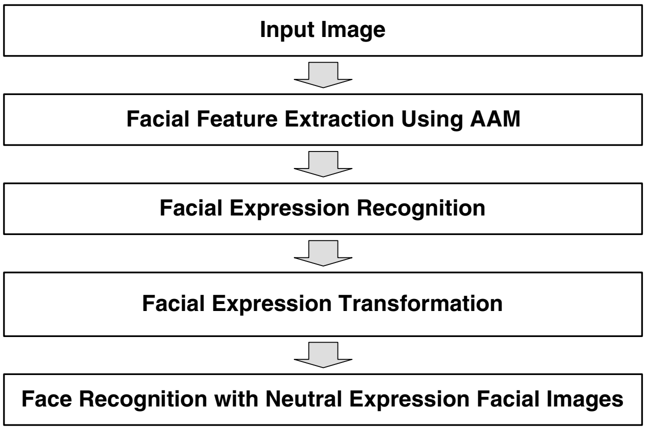
Input Image
Facial Feature Extraction Using AAM
Facial Expression Recognition
Facial Expression Transformation
Face Recognition with Neutral Expression Facial Images

Fig. 1. An overall procedure of the proposed expression-invariant face recognition method.

images synthesized by their method was far different from the ground-truth image, thus their method is not appropriate for face recognition. In our previous work (Lee and Kim, 2006), we adopted the result of Zhou and Lin (2005) to synthesize realistic facial expression images applied it to face recognition. In this paper, we extended our previous work to the multi-linear model.

Fig. 1 shows the overall procedure of our proposed expression-invariant face recognition method. First, we extract the facial feature vector from the input image using AAM. Then, we obtain the expression state of the feature vector from the facial expression recognizer. Then, we transform the input feature vector into its corresponding neutral expression vector using the direct or indirect facial expression transformation. The multi-linear model is used for this transformation. We converted the neutral expression vector into the neutral facial expression image via AAM reconstruction to show the transformation results. Finally, we perform the expression-invariant face recognition by the distance-based matching techniques: nearest neighbor classifier, linear discriminant analysis (LDA) (Etemad and Chellappa, 1997), and generalized discriminant analysis (GDA) (Baudat and Anouar, 2000).

This paper is organized as follows: Section 2 describes the theoretical background of AAM. Section 3 reviews the multi-linear analysis, the model construction and the translation, and the ridge regressive bilinear model. Section 4 describes the proposed expression-invariant face recognition method. Section 5 shows some experimental results and discussion. Finally, Section 6 presents our conclusions.

2. Facial feature extraction using AAM

In the following, we denote scalar by lower-case letters $(a, b, \ldots)$, vectors by bold lower-case letters $(\boldsymbol{a}, \boldsymbol{b}, \ldots)$, matrices by bold upper-case letter $(\boldsymbol{A}, \boldsymbol{B}, \ldots)$, and higher-order tensors by italic letters $(\mathcal{A}, \mathcal{B}, \ldots)$.

The active appearance model (AAM) (Cootes et al., 1998; Matthews and Baker, 2004) treats the shape and 2D appearance of face images separately. The 2D shape of an AAM is defined by the vertex locations of 2D triangulated meshes of face images. Mathematically, we define the shape $\boldsymbol{s}$ of an AAM as the 2D coordinates of the l vertices that consist of the meshes:

$$\boldsymbol{s} = \begin{pmatrix} x_1 & x_2 & \cdots & x_l \\ y_1 & y_2 & \cdots & y_l \end{pmatrix}. \tag{1}$$

AAM allows a linear shape variation, which means that the shape $\boldsymbol{s}$ can be expressed as the sum of the base shape $\boldsymbol{s}_0$ and a linear combination of m orthogonal shape bases $\{\boldsymbol{s}_i,\ i = 1, 2, \ldots, m\}$:

$$\boldsymbol{s} = \boldsymbol{s}_0 + \sum_{i=1}^{m} p_i \boldsymbol{s}_i, \tag{2}$$

where $\boldsymbol{s}_0$, p_i and $\boldsymbol{s}_i$ are the mean shape, the ith shape parameter, and the ith shape eigenvector corresponding to the ith largest eigenvalue, respectively. The shape eigenvectors are obtained by applying principal component analysis (PCA) to the training data set that consists of a set of images with manually land-marked shape vertices.

The 2D appearance of an AAM is defined on the mean shape $\boldsymbol{s}_0$ and its variation is modelled by a sum of the mean appearance A_0 and a linear combination of m orthogonal appearance bases $\{A_i,\ i = 1, 2, \ldots, m\}$:

$$A = A_0 + \sum_{i=1}^{m} \alpha_i A_i, \tag{3}$$

where α_i and A_i are the ith texture parameter and the ith appearance eigenvector, respectively. The appearance eigenvectors are obtained by applying PCA to the training data set that consists of a set of warped images defined on the mean shape $\boldsymbol{s}_0$.

Finally, to control both the shape and appearance of the face model, we define the concatenated feature vector $\boldsymbol{b}$ which is the combination of weighted shape parameter vector and appearance parameter vector:

$$\boldsymbol{b} = \begin{pmatrix} \boldsymbol{\Psi}_s \boldsymbol{p} \\ \alpha \end{pmatrix}, \tag{4}$$

where $\boldsymbol{\Psi}_s$ is a diagonal matrix of weights for each shape parameter. Since the shape and appearance models are constructed independently, the ranges of two parameter vectors are usually different. The weight matrix $\boldsymbol{\Psi}_s$ is used to balance their different numerical ranges. The concatenated feature vector $\boldsymbol{b}$ is also modelled by a linear combination of m orthogonal concatenated feature bases $\{B_i,\ i = 1, 2, \ldots, m\}$:

$$\boldsymbol{b} = \sum_{i=1}^{m} y_i B_i, \tag{5}$$

where y_i and B_i are the ith concatenated feature parameter and the ith concatenated feature eigenvector, respectively. We use the concatenated feature parameter vector $\boldsymbol{y}$ as the input vector of the multi-linear model that will be explained in the following section.

3. Multi-linear model

A tensor is a generalization of vectors and matrices. For example, a vector is a first-order tensor and a matrix is a second-order tensor. This tensor concept allows us to manipulate the quantities of higher-order data.

A tensor of order N is given by $\mathcal{A} \in R^{I_1 \times I_2 \times \cdots \times I_N}$, where N is the order. We can unfold the tensor $\mathcal{A}$ by stacking the mode-n vectors of it as columns in a matrix as $\boldsymbol{A}_{(n)} \in R^{I_n \times (I_1 I_2 \cdots I_{n-1} I_{n+1} \cdots I_N)}$. This tensor unfolding allows easy manipulation of the tensor.

