Bayesian Shape Model for Facial Feature Extraction and Recognition

Zhong Xue [b] Stan Z. Li [a] Eam Khwang Teoh [b,1]

[a] *Microsoft Research, 5/F, Beijing Sigma Center, 49 Zhi Chun Road, 100080, Beijing, China*

[b] *School of EEE, Nanyang Technological University, Singapore, 639798.*

Abstract

A facial feature extraction algorithm using the Bayesian Shape Model (BSM) is proposed in this paper. A full-face model consisting of the contour points and the control points is designed to describe the face patch, using which the warping/normalization of the extracted face patch can be performed efficiently. First, the BSM is utilized to match and extract the contour points of a face. In BSM, the prototype of the face contour can be adjusted adaptively according to its prior distribution. Moreover, an affine invariant internal energy term is introduced to describe the local shape deformations between the prototype contour in the shape domain and the deformable contour in the image domain. Thus both global and local shape deformations can be tolerated. Then, the control points are estimated from the matching result of the contour points based on the statistics of the full-face model. Finally, the face path is extracted and normalized using the piece-wise affine triangle warping algorithm. Experimental results based on real facial feature extraction demonstrate that the proposed BSM facial feature extraction algorithm is more accurate and effective as compared to that of the Active Shape Model (ASM).

Key words: Active Shape Model, Beyasian Shape Model, Facial Feature Extraction, Face Recognition, Principle Component Analysis.

[1] Corresponding author. Tel: +65-6790-5393; Fax: +65-6792-0415.

E-mail address: eekteoh@ntu.edu.sg (EK Teoh)

1 Introduction

Facial feature extraction is widely used in the areas of face recognition, database indexing, image retrieval, multimedia computation and other model-based coding of image sequences containing human faces. For example, in face recognition, a main procedure is first matching and extracting the face patch, and then warping it into the standard view and the normal expression. In this paper, we focus on the problem of facial feature extraction and normalization of the extracted face patches. The newly developed affine invariant deformable model, the Bayesian Shape Model (1; 2) is utilized to match and extract the face contours.

Compared with the traditional rigid models, deformable models have attracted much attention in the areas of object detection and matching because of their ability of adapting themselves to fit objects more closely. Generally, deformable models can be classified into two classes (3): the free-form models and the parametric models. The free-form models, *e.g.* active contours or snakes, can be used to match any arbitrary shape provided some general regularization constraints, such as continuity and smoothness, are satisfied. On the other hand, the parametric models are more constrained because some prior information of the geometrical shape is incorporated. Compared with the free-form deformable models, it has been demonstrated that the parametric models are more robust to irrelevant structures and occlusions when being applied to detect specific shapes of interest.

The parametric models, such as deformable templates/models (4; 3; 5; 6; 7), G-Snake (8) and Active Shape Model (ASM) (9), encode specific characteristics of a shape and its variations using global shape model, which is formed by a set of feature parameters or well defined landmark/boundary points of that shape. A quite successful and versatile scheme in this field is statistics-based shape models in Bayesian framework (3; 8). In these models, the prior knowledge of the object as well as the observation statistics are utilized to define the optimal Bayesian estimate. However, most of these existing parametric models encode the shape information in a “hard” manner in that the prototype contour is fixed during the matching process. As a result, only a small amount of local deformations can be tolerated.

The Active Shape Model (ASM) encodes the prior information of object shapes using Principal Component Analysis (PCA), and matches object by directly using the geometrical transformed version of the ASM model. In ASM, the prototype used to match object can be dynamically adjusted in the matching procedure, which is constrained by the prior distribution of sample data. Therefore, some global/large shape variations that present in the samples can be tolerated.

However, ASM takes the reconstructed object model, which matches the image best, as the matching results. It may not be able to match a new image accurately if the variations of the shape are not presented in the sample data, or if the number of model modes has been truncated too severely. Therefore, to match a new object more accurately, a new method with the name of Bayesian Shape Model (BSM) is proposed to deal with not only the global deformations defined by the prior distribution of object shapes, but also the local random variations.

In the BSM, the prior distribution of object shapes, which reflects the global shape variations of the object, is estimated from the sample data. In the matching procedure, this prior distribution is used to constrain the dynamically adjustable prototype. In this way, large shape deformations due to the variations of samples can be tolerated. Moreover, since the shapes subject to some transformations between the shape space and the image space, such as affine transformation, it is expected that the algorithms developed should be able to deal with the rotation, translation, scaling and even shearing. Therefore, an affine invariant internal energy term is introduced in the BSM to describe mainly the local shape deformations between the prototype contour in the shape domain and the deformable contour in the image domain. Because the deformable contour used to match object has been modeled as the transformed and deformed version of the prototype contour, which can also be dynamically adjusted to adapt itself to the shape variations using the information gathered from the matching process, the proposed BSM has the advantage of matching objects with both global and local variations.

