A comparative study of facial appearance modeling methods for active appearance models

Sung Joo Lee\textsuperscript{a}, Kang Ryoung Park\textsuperscript{b}, Jaihie Kim\textsuperscript{a,\textasteriskalign{left}{*}}

\textsuperscript{a}School of Electrical and Electronic Engineering, Yonsei University, Biometrics Engineering Research Center, Seoul 120-749, Republic of Korea
\textsuperscript{b}Department of Electronics Engineering, Dongguk University, Biometrics Engineering Research Center, Republic of Korea

\textbf{A R T I C L E   I N F O}

Article history:
Received 11 April 2008
Received in revised form 27 May 2009
Available online 10 June 2009
Communicated by H.H.S. Ip

\textbf{Keywords:}
Active appearance model
Non-negative matrix factorization
Facial appearance modeling
Local non-negative matrix factorization
Non-smooth non-negative matrix factorization

\textbf{A B S T R A C T}

Active appearance models (AAMs) have been widely used in many face modeling and facial feature extraction methods. One of the problems of AAMs is that it is difficult to model a sufficiently wide range of human facial appearances, the pattern of intensities across a face image patch. Previous researches have used principal component analysis (PCA) for facial appearance modeling, but there has been little analysis and comparison between PCA and many other facial appearance modeling methods such as non-negative matrix factorization (NMF), local NMF (LNMF), and non-smooth NMF (ns-NMF). The main contribution of this paper is to find a suitable facial appearance modeling method for AAMs by a comparative study. In the experiments, PCA, NMF, LNMF, and ns-NMF were used to produce the appearance model of the AAMs and the root mean square (RMS) errors of the detected feature points were analyzed using the AR and BERC face databases. Experimental results showed that (1) if the appearance variations of testing face images were relatively non-sparser than those of training face images, the non-sparse methods (PCA, NMF) based AAMs outperformed the sparse methods (nsNMF, LNMF) based AAMs. (2) If the appearance variations of testing face images are relatively sparser than those of training face images, the sparse methods (nsNMF) based AAMs outperformed the non-sparse methods (PCA, NMF) based AAMs.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Human faces vary according to complex factors such as identity, illumination, expression, and pose variations. Because of this complexity, modeling human faces still attracts a lot of attention among computer vision researchers. Since the eigenface was introduced by Turk and Pentland (1991), various statistical learning methods such as independent component analysis (ICA), NMF, Kernel PCA (KPCA), and so on have been used to model human appearance variations and recognize faces (Gregory and Baback, 2004; Lee and Seung, 1999). However, these methods only model the appearance of the face without the shape modeling. Therefore, by using these methods, we cannot represent facial shape variations caused by expression and pose variations exactly.

Active appearance models (AAMs) belong to generative parametric models that model not only the appearance but also the shape of the human face. Because of this property, AAMs have been widely used to model faces and find facial feature points, such as eyes, and lip contours, and so on (Cootes et al., 2001; Matthews and Baker, 2004). Generally, AAMs consist of two parts. The first part is modeling, which consists of parametric shape and appearance models to represent and generate various faces. If these parametric shape and appearance models are used independently, the model is called an independent AAM. If the two models are combined, it is called a combined-AAM (Cootes et al., 2001; Matthews and Baker, 2004). The second part is fitting, where the goal is to find the shape and appearance parameters (or combined parameters) that minimize the appearance error between the synthesized face image and the warped input face image.

For improving the modeling parts, Costen et al. (2002) proposed a method for distinguishing truly ‘facial’ variations from ensemble-specific variations. Gross et al. (2005) found that producing a generic facial shape model was easier than producing a generic facial appearance model. To solve this problem, Gross et al. (2005) proposed a person-specific appearance model.

There has been a lot of previous research into improving convergence speed and accuracy during the fitting part. The original AAM proposed by Cootes et al. (1998) used a simple least square regression method for the AAM fitting part. In later work, Cootes et al. (2001) replaced the least square fitting part by Gauss–Newton optimization approach with a fixed Jacobian matrix to improve fitting accuracy and increase convergence speed. Batur and Hayes (2005) proposed an adaptive AAM, in which the fixed Jacobian matrix was replaced by an adaptive Jacobian matrix to further
improve fitting accuracy, Donner et al. (2006) proposed an efficient fitting algorithm based on canonical correlation analysis (CCA). Using CCA, these researchers efficiently modeled the dependency between the appearance residuals and the model parameters. Sung and Kim combined an active contour with an AAM to produce a background-robust AAM (Sung and Kim, 2007a). In addition, Sung and Kim proposed a fitting method that combined an active shape model and an AAM to improve fitting accuracy (Sung and Kim, 2007b). Matthews and Baker proposed the Project Out Inverse Compositional (POIC) algorithm to improve fitting accuracy and increase convergence speed (Matthews and Baker, 2004). In this approach, they projected out appearance variations during the fitting part and updated the shape and appearance model parameters sequentially. Baker et al. (2003) also proposed the Simultaneous Inverse Compositional (SIC) algorithm, where the shape and appearance model parameters were updated simultaneously. Gross et al. (2005) compared the POIC and SIC and found that the fitting performance of the SIC was far better than that of the POIC when working with unseen data. Therefore, in this paper, we used the SIC as the standard fitting method. A review of other fitting methods can be found in (Cootes and Kittipanya-ngam, 2002 and Li and Jain, 2005).