The multiplication of higher-order tensor $\mathcal{A} \in R^{I_1 \times I_2 \times \cdots \times I_N}$ by a matrix $\boldsymbol{M} \in R^{J_n \times I_n}$ is represented as $\mathcal{B} = \mathcal{A} \times_n \boldsymbol{M}$, where $\mathcal{B} \in R^{I_1 \times I_2 \times \cdots \times I_{n-1} \times J_n \times I_{n+1} \times \cdots \times I_N}$, and its entries are given by

$$(\mathcal{A} \times_n \boldsymbol{M})_{i_1 i_2 \cdots i_{n-1} j_n i_{n+1} \cdots i_N} = \sum_{i_n} a_{i_1 \cdots i_{n-1} i_n i_{n+1} \cdots i_N} m_{j_n i_n}. \tag{6}$$

This mode-n product of tensor and matrix can be expressed in unfolded form:

$$\boldsymbol{B}_{(n)} = \boldsymbol{M}\boldsymbol{A}_{(n)}, \tag{7}$$

where $\boldsymbol{A}_{(n)}$ and $\boldsymbol{B}_{(n)}$ are mode-n matrix unfolding of tensor $\mathcal{A}$ and $\mathcal{B}$, respectively.

Using HOSVD, the tensor $\mathscr{A} \in R^{I_1 \times I_2 \times \cdots \times I_N}$ can be decomposed as the mode-n product between N orthogonal mode matrices $\boldsymbol{U}_1 \cdots \boldsymbol{U}_N$ and a core tensor $\mathscr{Z} \in R^{I_1 \times I_2 \times \cdots \times I_N}$:

$$\mathscr{A} = \mathscr{Z} \times_1 \boldsymbol{U}_1 \cdots \times_N \boldsymbol{U}_N. \tag{8}$$

The core tensor $\mathscr{Z}$ is analogous to the diagonal singular value matrix of conventional SVD, but it is a full rank tensor. In addition, it governs the interaction between the mode matrices $\boldsymbol{U}_n$, where $n = 1, \ldots, N$. The mode matrix $\boldsymbol{U}_n$ contains the orthonormal vectors spanning the column space of the matrix $\boldsymbol{A}_{(n)}$. The higher-order singular value decomposition (HOSVD) (Lathauwer et al., 2000) of a tensor $\mathscr{A}$ can be computed as follows:

(1) Set the mode-n matrix $\boldsymbol{U}_n$ by the left singular matrix of the mode-n matrix unfolding of $\boldsymbol{A}_{(n)}$ for $n = 1, \ldots, N$.
(2) Compute the core tensor by

$$\mathscr{Z} = \mathscr{A} \times_1 \boldsymbol{U}_1^{\mathrm{T}} \cdots \times_N \boldsymbol{U}_N^{\mathrm{T}}. \tag{9}$$

3.1. Multi-linear model construction and translation

We construct a third-order tensor $\mathscr{D} \in R^{I \times J \times K}$ to represent the face images, where I, J and K are the number of subjects, the number of facial expressions and the dimension of the input vector, respectively. By applying the HOSVD to the constructed tensor, it is decomposed by three factors as

$$\mathscr{D} = \mathscr{Z} \times_1 \boldsymbol{U}_{\mathrm{id}} \times_2 \boldsymbol{U}_{\mathrm{exp}} \times_3 \boldsymbol{U}_{\mathrm{f}}, \tag{10}$$

where $\mathscr{Z}$ is the core tensor which governs the interaction between the three mode matrices and $\boldsymbol{U}_{\mathrm{id}}$, $\boldsymbol{U}_{\mathrm{exp}}$ and $\boldsymbol{U}_{\mathrm{f}}$ represent the people identity subspace, the facial expression subspace and the facial feature subspace, respectively.

Based on the TensorFaces concept which is proposed by Vasilescu and Terzopoulos (2002), we define a basis tensor

$$\mathscr{B} = \mathscr{Z} \times_2 \boldsymbol{U}_{\mathrm{exp}} \times_3 \boldsymbol{U}_{\mathrm{f}}, \tag{11}$$

which spans all the images irrespective of the facial expression. We can index this basis tensor for a particular expression s to make a basis subtensor as

$$\mathscr{B}_s = \mathscr{Z} \times_2 \boldsymbol{U}_{\mathrm{exp}}^{\mathrm{T}}(s) \times_3 \boldsymbol{U}_{\mathrm{f}}, \tag{12}$$

which spans all the images with a particular expression s, where $\boldsymbol{U}_{\mathrm{exp}}^{\mathrm{T}}(s)$ represent the specific row vector of the expression subspace $\boldsymbol{U}_{\mathrm{exp}}$. We obtain the expression-specific basis matrices by unfolding the subtensors $\mathscr{B}_s$ along the facial feature mode: $\boldsymbol{W}_s = \boldsymbol{B}_{s(f)},\ s = 1, \ldots, m$, where m is a total number of expressions. When the input image has expression s, the identity vector can be computed by a single pseudoinverse operation as

$$\boldsymbol{b} = (\boldsymbol{W}_s)^{\dagger} \boldsymbol{y}, \tag{13}$$

where the symbol † denotes the pseudoinverse operation. Then the model fitted vector $\boldsymbol{y}_s$ is expressed as

$$\boldsymbol{y}_s = \boldsymbol{W}_s \boldsymbol{b} \tag{14}$$

$$\boldsymbol{y}_s = \mathscr{Z} \times_1 \boldsymbol{b}^{\mathrm{T}} \times_2 \boldsymbol{U}_{\mathrm{exp}}^{\mathrm{T}}(s) \times_3 \boldsymbol{U}_{\mathrm{f}}. \tag{15}$$

The multi-linear translation from the input expression s to the neutral expression n is accomplished by changing the expression factor from s to n as

$$\boldsymbol{y}_n = \mathscr{Z} \times_1 \boldsymbol{b}^{\mathrm{T}} \times_2 \boldsymbol{U}_{\mathrm{exp}}^{\mathrm{T}}(n) \times_3 \boldsymbol{U}_{\mathrm{f}}. \tag{16}$$

This translation also can be expressed in a matrix form as

$$\boldsymbol{y}_n = \boldsymbol{W}_n \boldsymbol{b}. \tag{17}$$

3.2. Ridge regressive bilinear model

In our previous work (Shin et al., 2008), we introduced the ridge regressive bilinear model that combined the ridge regression technique into the bilinear model in order to improve the stability of the existing bilinear model. To give the properties of ridge regression to the bilinear model, we modify the objective function of the bilinear model as follows:

$$\boldsymbol{E}(\boldsymbol{a}, \boldsymbol{b}) = (\boldsymbol{y} - (\boldsymbol{Wb})^{\mathrm{VT}} \boldsymbol{a})^{\mathrm{T}} (\boldsymbol{y} - (\boldsymbol{Wb})^{\mathrm{VT}} \boldsymbol{a}) + \lambda \boldsymbol{a}^{\mathrm{T}} \boldsymbol{a} \boldsymbol{b}^{\mathrm{T}} \boldsymbol{b}, \tag{18}$$

where the second term was introduced to constrain the magnitude of the style and content parameter vectors to stabilize the model.