Most of the above approaches used principal component analysis (PCA) during the appearance modeling part. PCA is the theoretically optimal dimension reduction method for training data in terms of the least square error. However, PCA has two limitations for facial appearance modeling. Firstly, PCA generates a Gaussian space to represent training face data but this Gaussian space cannot represent untrained face data exactly because the distribution of faces is not exactly Gaussian (Gong et al., 2000; Bartlett et al., 2002). Secondly, the PCA bases are basically global if there are no dominant local appearance variations in the training data. Therefore, the PCA bases cannot represent untrained local appearance variations caused by different illumination conditions, expression, and pose changes.

As an alternative, parts-based representation methods have been proposed (Lee and Seung, 1999; Li et al., 2001; Pascual-Montano et al., 2006). The basic assumption of these methods is to generate the entire face from the summation of facial parts. For this purpose, a non-negative constraint is added to the bases and coefficients. As a result, the bases and coefficients of these methods are sparse. Therefore, untrained local appearance variations caused by different illumination conditions, expression and pose changes can be effectively represented by these sparse bases. In addition, except for the non-negative constraint, parts-based representation methods do not have the statistical constraint of a Gaussian face distribution. Therefore, parts-based representation methods can describe general facial distributions better than PCA.

In this paper, we make various bases for the appearance model of the AAM by using PCA, non-negative matrix factorization (NMF), local NMF (LNMF), and non-smooth NMF (ns-NMF), and analyze their effects on AAM performance when using the AR and BERC face databases.

The rest of this paper is organized as follows: in Section 2, the traditional AAM modeling procedure and the SIC fitting algorithm are described. In Section 3, parts-based representation methods are explained. In Section 4, an algorithm that combines AAM and parts-based representation methods is presented. Finally, Sections 5 and 6 present the experimental results and offer conclusions.

2. Active appearance models

2.1. AAM modeling method

Traditional AAMs represent faces with statistical facial shape and appearance models that use PCA (Cootes et al., 2001; Matthews and Baker, 2004). In order to produce the facial shape model, k vertices \((x_1, y_1), \ldots, (x_k, y_k)\) were annotated manually for each training facial image (Fig. 1) and these made up a facial shape \(s = [x_1, y_1, \ldots, x_k, y_k]\). Then, \(s\) was represented by a linear combination of the mean facial shape \(s_0\) and \(n\) facial shape variation vectors \(s_i\):

\[
s = s_0 + \sum_{i=1}^{n} p_i s_i
\]

where, \(p = [p_1, p_2, \ldots, p_n]\) represents the shape parameter vector.

After the shape model was produced, each training facial image was warped to obtain the mean facial shape \(s_0\) and the shape-free facial appearance \(A\), which consisted of the intensities of the warped input image as shown in Fig. 1, was modeled by a linear combination of the mean facial appearance \(A_0\) and \(m\) facial appearance variation vectors \(A_i\):

\[
A = A_0 + \sum_{i=1}^{m} \lambda_i A_i
\]

where, \(\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_m]\) represents the appearance parameter vector. Note that \(s\) and \(A\) were obtained from the training facial images using PCA (Cootes et al., 2001; Matthews and Baker, 2004). The appearance model for the particular image pixel can be written as:

\[
A(x) = A_0(x) + \sum_{i=1}^{m} \lambda_i A_i(x) \quad \forall x \in p(s_0)
\]

where \(p(s_0)\) denotes the set of pixels \(x = (x,y)^T\) that lay inside the mean facial shape \(s_0\) and \(A(x)\) means the intensity value of the facial appearance \(A\) at point \(x\). A detailed explanation of the AAM modeling part can be found in (Cootes et al., 2001).

2.2. AAM fitting method

The goal of AAM fitting methods is to find the shape and appearance parameter vectors that minimize the errors between the synthesized face image and the warped input face image. The objective function is as follows:

\[
\sum_{x \in p(s_0)} \left[ A_0(x) + \sum_{i=1}^{m} \lambda_i A_i(x) - I(W(x, p)) \right]^2
\]

where \(W\) represents the warping function to change the point location from point \(x\) in the input face image coordinate to point \(W(x, p)\) in the synthesized image coordinate (Matthews and Baker, 2004).