In order to get the minimum value of $\boldsymbol{E}$, we differentiate the objective function $\boldsymbol{E}$ with the model parameter vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ and make it zero. This provides the optimal style and content parameter vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ as

$$\boldsymbol{a} = (((\boldsymbol{Wb})^{\mathrm{VT}})^{\mathrm{T}} ((\boldsymbol{Wb})^{\mathrm{VT}}) + \lambda \boldsymbol{b}^{\mathrm{T}} \boldsymbol{b})^{-1} ((\boldsymbol{Wb})^{\mathrm{VT}})^{\mathrm{T}} \boldsymbol{y}, \tag{19}$$

$$\boldsymbol{b} = ((\boldsymbol{W}^{\mathrm{VT}} \boldsymbol{a})^{\mathrm{T}} (\boldsymbol{W}^{\mathrm{VT}} \boldsymbol{a}) + \lambda \boldsymbol{a}^{\mathrm{T}} \boldsymbol{a})^{-1} (\boldsymbol{W}^{\mathrm{VT}} \boldsymbol{a})^{\mathrm{T}} \boldsymbol{y}. \tag{20}$$

In the asymmetric ridge regressive bilinear model, the content vector can be computed by a single operation as

$$\boldsymbol{b} = (\boldsymbol{W}_s^{\mathrm{T}} \boldsymbol{W}_s + \lambda \boldsymbol{I})^{-1} \boldsymbol{W}_s^{\mathrm{T}} \boldsymbol{y}. \tag{21}$$

Real implementations control the eigenvalues by the following strategy. The eigenvalues are kept for dominant eigenvectors whose eigenvalues are greater than λ, but eigenvalues are changed to λ for non-significant eigenvectors with eigenvalues less than λ.

Since our model translation (Eq. (17)) is exactly the same with the asymmetric bilinear model (Lee and Kim, 2006), we can apply the ridge regression technique to our model translation.

4. Expression-invariant face recognition

Expression-invariant face recognition requires that the face images in the gallery have only neutral facial expressions. Then, when a new face image with an arbitrary facial expression is queried, it is transformed into its corresponding neutral facial expression image. We adopt the tensor model for this facial expression transformation. We assume that the facial expression of the input face image is already known from the facial expression recognition. Then, we transform the input face image with a known facial expression into its corresponding neutral facial expression image by the direct or indirect facial expression transformation, which will be explained later. Then, we perform the expression-invariant face recognition using the transformed neutral facial feature vectors.

4.1. Direct facial expression transformation

The facial expression factor s of the input face image is assumed to be provided by the facial expression recognizer (Sung et al., 2006). This double layered GDA-based method uses a combination of the 2D shape and 2D appearance of the input image to recognize its facial expression. It outputs the facial expression state s, which is happy, surprise, or anger. The direct method transforms an input image with an arbitrary expression into its corresponding neutral expression image as follows.

First, AAM fitting is performed for the input image I and the facial feature vector $\boldsymbol{y}$ is extracted by the method described in Section 2. Second, the facial expression factor s of the input image is obtained by the facial expression recognizer. Third, the identity vector $\boldsymbol{b}_{\mathrm{iden}}$ is computed by a simple operation like Eq. (13) or (21) using the known expression-specific basis matrix $\boldsymbol{W}_s$, where the style factor s is given by the facial expression recognizer.

Fourth, the model fitted vector $\boldsymbol{y}_s$ is translated into its corresponding neutral expression vector $\boldsymbol{y}_n$ by multiplying the neutral expression-specific basis matrix $\boldsymbol{W}_n$ with the identity factor $\boldsymbol{b}_{\text{iden}}$. Finally, the neutral expression image I' is obtained by reconstructing the AAM parameters of $\boldsymbol{y}_n$. Fig. 2 summarizes the overall procedure that transforms the input image with a specific expression into its corresponding neutral expression image.

4.2. Indirect facial expression transformation

Generally, the identity discrepancy between the transformed images and the ground-truth images comes from the fact that the tensor model itself cannot express a new person who is not contained in the training set. To avoid this problem, indirect facial expression transformation is used, which performs the model translation to obtain the relative expression parameters: shape difference and appearance ratio, and transforms the expression indirectly using them.

Zhou and Lin (2005) proposed the relative expression parameters for robust facial expression image synthesis. The basic idea of their approach is as follows. When two persons with the same facial expressions are in the same pose and lighting condition, the shape difference $\Delta\boldsymbol{s}$ and the appearance ratio $R(u,v)$ between two different expressions which are defined as

$$\Delta\boldsymbol{s} = \boldsymbol{s}_n - \boldsymbol{s}_s, \quad R(u,v) = \frac{A_n(u,v)}{A_s(u,v)} \tag{22}$$

are almost identical for both people. Here $\boldsymbol{s}_n$ and $\boldsymbol{s}_s$ are the shape vectors of the neutral and s expression state, A_n and A_s are the appearance images of the neutral and s expression state, and (u,v) are the 2D coordinates of a pixel in the appearance image.

Thus, $\Delta\boldsymbol{s} \simeq \Delta\boldsymbol{s}'$, where $\Delta\boldsymbol{s} = \boldsymbol{s}_n - \boldsymbol{s}_s$ and $\Delta\boldsymbol{s}' = \boldsymbol{s}'_n - \boldsymbol{s}'_s$ are the shape differences between two different expressions of two persons, and the subscripts n and s denote the neutral and s facial expression state, respectively. Similarly, we also know that $R(u,v) \simeq R'(u,v)$.