The SIC algorithm minimized Eq. (4) by performing a Gauss-Newton gradient descent optimization simultaneously on the shape parameter \(p\) and the appearance parameter \(\lambda\) (Baker et al., 2003; Gross et al., 2005). The algorithm iteratively minimized:

\[
\sum_{x \in p(s_0)} \left[ A_0(W(x, \Delta p)) + \sum_{i=1}^{m} (\lambda_i + \Delta \lambda_i) A_i(W(x, \Delta p)) - I(W(x, p)) \right]^2
\]
with respect to \( \Delta p, \Delta \lambda = [\Delta \lambda_1, \Delta \lambda_2, \ldots, \Delta \lambda_m] \). When we denoted the combined parameter vector \( q \) and update of \( q \) as:

\[
q = \begin{pmatrix} p \\ \lambda \end{pmatrix}, \quad \Delta q = \begin{pmatrix} \Delta p \\ \Delta \lambda \end{pmatrix}
\]

(6)

then the \( \Delta q \) was found as follows:

\[
\Delta q = -H^{-1} \sum_x S^D(x)E_{app}(x)
\]

(7)

where the steepest descent image \( SD, \) the hessian \( H, \) and the appearance error \( E_{app}(x) \) were calculated as follows:

\[
SD(x) = \left( \nabla A \frac{\partial W}{\partial p_1}, \ldots, \nabla A \frac{\partial W}{\partial p_n}, A_1(x), \ldots, A_m(x) \right)
\]

(8)

\[
H^{-1} = \sum_x SD(x)SD(x)'
\]

(9)

\[
E_{app}(x) = I(W(x;p)) - \left[ A_0(x) + \sum_{i=1}^m \lambda_i A_i(x) \right]
\]

(10)

A detailed explanation of this algorithm can be found in (Baker et al., 2003; Gross et al., 2005). Other than the SIC algorithm, a POIC algorithm has also been proposed but performance was far lower than that of the SIC for unseen data (Gross et al., 2005). Therefore, in this paper, we used the SIC algorithm as the standard fitting method.

3. Facial appearance modeling methods

Unlike the PCA-based appearance modeling method, parts-based face representation methods do not require facial mean appearance and variations from this mean appearance. They simply represent facial appearances by using a linear combination of certain facial parts.

Non-negative matrix factorization (NMF), a parts-based representation method, was introduced by Lee and Seung (1999). The basic data representation model of this method is as follows:

\[
V = WH
\]

(11)

where, \( V \in \mathbb{R}^{d \times s} \) is a positive data matrix obtained from \( s \) training face images with \( d \) dimensional intensities of a face, respectively. \( W \in \mathbb{R}^{m \times m} \) is a basis matrix composed of \( m \) basis vectors. \( H \in \mathbb{R}^{m \times m} \) is a coefficient matrix composed of \( m \) dimensional coefficient vectors for \( s \) training face images. In addition, all the components of \( W \) and \( H \) should be nonnegative, which makes it possible to represent data by using an additive combination of "parts". In order to obtain \( W \) and \( H \), an iterative matrix update algorithm was proposed in (Lee and Seung, 1999).

The bases obtained by the NMF were relatively global because non-negative constraints alone are not enough to make them local. For this, local nonnegative matrix factorization (LNMF) was introduced by Li et al. (2001). Besides the non-negative constraints, these researchers proposed three additional constraints, called maximum sparseness in \( H \), maximum expressiveness of \( W \), and maximum orthogonality of \( W \). As a result, it was found that the bases obtained by the LNMF were more local than those obtained by the NMF. The iterative matrix update algorithm of the LNMF for \( H \) was proposed in (Li et al., 2001). The updating method of the LNMF for \( W \) was the same as that of the NMF.

Another parts-based representation method, non-smooth non-negative matrix factorization (nsNMF), has also been proposed (Pascual-Montano et al., 2006). Unlike the NMF and LNMF, the nsNMF method can control the sparseness of the bases. The basic idea of the nsNMF is that sparseness in the bases will almost certainly force “non-spariness” or smoothness in the coefficients (and vice versa) because of the multiplicative nature of the model, as shown in Eq. (11). Based on this insight, a smoothing matrix \( S \) was added to the basic data representation model of the NMF and the following data representation model was obtained:

\[
V = WSH
\]

(12)

where the smoothing matrix \( S \in \mathbb{R}^{m \times m} \) appeared as follows:

\[
S = (1 - \theta)I + \frac{\theta}{q}11'
\]

(13)

where \( I \) represents the identity matrix, \( 1 \) represents a vector of ones, and the parameter \( \theta \) ranged from 0 to 1. To explain how \( S \) works, let \( X \) be a positive vector and \( Y = SX \) be a transformed vector. When \( \theta = 0, S \) becomes an identity matrix and no smoothing occurs. However, when \( \theta = 1, \) all the elements of \( Y \) have the average of the elements of \( X. \) In other words, \( Y \) becomes completely “non-sparse”. By using this property, we can control the sparseness of \( W \) and \( H, \) Eq. (12) can be rewritten as:

\[
V = WSH = WS\left( \frac{WH}{H} \right) = \left( \frac{W}{H} \right)SH
\]

(14)

As can be seen in Eq. (13), as \( \theta \) value increases, \( SH \) and \( WS \) are smoothed. As a result, \( W \) and \( H \) become sparse. Updating \( W \) and \( H \) in the nsNMF was shown in (Pascual-Montano et al., 2006).

4. AAMs based on parts-based representation methods

AAMs based on parts-based face representation (parts-based AAMs) and PCA-based AAMs are different. The appearance model of parts-based AAMs is as follows:

\[
A_p = \sum_{i=1}^m \lambda_i W_i
\]

(15)

where, \( W_i \) represents the \( i \)th column vector of the basis matrix and \( \lambda_i \) represents the corresponding coefficient for the facial appearance \( A_p \) obtained by parts-based representation. Unlike PCA-based AAMs, parts-based AAMs do not use mean facial appearances and facial variations from mean facial appearances. They simply represent facial appearances by using a linear combination of the bases, which represent parts of the facial appearance.

In addition, because the appearance model was modified, Eqs. (8)–(10) were also changed as follows:

\[
SD_p(x) = \left( \nabla A_{p_1} \frac{\partial W}{\partial p_1}, \ldots, \nabla A_{p_n} \frac{\partial W}{\partial p_n}, W_{1p}(x), \ldots, W_{mp}(x) \right)
\]

(16)

\[
H_p^{-1} = \sum_x SD_p(x)SD_p(x)'
\]

(17)

\[
E_{app}(x) = I(W(x;p)) - \left[ A_0(x) + \sum_{i=1}^m \lambda_i W_i(x) \right]
\]

(18)

Finally, the update of the shape and appearance parameters \( \Delta q \) were obtained by using the following equation:

\[
\Delta q = -H_p^{-1} \sum_x SD_p(x)E_{app}(x)
\]

(19)

Before updating the shape and appearance parameters iteratively by using Eq. (19), we had to set the initial appearance parameter. In PCA-based AAMs, the mean appearance is given as the initial appearance by making the element of \( A \) in Eq. (2) zero. Although the mean appearance is not exact, it can be used as a reasonable initial appearance for parts-based AAMs. Particularly, the initial appearance parameter set \( A_{ini} \) was calculated by projecting the mean appearance to the basis matrix:

\[
A_{ini} = (W^T W)^{-1} W^T A_0
\]

(20)

5. Experimental results

5.1. Database

In order to evaluate the performance of the AAMs, two databases were used. The first database is the AR face database (Mar-
The resolution of the images was 768 × 576 pixels. There were 135 subjects (76 men and 59 women) in the first session and 120 subjects (65 men and 55 women) in the second session. For each subject in each session, 13 images were captured under the following conditions: 1 neutral, 3 different expressions, 3 different illuminations, and 6 occlusion conditions. It was nearly impossible to obtain manually-marked ground truth points in the occluded face images because the facial feature points were hidden. Therefore, we excluded the 6 occlusion conditions and only used the images that were obtained under the other conditions. Fig. 2 shows some sample images of the AR face database that were used in the experiments. The second database is the BERC face database. The resolution of the images was 640 × 480 pixels. There were 94 subjects (72 men and 22 women). For each subject, 9 images were captured under the following conditions: 1 neutral, 2 different poses, 3 different expressions, and 3 different illuminations. Fig. 3 shows some sample images from the BERC face database.

5.2. Model building and experimental scenarios

In this paper, six AAMs were produced using the following methods: PCA, NMF, LNMF, and nsNMF when \( h = 0.3 \) (nsNMF03), 0.5 (nsNMF05), and 0.8 (nsNMF08). From 80 manually-marked points (as shown in Fig. 1), the same PCA-based shape model was made for the AAMs. However, the different appearance models were produced for the AAMs by using the six methods because we focus on appearance modeling methods and their effects on AAM.
in this research. The number of shape bases was determined to retain 95% shape variations of the training face images. The number of appearance bases was empirically determined to maximize the performance of the AAM.

In addition, we designed two experimental scenarios to evaluate the robustness of the AAMs with respect to facial variations caused by poses, illuminations and expression changes.

The first scenario was an ideal case in which the training and testing pose, illumination, and expression conditions are similar. Images of half of the subjects in each database captured under all conditions were used to generate the shape and appearance models of the AAMs. The performances of the AAMs were measured from the other images of half of the subjects captured under all conditions. The second scenario was a practical case in which the training and testing poses, illuminations and expressions were different. Images of half of the subjects in each database captured under only the neutral condition of the two databases were used to generate the shape and appearance models of the AAMs. The performances of the AAMs were measured from the other images of half of the subjects captured under all conditions. Table 1 summarizes the images used in each experimental scenario.

5.3. Evaluation metrics

To measure the performance of the AAMs, the shape root mean square (RMS) errors and the percentage of divergence (POD) were measured. First, the initial shape errors were provided (as shown in Fig. 4) by changing the image scale ±10%, rotating ±π/10 radians, and displacing the ground truth shape ±10 pixels in the horizontal and vertical directions. After the AAMs fitting, we determined first whether the AAMs diverged or not. We determined that the AAM diverged on a testing image if the shape RMS errors were larger than 10 pixels, because about 10 pixels were provided as an initial error. From the converged testing databases, we measured the shape RMS errors between the manually-marked points and the automatically-marked points by using the AAMs and averaged them to evaluate how the AAMs find the facial feature points accurately. In addition, the POD, which is the rate of diverged testing face images to the total number of testing face images, was measured to evaluate how the AAMs find the facial feature points reliably.

Furthermore, in order to analyze the characteristics of the six appearance modeling methods and their effects on the performance of the AAMs, the reconstruction errors, sparseness of bases, and kurtosis of the appearance parameters were measured.

The reconstruction errors refer to how close the optimal synthesized appearance is to the appearance of the testing facial image. The AAM fitting part finds the optimal appearance parameter that makes up the optimal appearance, the closest appearance to the appearance of the testing facial image. If the appearance model can represent the facial appearance variations well, the synthesized optimal appearance can be similar to the appearance of the testing image.