We can obtain the neutral shape vector $\boldsymbol{s}'_n$ of a new person from the new person's shape vector $\boldsymbol{s}'_s$ with the facial expression s and the shape difference $\Delta\boldsymbol{s}$ of a given person by the following:

$$\boldsymbol{s}'_n = \boldsymbol{s}'_s + \Delta\boldsymbol{s}' \simeq \boldsymbol{s}'_s + \Delta\boldsymbol{s}. \tag{23}$$

Similarly, we can obtain the neutral appearance image $A'_n(u,v)$ of a new person from the new person's appearance image $A'_s(u,v)$ with the facial expression s and the appearance ratio $R(u,v)$ of a given person by the following:

$$A'_n(u,v) = R'(u,v)A'_s(u,v) \simeq R(u,v)A'_s(u,v). \tag{24}$$

Fig. 3 summarizes the overall procedure of indirect facial expression transformation of a new person using relative expression parameters (shape difference and appearance ratio). The detailed explanation of the procedure is given below:

(1) Perform the AAM fitting for the input image I and extract the facial feature vector $\boldsymbol{y}$ by the method described in Section 2.
(2) Perform facial expression recognition to obtain the expression state s.
(3) Obtain a style specific basis matrix $\boldsymbol{W}_s$. Then, compute the content (identity) vector $\boldsymbol{b}_{\text{iden}}$ and obtain the model fitted feature vector $\boldsymbol{y}_s = \boldsymbol{W}_s\boldsymbol{b}_{\text{iden}}$.
(4) Obtain the neutral facial expression feature vector $\boldsymbol{y}_n = \boldsymbol{W}_n\boldsymbol{b}_{\text{iden}}$.
(5) Compute the relative expression parameters such as shape difference and appearance ratio as

$$\Delta\boldsymbol{s} = \boldsymbol{s}_{\boldsymbol{y}_n} - \boldsymbol{s}_{\boldsymbol{y}_s}, \quad R(u,v) = \frac{A_{\boldsymbol{y}_n}(u,v)}{A_{\boldsymbol{y}_s}(u,v)}, \tag{25}$$

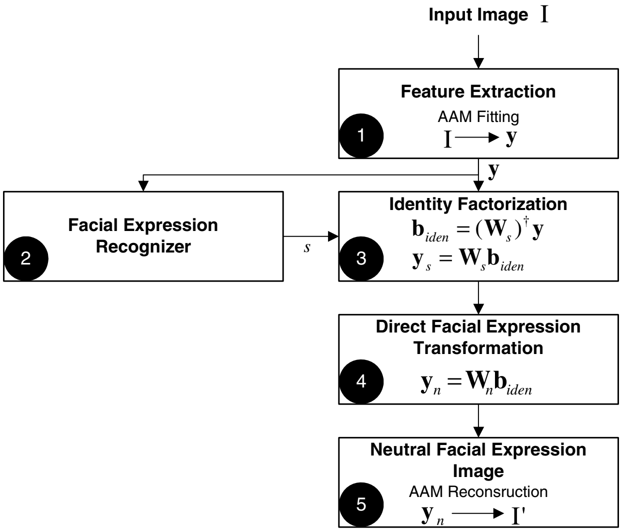
Input Image I
Feature Extraction
AAM Fitting
I → y
1
y
Facial Expression Recognizer
2
s
Identity Factorization
$\mathbf{b}_{iden} = (\mathbf{W}_s)^{\dagger}\mathbf{y}$
$\mathbf{y}_s = \mathbf{W}_s\mathbf{b}_{iden}$
3
Direct Facial Expression Transformation
$\mathbf{y}_n = \mathbf{W}_n\mathbf{b}_{iden}$
4
Neutral Facial Expression Image
AAM Reconsruction
$\mathbf{y}_n \rightarrow \mathbf{I}'$
5

Fig. 2. Overall procedure of the direct facial expression transformation.

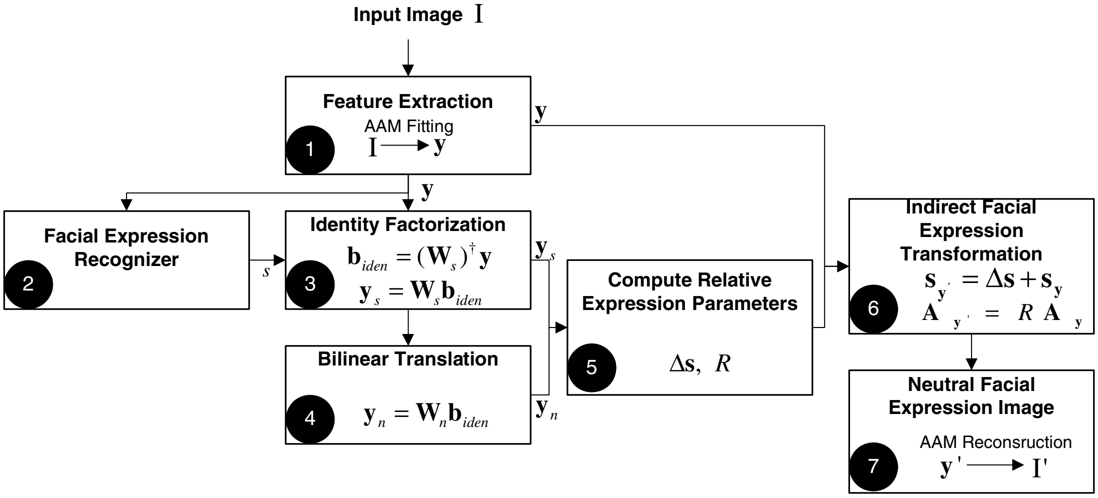
Input Image I
Feature Extraction
AAM Fitting
I → y
1
y
y
Facial Expression Recognizer
2
s
Identity Factorization
$\mathbf{b}_{iden} = (\mathbf{W}_s)^{\dagger}\mathbf{y}$
$\mathbf{y}_s = \mathbf{W}_s\mathbf{b}_{iden}$
3
$\mathbf{y}_s$
Bilinear Translation
$\mathbf{y}_n = \mathbf{W}_n\mathbf{b}_{iden}$
4
$\mathbf{y}_n$
Compute Relative Expression Parameters
$\Delta\mathbf{s},\ R$
5
Indirect Facial Expression Transformation
$\mathbf{s}_{\mathbf{y}'} = \Delta\mathbf{s} + \mathbf{s}_{\mathbf{y}}$
$\mathbf{A}_{\mathbf{y}'} = R\,\mathbf{A}_{\mathbf{y}}$
6
Neutral Facial Expression Image
AAM Reconsruction
$\mathbf{y}' \rightarrow \mathbf{I}'$
7

Fig. 3. Overall procedure of the indirect facial expression transformation.

where $\mathbf{s}_{\mathbf{y}_n}$ and $\mathbf{s}_{\mathbf{y}_s}$ are the shape vectors of $\mathbf{y}_n$ and $\mathbf{y}_s$, $A_{\mathbf{y}_n}$ and $A_{\mathbf{y}_s}$ are the appearance images of $\mathbf{y}_n$ and $\mathbf{y}_s$, and (u,v) are the 2D coordinates of a pixel in the appearance image.

(6) Transform the input feature vector $\mathbf{y}$ into its corresponding neutral facial expression feature vector $\mathbf{y}'$ using the relative expression parameters $\Delta\mathbf{s}$ and R:

$$\mathbf{s}_{\mathbf{y}'} = \Delta\mathbf{s} + \mathbf{s}_{\mathbf{y}}, \quad A_{\mathbf{y}'}(u,v) = R(u,v)A_{\mathbf{y}}(u,v). \tag{26}$$

(7) Obtain the neutral facial expression image I' by reconstructing the AAM parameters of the neutral facial expression feature vector $\mathbf{y}'$.