Table 1

<table>
<thead>
<tr>
<th>Database</th>
<th>Scenario 1</th>
<th>Scenario 2</th>
<th>Scenario 1</th>
<th>Scenario 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training face images</td>
<td>Testing face images</td>
<td>Training face images</td>
<td>Testing face images</td>
<td></td>
</tr>
<tr>
<td>AR</td>
<td>(1 neutral + 3 illumination + 3 expression) × 128 subjects = 894 images</td>
<td>(1 neutral + 3 illumination + 3 expression) × the other 127 subjects = 889 images</td>
<td>1 neutral × 128 subjects = 2 (broken images)</td>
<td>1 neutral × 128 subjects = 2 (broken images)</td>
</tr>
<tr>
<td>BERC</td>
<td>(1 neutral + 2 pose + 3 illumination + 3 expression) × 47 subjects = 423 images</td>
<td>(1 neutral + 2 pose + 3 illumination + 3 expression) × the other 47 subjects = 423 images</td>
<td>(1 neutral + 2 pose + 3 illumination + 3 expression) × 47 subjects = 47 images</td>
<td>(1 neutral + 2 pose + 3 illumination + 3 expression) × the other 47 subjects = 423 images</td>
</tr>
</tbody>
</table>

Table 2

The sparseness of the appearance variations in the AR face database.

<table>
<thead>
<tr>
<th>Target database</th>
<th>AR (Training)</th>
<th>AR (Testing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target scenario</td>
<td>Scenario 1</td>
<td>Scenario 2</td>
</tr>
<tr>
<td>Sparseness</td>
<td>0.1445</td>
<td>0.2166</td>
</tr>
</tbody>
</table>

Table 3

The sparseness of the appearance variations in the BERC face database.

<table>
<thead>
<tr>
<th>Target database</th>
<th>BERC (Training)</th>
<th>BERC (Testing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target scenario</td>
<td>Scenario 1</td>
<td>Scenario 2</td>
</tr>
<tr>
<td>Sparseness</td>
<td>0.1174</td>
<td>0.1235</td>
</tr>
</tbody>
</table>
testing facial image, but if not, the synthesized optimal appearance cannot be similar to the appearance of the testing facial image and the shape RMS errors inevitably increase. Therefore, we measured the reconstruction errors as the mean square error (MSE) between a testing facial image $A$ and a reconstructed face image $\hat{A}$ as follows:

$$\text{MSE} = \frac{1}{k} \sum_{x \in p(s_0)} |A(x) - \hat{A}(x)|^2, \quad k = \sum_{x \in p(s_0)}$$

where $p(s_0)$ denotes the set of pixels $x = (x, y)^T$ that lie inside the mean facial shape $s_0$ and the reconstructed face image $\hat{A}$ was produced by using PCA and parts-based representation methods.

![Fig. 5.](image) The shape RMS errors of the six AAMs in each scenario. The PCA-based AAM showed the smallest RMS point errors among the six methods.

![Fig. 6.](image) The POD of the six AAMs in each scenario. The PCA-based AAM showed the smallest POD among the six methods.

<table>
<thead>
<tr>
<th></th>
<th>PCA</th>
<th>NMF</th>
<th>nsNMF03</th>
<th>nsNMF05</th>
<th>nsNMF08</th>
<th>LNMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression 1</td>
<td>4.5992</td>
<td>4.6426</td>
<td>4.7512</td>
<td>4.8118</td>
<td>4.7207</td>
<td>5.2223</td>
</tr>
<tr>
<td>Expression 2</td>
<td>4.8144</td>
<td>4.8080</td>
<td>4.7512</td>
<td>4.8118</td>
<td>4.8827</td>
<td>5.0965</td>
</tr>
<tr>
<td>Expression 3</td>
<td>6.0286</td>
<td>5.9455</td>
<td>4.7858</td>
<td>4.9034</td>
<td>4.8111</td>
<td>5.3443</td>
</tr>
<tr>
<td>Illumination 1</td>
<td>4.6193</td>
<td>4.6261</td>
<td>4.5931</td>
<td>4.6917</td>
<td>4.6958</td>
<td>4.9843</td>
</tr>
<tr>
<td>Illumination 2</td>
<td>4.8297</td>
<td>4.8672</td>
<td>4.8549</td>
<td>4.8111</td>
<td>4.9528</td>
<td>5.4077</td>
</tr>
<tr>
<td>Illumination 3</td>
<td>4.5084</td>
<td>4.6195</td>
<td>4.5425</td>
<td>4.5662</td>
<td>4.6206</td>
<td>5.4077</td>
</tr>
<tr>
<td>Total data</td>
<td>4.8200</td>
<td>4.825</td>
<td>4.832</td>
<td>4.892</td>
<td>4.887</td>
<td>5.277</td>
</tr>
</tbody>
</table>