Fig. 4 shows some examples of the transformed facial images, where each column represents five different subjects and each rows represents the input happy facial expression images (row 1), the model fitted images (row 2), the transformed neutral expression images obtained from the direct expression transformation (row 3), the neutral facial expression images using the indirect facial expression transformation (row 4) and the ground-truth neutral expression images (row 5).

As you can see from Fig. 4, we successfully transformed the input images (row 1) into the neutral expression images (row 3). If the transformation were perfect, the transformed images would be almost identical to the ground-truth images. However, the transformed images are far from the ground-truth face images. This is mainly because we have a small number of subjects in our training set and the tensor model trained with a limited training set cannot fully express a new subject who is far different from the subjects in the training data set. As a result, the performance of face recognition using the direct facial expression transformation is poor.

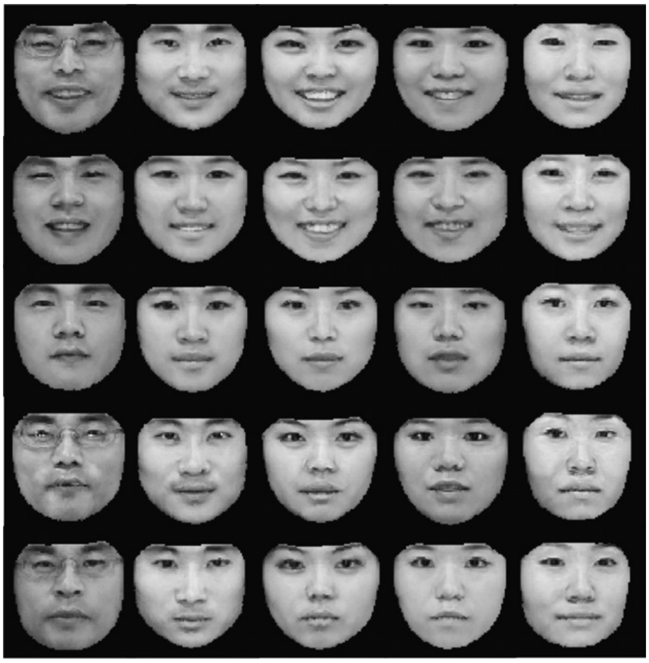

Fig. 4. Some facial expression images obtained from the facial expression transformation.

In contrast, the neutral images in the fourth row are almost identical with the ground-truth images. This is mainly because the directly transformed images were obtained from a combination of the training data of the tensor model, but the indirectly transformed images were obtained by modifying the input images using the relative expression parameters. This provides more effective facial expression transformations for people not included in training data, greatly improving facial recognition performance.

5. Experimental results and discussion

5.1. Experiment setup

We used the database PF07 (Postech Faces 2007) (Lee et al., 2007), which includes 100 male and 100 female subjects and the face images of each subject were captured with four different expressions in five different poses under 16 illuminations. The four expressions are neutral, happy, surprise, and angry. The five poses are front, left, right, upper, and down, and the angle between the frontal pose and other poses is 22.5°. The 16 illuminations consist of no light condition and 15 different light conditions, where each light condition corresponds to the turn-on of the light on a specific location, and 15 locations are the intersection points of three vertical positions (high, middle, and low) and five horizontal positions ($-90°$, $-45°$, $0°$, $45°$, $90°$). Therefore, the database consist of 200 subjects and there are $4 \times 5 \times 16 = 320$ images for a subject, it contains a total of 64,000 images. Fig. 5 shows four facial expressions of a specific person in the database.

Since we consider only expression variations, we selected 1280 (=$80 \times 4 \times 4$) face images of 80 subjects with four facial expressions in the frontal pose under four moderate illuminations to evaluate the proposed expression-invariant face recognition method. These images were used to construct the active appearance model for facial feature extraction. The model was built using 21 shape bases, 187 2D appearance bases and 110 concatenated feature bases. Each number of bases was selected to account for 95% shape, appearance, and feature variation.

We divided 1280 face images into two disjoint sets: training set and test set. Among 80 subjects, we randomly took 40 subjects for training set and remaining 40 subjects for test set, where the training set was used to build the tensor model and the test set was used to evaluate the performance of the proposed method. The test set was divided into two independent sets: a probe set and a gallery set, where the gallery set consisted of the face images with the neutral expression and the probe set consisted of the face images with other expressions. We set $I = 40$, $J = 4$ and $K = 110$ for building the tensor model (Section 3.1) and set $\lambda = 8$ for the ridge regressive bilinear model, which was chosen to provide the best classification performance through many different trials.

Fig. 6 shows the detailed experimental procedure. First, the facial features were extracted from all face images in the database

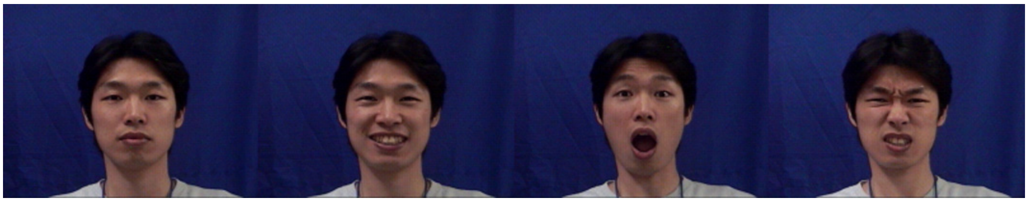

Fig. 5. The four facial expressions of one person in the PF07 database.