Table 4
The shape RMS errors at the optimal dimension of scenario 1 for each condition. Numbers in bold mean the smallest RMS errors for each database among the six appearance modeling methods.
The sparseness of the bases was also measured to analyze the sparse characteristics of the six appearance modeling methods and their effects on the performance of the AAMs. One way to measure the sparseness has been proposed as follows (Hoyer, 2004):

\[
\text{Sparseness}(x) = \frac{\sqrt{d} - \left(\sum x_i \right)/\sqrt{\sum x_i^2}}{\sqrt{d} - 1}
\]  

(22)

where \(d\) represents the dimensionality of the vector \(x\) whose \(j\)th component is \(x_j\). This sparseness value measures how much energy of a vector is concentrated on a few components. This function has a value of 1 if and only if \(x\) contains only a single nonzero component, and takes a value of 0 if and only if all components are equal. The PCA bases are arranged according to their eigenvalues. A large percentage of energy is concentrated on a few bases with large eigen-

Table 5
The shape RMS errors at the optimal dimension of scenario 2 for each condition. Numbers in bold mean the smallest RMS errors for each database among the six appearance modeling methods.

<table>
<thead>
<tr>
<th></th>
<th>PCA</th>
<th>NMF</th>
<th>nsNMF03</th>
<th>nsNMF05</th>
<th>nsNMF08</th>
<th>LNMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression 1</td>
<td>8.1663</td>
<td>8.1147</td>
<td>7.9475</td>
<td>7.936</td>
<td>8.2182</td>
<td>8.1751</td>
</tr>
<tr>
<td>Expression 2</td>
<td>5.1014</td>
<td>5.0716</td>
<td>5.2012</td>
<td>5.2516</td>
<td>5.2203</td>
<td>5.2528</td>
</tr>
<tr>
<td>Expression 3</td>
<td>8.9415</td>
<td>9.1602</td>
<td>8.8868</td>
<td>8.9183</td>
<td>8.9486</td>
<td>8.7674</td>
</tr>
<tr>
<td>Illumination 2</td>
<td>7.6957</td>
<td>7.5437</td>
<td>8.5299</td>
<td>8.3983</td>
<td>7.9373</td>
<td>8.194</td>
</tr>
</tbody>
</table>

Fig. 7. The reconstruction errors for the training face images of the AR database in each scenario. PCA showed the smallest reconstruction errors in the training database.

Fig. 8. The reconstruction errors for the testing face images of the AR database in each scenario. PCA, NMF, nsNMF03, and nsNMF05 showed similar reconstruction errors in the testing database.
Fig. 9. The sparseness of the bases in each scenario. LNMF showed the biggest sparseness among the six methods.

Fig. 10. The kurtosis of the appearance parameters in each scenario. The kurtosis values of PCA and LNMF were smaller than those of other methods.

Fig. 11. The shape RMS errors of the six AAMs in each scenario. The nsNMF08-based AAM showed the smallest RMS point errors among the six methods.
values. Therefore, the sparseness of the PCA bases was measured by using the weighted average, as follows:

$$S_{\text{PCA}} = \frac{1}{E} \sum_{i=1}^{m} E_{i} \cdot \text{Sparseness}(A_i) = \sum_{i=1}^{E} E_{i} \cdot \text{Sparseness}(A_i)$$

(23)

where $A_i, E_i, m$ represent the PCA bases, their corresponding eigenvalues, and the number of bases, respectively. Unlike PCA, the bases of parts-based representation methods are not ordered by energy. Therefore, the sparseness of the bases of parts-based representation methods can be measured by finding the simple average, as follows:

$$S_{\text{Parts}} = \frac{1}{m} \sum_{i=1}^{m} \text{Sparseness}(W_i)$$

(24)

where $W_i, m$ represent the bases of a parts-based representation method, and the number of bases, respectively.

Finally, the kurtosis of the appearance parameters was measured. Kurtosis of appearance parameters has two meanings. First, it describes the distribution of the appearance parameters. For example, if the kurtosis is about 3, the distribution of the appearance parameters is Gaussian, but if the kurtosis is larger than 3, the distribution is super-Gaussian (Hyvarinen et al., 2001). Second, kurtosis describes how many outliers, that were located far from the mean appearance parameter, exist among the entire appearance parameters of the testing database. If the number of outliers increases, the error probability of local minima at the AAM fitting will increase. The kurtosis of the appearance parameters $\lambda$ was calculated as follows:

$$\text{Kurtosis}(\lambda) = \frac{1}{m} \sum_{i=1}^{m} \text{Kurtosis}(W_i)$$

Fig. 12. The POD of the six AAMs in each scenario. The sparse methods (LNMF, and nsNMF08)-based AAMs showed relatively small POD on BERC database.

Table 6
The shape RMS errors at the optimal dimension of scenario 1 for each condition. Numbers in bold mean the smallest RMS errors for each database among the six appearance modeling methods.

<table>
<thead>
<tr>
<th></th>
<th>PCA</th>
<th>NMF</th>
<th>nsNMF03</th>
<th>nsNMF05</th>
<th>nsNMF08</th>
<th>LNMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neutral</td>
<td>2.6586</td>
<td>2.6064</td>
<td>2.6176</td>
<td>2.6583</td>
<td>2.5562</td>
<td>2.5924</td>
</tr>
<tr>
<td>Pose 1</td>
<td>2.7489</td>
<td>2.7576</td>
<td>2.7206</td>
<td>2.8258</td>
<td>2.7802</td>
<td>2.8695</td>
</tr>
<tr>
<td>Pose 2</td>
<td><strong>2.8244</strong></td>
<td>2.8367</td>
<td>2.9089</td>
<td>2.8574</td>
<td>2.8344</td>
<td>2.8336</td>
</tr>
<tr>
<td>Expression 1</td>
<td>3.1843</td>
<td><strong>3.0911</strong></td>
<td>3.1364</td>
<td>3.1345</td>
<td>3.1675</td>
<td>3.1136</td>
</tr>
<tr>
<td>Expression 2</td>
<td>3.2631</td>
<td>3.1821</td>
<td><strong>3.1009</strong></td>
<td>3.2213</td>
<td>3.1637</td>
<td>3.1486</td>
</tr>
<tr>
<td>Expression 3</td>
<td>3.1642</td>
<td>3.1484</td>
<td>3.1114</td>
<td>3.1702</td>
<td>3.1464</td>
<td><strong>3.0075</strong></td>
</tr>
<tr>
<td>Illumination 1</td>
<td>2.7328</td>
<td>2.7232</td>
<td><strong>2.7021</strong></td>
<td>2.8552</td>
<td>2.703</td>
<td>2.7487</td>
</tr>
<tr>
<td>Illumination 2</td>
<td>2.8744</td>
<td>2.9473</td>
<td>2.9305</td>
<td>2.9157</td>
<td><strong>2.8637</strong></td>
<td>2.9457</td>
</tr>
<tr>
<td>Illumination 3</td>
<td>2.8672</td>
<td>2.8651</td>
<td><strong>2.8424</strong></td>
<td>2.9598</td>
<td>2.8461</td>
<td>3.0482</td>
</tr>
<tr>
<td>Total data</td>
<td>2.924</td>
<td>2.906</td>
<td>2.896</td>
<td>2.955</td>
<td><strong>2.895</strong></td>
<td>2.923</td>
</tr>
</tbody>
</table>

Table 7
The shape RMS errors at the optimal dimension of scenario 2 for each condition. Numbers in bold mean the smallest RMS errors for each database among the six appearance modeling methods.

<table>
<thead>
<tr>
<th></th>
<th>PCA</th>
<th>NMF</th>
<th>nsNMF03</th>
<th>nsNMF05</th>
<th>nsNMF08</th>
<th>LNMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neutral</td>
<td>2.6757</td>
<td>2.6107</td>
<td>2.6622</td>
<td>2.652</td>
<td>2.5878</td>
<td>2.7171</td>
</tr>
<tr>
<td>Pose 1</td>
<td>4.9814</td>
<td>4.9335</td>
<td>4.942</td>
<td>4.8347</td>
<td><strong>4.7932</strong></td>
<td>4.8351</td>
</tr>
<tr>
<td>Pose 2</td>
<td><strong>5.1174</strong></td>
<td>5.1332</td>
<td>5.1991</td>
<td>5.1781</td>
<td>5.1783</td>
<td>5.1548</td>
</tr>
<tr>
<td>Expression 1</td>
<td>3.9018</td>
<td>4.7394</td>
<td>5.0167</td>
<td>5.1385</td>
<td>4.7765</td>
<td><strong>4.6585</strong></td>
</tr>
<tr>
<td>Expression 3</td>
<td>4.6523</td>
<td>4.5562</td>
<td>4.685</td>
<td>4.7851</td>
<td><strong>4.5016</strong></td>
<td>4.6006</td>
</tr>
<tr>
<td>Illumination 1</td>
<td>4.9871</td>
<td>5.674</td>
<td>5.0907</td>
<td><strong>4.8361</strong></td>
<td>4.8955</td>
<td>5.6282</td>
</tr>
<tr>
<td>Illumination 2</td>
<td>5.2496</td>
<td>5.5876</td>
<td>5.298</td>
<td>5.1358</td>
<td><strong>4.9041</strong></td>
<td>5.4468</td>
</tr>
<tr>
<td>Illumination 3</td>
<td>3.7375</td>
<td>3.6274</td>
<td>3.8286</td>
<td>3.8314</td>
<td><strong>3.4102</strong></td>
<td>4.1225</td>
</tr>
</tbody>
</table>
Kurtosis = \frac{m \sum_{i=1}^{m} (\hat{\lambda}_i - \bar{\lambda}_m)^4}{\sum_{i=1}^{m} (\hat{\lambda}_i - \bar{\lambda}_m)^2}

where, \hat{\lambda}_i and \bar{\lambda}_m(i) represents the ith component of the optimal appearance parameter and the mean appearance parameter, respectively.

5.4. Sparseness of the appearance variation of the databases

In order to measure how sparse the appearance variation from the mean appearance is, the sparseness of the appearance variation of the databases was measured as follows:

SparsenessDB = \frac{1}{s} \sum_{i=1}^{s} \text{Sparseness}(V_i - V_m) \tag{26}

where, V_i, V_m represent the ith face image in the target database and the mean of the face images in the training database of the target scenario. The sparseness of the appearance variation of the AR and BERC databases in the two scenarios are shown in Tables 2 and 3 respectively. From these tables, we found that (1) scenario 2 contained more sparse appearance variations than scenario 1, and (2) the training images in the AR database contained more sparse appearance variations than those of the BERC database.