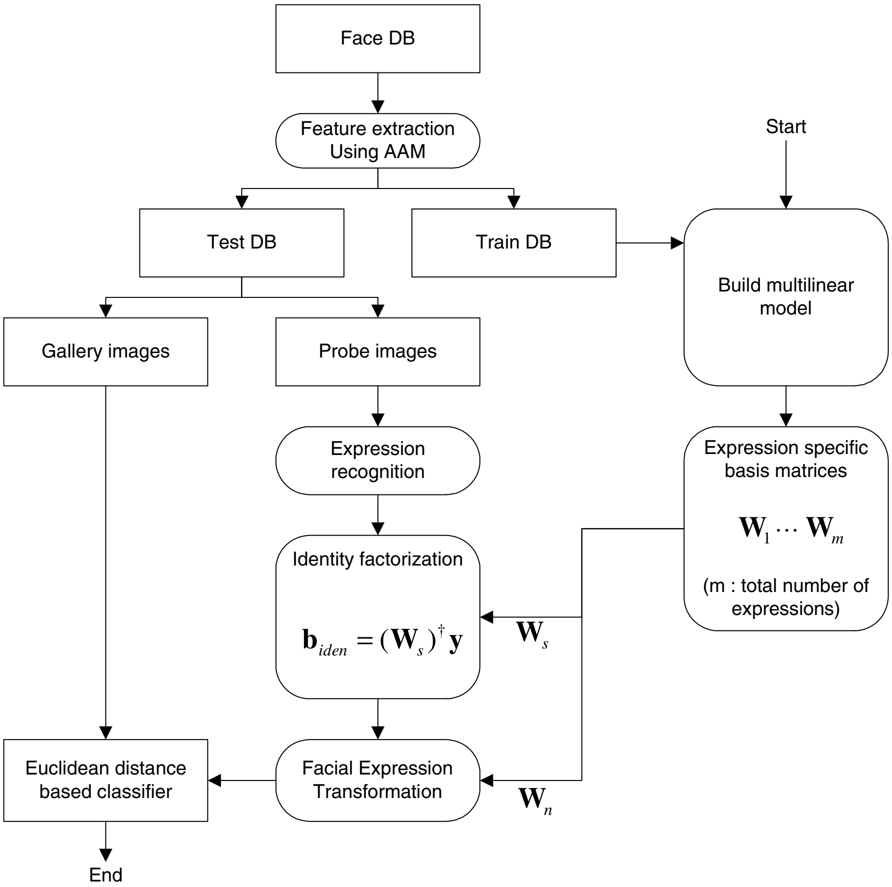
Face DB
Feature extraction Using AAM
Start
Test DB
Train DB
Build multilinear model
Gallery images
Probe images
Expression recognition
Expression specific basis matrices
$\mathbf{W}_1 \cdots \mathbf{W}_m$
(m : total number of expressions)
Identity factorization
$\mathbf{b}_{iden} = (\mathbf{W}_s)^\dagger \mathbf{y}$
$\mathbf{W}_s$
Euclidean distance based classifier
Facial Expression Transformation
$\mathbf{W}_n$
End

Fig. 6. Detailed experimental procedure of the proposed expression-invariant face recognition.

using the AAM. Then, the face database was divided into the training set and the test set. The training set was used to build the tensor model and to obtain the expression specific basis matrices $\boldsymbol{W}_s,\ s = 1,\ldots,4$, where s denotes a specific facial expression. The test set was divided into the gallery set and the probe set. The gallery set consisted of the face images with the neutral expression of all subjects and the probe set consisted of all remaining images. Then, the facial expression recognition was performed using the

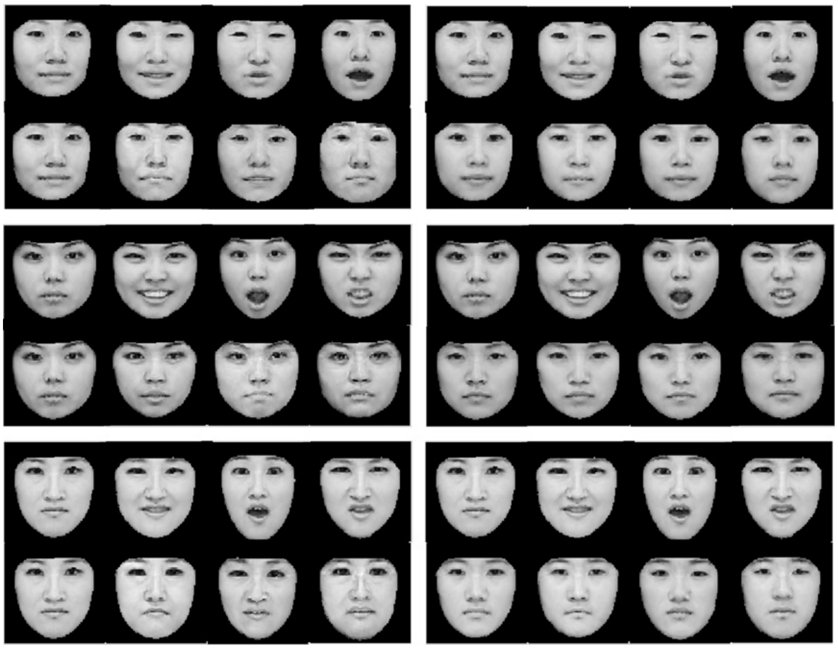

Fig. 7. Illustrative examples of expression transformations.

feature vector of the probe image, and the facial expression factor was determined. Then, the identity factor of the probe image was obtained by multiplying the pseudoinverse of the expression specific basis matrix and the feature vector of the probe image. Finally, the identity of the feature vector of gallery image which had the minimum Euclidean distance to the feature vector of the transformed probe image was selected as the identity of the probe image.

Table 1
Face recognition results

Transformations	Methods	Happy	Surprise	Anger	Average
TYPE 1	NN	45	22.5	67.5	45
	LDA + NN	77.5	80	72.5	76.67
	GDA + NN	85	75	77.5	79.16
TYPE 2	NN	62.5	47.5	37.5	49.17
	LDA + NN	87.5	67.5	6.5	74.17
	GDA + NN	80	70	67.5	72.5
TYPE 3	NN	62.5	50	42.5	51.67
	LDA + NN	87.5	75	77.5	80
	GDA + NN	82.5	70	75	75.83
TYPE 4	NN	77.5	55	60	68.33
	LDA + NN	100	92.5	92.5	95
	GDA + NN	100	97.5	92.5	96.67

5.2. Result of facial expression transformations

We selected some images from the test set and transformed each image to its corresponding neutral facial expression images. Fig. 7 shows the result of the facial expression transformations for three different subjects, where the left 4 column images are the result of indirect facial expression transformation and the right 4 column images are the result of direct facial expression transformation. In addition, for each subject the upper images are the input face images performing four different expressions and the lower images are the result of facial expression transformation. As you can see in Fig. 7a–c, the transformed neutral facial expression images of four different poses look similar to each other and they are distinguishable from other subjects. Moreover, when we compare the indirectly transformed neutral facial expression images and the directly transformed neutral facial expression images, indirectly transformed neutral facial expression images are more similar to the input face images and this will improve the performance of the face recognition.