5.5. Experimental results using the AR face database

From the testing database of the AR database (as explained in Section 5.2), the shape RMS errors and the POD of the six AAMs in scenarios 1 and 2 were measured (as shown in Figs. 5 and 6, respectively). Tables 4 and 5 show the shape RMS errors under each condition and in each scenario when the six AAMs had optimal dimensions. In addition, to analyze the experimental results, the reconstruction error, the sparseness of the bases, and the kurtosis of the appearance parameters were measured (as shown in Figs. 7–10).

From Figs. 5 and 6, it is clear that (1) in scenario 1, the PCA-based AAM showed the smallest RMS point errors on the total testing images among the six methods. This is because (A) for testing face images, the reconstruction error of PCA is relatively small.
which means that PCA can synthesize appearances close to the testing face images and (B) the kurtosis of the PCA appearance parameters is relatively small, which means that the PCA appearance parameters had relatively small outliers, which were located far from the mean appearance parameter, so the error probability of local minima at the AAM fitting was small. (2) However, in scenario 2, the NMF-based AAM had the smallest RMS point errors on the total testing images among the six methods. This is because (A) for the testing face images, the reconstruction error of NMF is relatively small. PCA showed the smallest reconstruction errors in the training database but in the testing database, PCA, NMF, nsNMF03, and nsNMF05 showed similar reconstruction errors because the Gaussian space obtained in the training database by PCA could not always represent the testing face data that had non-Gaussian distributions exactly. (B) In addition, the testing database of scenario 2 contained more sparse appearance variations than that of scenario 1 (as shown in Section 5.4) so that NMF, which was sparser than PCA (as shown in Fig. 9), represented such sparse appearance variations better than PCA. (C) Compared to scenario 1, the kurtosis of the PCA appearance parameters increased but that of the NMF decreased. (3) In both scenarios, the PCA-based AAM showed the smallest POD on the total testing images because (A) the reconstruction error and the kurtosis of PCA (as shown in Figs. 8 and 10) were relatively small. (4) The POD of scenario 2 was larger than that of scenario 1 because (A) scenario 2 was relatively not as good at representing the shape variations of the mouths in expressions 1 and 3, and (B) the appearance variations caused by expressions and illuminations as shown in Tables 4 and 5. Therefore, as the number of training face images became larger, the reliability of the AAM increased.

5.6. Experimental results using the BERC face database

From the testing face images of the BERC database (as explained in Section 5.2), the shape RMS errors and the POD of the six AAMs in scenarios 1 and 2 were measured (as shown in Figs. 11 and 12, respectively). Tables 6 and 7 show the shape RMS errors at each condition and scenario when the six AAMs had optimal dimensions. In addition, to analyze the experimental results, the reconstruction error, the sparseness of the bases, and the kurtosis of the appearance parameters were measured (as shown in Figs. 13–16).
From Figs. 11 and 12, it is clear that (1) in scenarios 1 and 2, the nsNMF08-based AAM showed the smallest RMS point errors on the total testing images among the six methods. The reconstruction error of nsNMF08 was not better than the relatively non-sparse methods (PCA, NMF, nsNMF03, and nsNMF05) but the nsNMF08 produced sparser bases. The training face images of the BERC database did not contain enough sparse appearance variations needed to represent the sparse appearance variations of the testing database (as shown in Section 5.4). Therefore, the nsNMF08, which produced sparse bases from such training face images, showed good performance. (2) For the same reason, unlike the AR face database, sparse methods (LNMF, and nsNMF08)-based AAM showed relatively small POD on the total testing images in scenarios 1 and 2 of the BERC database.

6. Conclusions

In this paper, the PCA, NMF, LNMF, and nsNMF-based appearance models were produced for the AAM and their effect on the performance of the AAM on the AR and BERC face databases was discussed. In the experimental results, we found that (1) if the appearance variations of testing face images are relatively non-sparser than those of training face images, the non-sparse methods (PCA, NMF) based AAMs outperformed the sparse methods (nsNMF, LNMF) based AAMs because (A) the non-sparse methods showed better reconstruction ability than the sparse methods and (B) from the training face images, the non-sparse methods also can train the sparse appearance variations from the mean appearance. (2) If the appearance variations of the testing face images are relatively sparser than those of the training face images, the sparse methods (nsNMF) based AAMs outperformed the non-sparse methods (PCA, NMF) based AAMs even though the sparse methods showed poorer reconstruction ability than the non-sparse methods because in such training face images, the sparse methods have better ability to represent the sparse appearance variations than the non-sparse methods.

The proposed AAMs based on parts-based representation methods can change the appearance of facial parts by varying the corresponding appearance parameters. In the future, by using such characteristics, we will research a virtual plastic surgery system using these methods.

Acknowledgements

The authors would like to thank Dr. J. Sung, and Prof. D. Kim for their help on implementing the SIC algorithm. This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center (BERC) at Yonsei University. (R112002105070030(2009)).

References