5.3. Comparison of face recognition performance

We compared the recognition rate of the proposed facial expression transformation method with those of the well-known face recognition methods. For doing this, we performed four

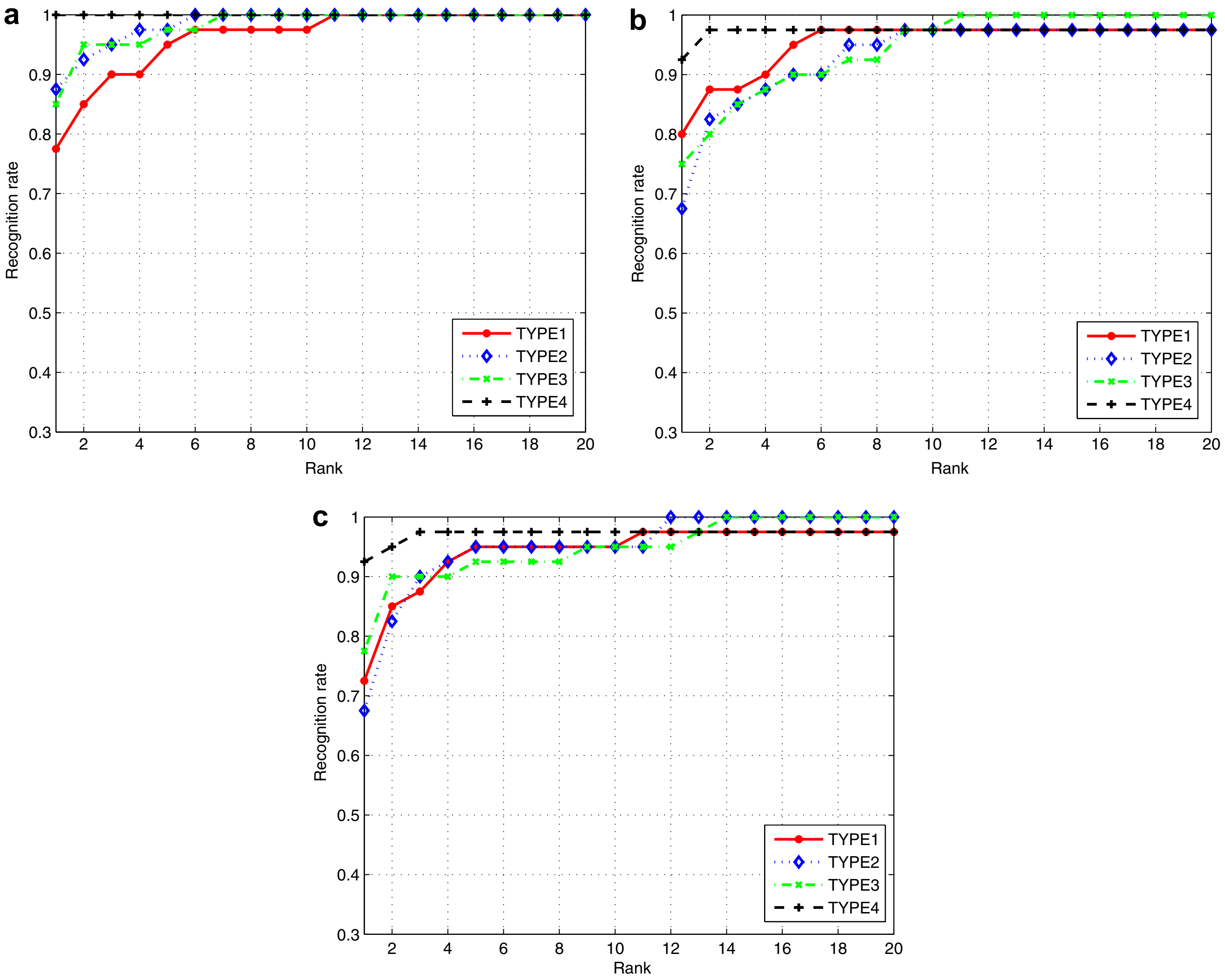

Fig. 8. Face recognition rates for (a) happy, (b) surprise, and (c) anger facial expression images.

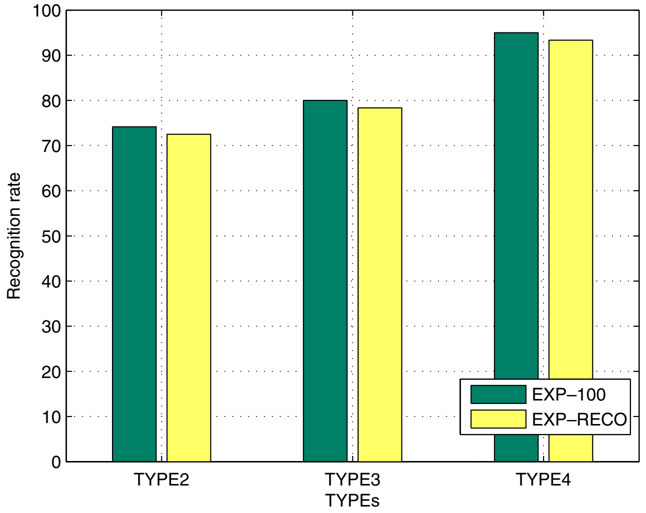

Fig. 9. Comparison of the face recognition rates when the perfect and real face expression recognizers are used.

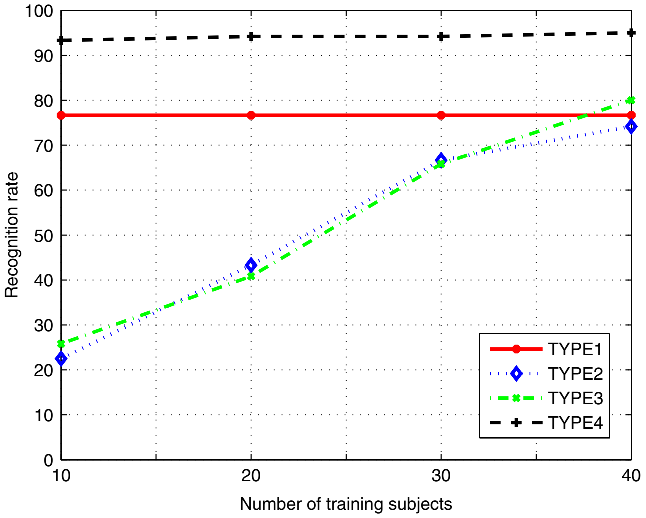

Fig. 10. The face recognition rate over the different numbers of training subjects.

different types of face recognition: the face recognition with no facial expression transformation (TYPE 1), the face recognition with the direct facial expression transformation (TYPE 2), the face recognition with the direct facial expression transformation using the ridge regressive bilinear model (TYPE 3), and the face recognition with the indirect facial expression transformation (TYPE 4). Also, we took three different classification methods: the nearest neighbor method (NN), the linear discriminant analysis followed by NN (LDA + NN), and the generalized discriminant analysis followed by NN (GDA + NN).

Table 1 summarizes the face recognition results. As you can see in this table, we know that (1) the face recognition using the TYPE 4 outperforms other methods by about 20% on average, (2) when the NN classifier is used, the recognition rates of the TYPE 2 and TYPE 3 are better than that of the TYPE 1, (3) when the discriminant analysis is applied, the recognition rates of TYPE 1, TYPE 2, and TYPE 3 are shown to be roughly comparable, (4) when the NN classifier is used, the recognition rates of the TYPE 4 are greatly better than those of TYPE 1 for the happy and surprise facial expression because the happy and surprise images are much more different from the neutral images than the others and our transformation method is more effective for these extreme expressions.

Fig. 8a–c shows the face recognition rates for the happy, surprise, and anger facial expression in terms of the cumulative match characteristic (CMC) curves, respectively, where the rank p in the horizontal axis implies that the matching is successful if there exists more than one correct recognition results within the top p highest degree of matching. The proposed face recognition of the indirect expression transformation method (TYPE 4) always outperforms those of the direct expression transformation method (TYPE 2 and TYPE 3) and that of the conventional methods (TYPE 1).

We also investigated the effect of the facial expression recognizer on the face recognition performance. For the above face recognition experiments, we assumed that the facial expressions of the input face images were known. In real situation, the facial expression recognizer is not perfect and there should be some recognition error of facial expression. To consider the effect of facial expression recognizer, we recognize the facial expression of the input face images using the facial expression recognizer (Sung et al., 2006), which was developed by another group of our laboratory and use the expression specific basis matrix which is obtained from the recognized facial expression results. The average facial expression recognition rate for the test image set was 84.38%. Fig. 9 compares the face recognition rate obtained from the perfect facial expression recognition (EXP-100) with the face recognition rate obtained from the real facial expression recognizer (EXP-RECO). The recognition rate of EXP-RECO is slightly lower than that of EXP-100, but the difference is not big.

5.4. The effect of the number of training subjects

We evaluated the recognition rate when the number of training subject is changed. For doing this, the expression specific basis matrices were built with four different numbers of training subjects: 10, 20, 30, and 40. Next, each probe image was transformed into a neutral image using different expression specific basis matrices, and its identity was classified using the linear discriminant analysis followed by nearest neighbor classifier (LDA + NN). Fig. 10 shows the face recognition rates for different numbers of training subject. As you can see in this figure, we know that (1) the recognition rate increases as the number of training subjects increases for TYPE 2 and TYPE 3 because the direct transformation to the neutral images becomes better with the larger numbers of training subjects, and (2) the face recognition rate of TYPE 4 is almost constant over the different numbers of training subjects because the transformed neutral expression images are quite similar with the gallery images regardless of the number of training subjects.

6. Conclusion

In this paper, we proposed an expression-invariant face recognition method. To achieve expression-invariance, we first extract the facial feature vector from the input image using AAM. Then, we obtain the facial expression state of the input facial feature vector by the facial expression recognizer. Next, we transform the input facial feature vector into its corresponding neutral facial expression vector using direct or indirect facial expression transformation. Finally, we perform the expression-invariant face recognition by distance-based matching techniques nearest neighbor classifier, linear discriminant analysis (LDA), and generalized discriminant analysis (GDA).

The experimental results validate that the face recognition rate of the indirect facial expression transformation is greatly improved over that without transformation, especially for the extreme expressions happy and surprise. In addition, since we obtain the relative expression parameters and transform the facial expression separately, the face recognition rate of the indirect facial expression transformation is almost constant over the different numbers of training subjects, while that of the direct facial expression transformations decreases as the number of training subject decreases. This indicates that the proposed facial expression transformation method is an appropriate and practical face recognition method even when the number of training subjects is very limited.

Since we proposed the transformation method using multi-linear model, the extension of this work could be the development of a face recognition method which is robust to multiple variations like pose, illumination, and expression.

Acknowledgements

This work was partially supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center (BERC) at Yonsei University. Also, it was supported by the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Knowledge Economy (MKE).

References

Abboud, B., Davoine, F., 2004. Face appearance factorization for expression analysis and synthesis. In: Proc. Workshop on Image Analysis for Multimedia Interactive Services.

Baudat, G., Anouar, F., 2000. Generalized discriminant analysis using a kernel approach. Neural Comput. 12 (10), 2385–2404.

Bronstein, A., Bronstein, M., Kimmel, R., 2003. Expression-invariant 3D face recognition. In: Proc. Audio- and Video-Based Biometric Person Authentication, pp. 62–70.

Cootes, T., Edwards, G., Taylor, C., 1998. Active appearance models. In: Burkhardt, H., Neumann, B. (Eds.), Proc. European Conf. on Computer Vision, pp. 484–498.

Elad, A., Kimmel, R., 2001. On bending invariant signatures for surfaces. IEEE Trans. Pattern Anal. Machine Intell. 25 (10), 1285–1295.

Etemad, K., Chellappa, R., 1997. Discriminant analysis for recognition of human face images. J. Opt. Soc. Am. 14 (8), 1724–1733.

Kanade, T., 1973. Picture processing by computer complex and recognition of human face. Ph.D. Thesis, Kyoto University.

Lathauwer, L.D., Moor, B.D., Vandewalle, J., 2000. A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. 21 (4), 1253–1278.

Lee, H.-S., Kim, D., 2006. Facial expression transformations for expression-invariant face recognition. In: Proc. Internat. Symposium on Visual Computing, pp. 323–333.

Lee, H.-S., Park, S., Kang, B., Shin, J., Lee, J.-Y., Je, H.-M., Jun, B., Kim, D., 2007. Asian face image database PF07. Technical Report, Intelligent Media Lab, Department of CSE, POSTECH.

Li, X., Mori, G., Zhang, H., 2006. Expression-invariant face recognition with expression classification. In: Proc. Third Canadian Conf. on Computer and Robot Vision, p. 77.

Liu, Y., Schmidt, K., Cohn, J., Mitra, S., 2003. Facial asymmetry quantification for expression invariant human identification. Computer Vision and Image Understanding 91 (1), 138–159.

Matthews, I., Baker, S., 2004. Active appearance models revisited. Internat. J. Comput. Vision 60 (2), 135–164.

Shin, D., Lee, H.-S., Kim, D., 2008. Illumination robust face recognition using ridge regressive bilinear models. Pattern Recognition Lett. 29 (1), 49–58.

Sung, J., Lee, S.-J., Kim, D., 2006. A real-time facial expression recognition using the STAAM. In: Proc. Internat. Conf. on Pattern Recognition, pp. 275–278.

Turk, M., Pentland, A., 1991. Eigenfaces for recognition. J. Cognit. Neurosci. 3, 72–86.

Vasilescu, M., Terzopoulos, D., 2002. Multilinear analysis of image ensembles: Tensorfaces. In: Proc. European Conf. on Computer Vision, pp. 447–460.

Wang, H., Ahuja, N., 2003. Facial expression decomposition. In: Proc. IEEE Internat. Conf. on Computer Vision, pp. 958–964.

Zhou, C., Lin, X., 2005. Facial expressional image synthesis controlled by emotional parameters. Pattern Recognition Lett. 26, 2611–2627.