ASM has been demonstrated to be one of the most successful algorithms (9; 10) in the applications of facial feature extraction. Unfortunately, it may not match the local shape variation accurately, and it is also dependent on sample data and image background: in case the background of the input image is different from the training samples, undesirable results are observed. All these shortcomings limit the applications of the ASM in facial feature extraction, where not only global but also local shape variations are presented, and the interested face is generally captured in a complex environment.

Comparing to ASM, the BSM considers both the global and local shape deformations, and it is also affine invariant and less dependent on background. Therefore, the BSM for extraction of frontal facial features is studied in this paper.

Unlike the other applications, in facial feature extraction, there are two specific characteristics to be considered. First, the algorithm should be able to distinguish different expressions, thus both global variations of the whole face and the individual shape variations of separate features have to be taken into account. Second, the algorithm should be efficient to obtain the face features for further processing, *e.g.*. extract the face patch, and perform warping and normalization.

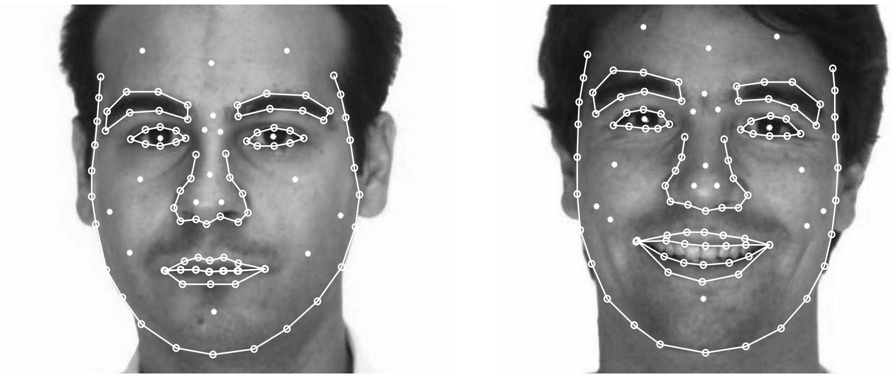


Fig. 1. Two examples of the manually marked face. 'o': 88 contour points, '*': 18 control points.

Generally, the feature points used for matching are the contours of facial features, *i.e.* the contours of the outlines of the face, two eyebrows, two eyes, one nose and one mouth (see Fig.1). Although these points have been proven to be enough for describing the shape of a face, they are insufficient in dealing with face image warping/normalization (10). To give an example, the normalized face patch, or the warped face patch that removes the expressions are quite useful for face recognition (by using the Eigenface algorithm (11)). In order to deal with the face patch warping, in (10), the Delauney triangulation is utilized to partition the interior region that is surrounded by the contour points into a set of triangles. However, this algorithm is depended on or sensitive to the position and shape of the contour points. It does not consider the characteristics of the expression/action of a face, thus the triangles obtained differ greatly when the position of the contour points changes.

To solve this problem, in this paper a 2-D full-face model is built based on the *CANDIDE* face model (12) and the facial feature model in MPEG-7 proposals. A set of points are recruited to represent the shape of a whole face. The full-face model consists of two sets: the contour points and the control points (see Fig.1). The contour points represent mainly the outline contours of facial components, *i.e.* the landmark points of the face outline, eyebrows, eyes, nose and mouth; the control points are the feature points that placed in a number of key positions on the face being used to formulate the triangular face units for facial image warping. Based on a number of manually marked training samples, the prior distribution of the full-face model is obtained by using the PCA, and the individual shape model of each facial component contour is also built. In this way, not only the separate shape variation of each component contour of the facial features, but also the global shape variation can be represented conveniently.

Since the control points are not corresponding to some salient features such as edges of the image, only the contour points of the full-face model are utilized for matching. The position of the control points can be estimated from that of the contour points based on their joint prior distribution. This estimation is possible since the points of the full-face model consist of redundant information.

Then, the BSM is utilized for facial feature extraction based on the proposed full-face model: given an input face image, the contour points of the full-face model are incorporated into the BSM matching framework. Finally, the control points are estimated from the matching results of the BSM according to the prior distribution of the full-face model. The face patch is then extracted from the input image, and further operations such as facial image warping and normalization, sampling, or facial expression cancellation, can be performed to the extracted face patch very easily by using the piece-wise affine algorithm.

Using a number of face images download from the AR face database (13), comparative study was carried out by extracting faces with various expressions and backgrounds using the ASM and BSM. The experiment results show that better performance is achieved by using the BSM.

The rest of the paper is organized as follows: Section 2 introduces the BSM algorithm, Section 3 builds the statistical full-face model using PCA and describes the estimation algorithm of the control points. In Section 4, the BSM algorithm for facial feature extraction is introduced, and the comparative and experimental results between the ASM and BSM are presented in Section 5. Section 6 summaries the conclusion from this study.

2 Bayesian Shape Model (BSM)

2.1 Bayesian Framework and Energy Terms

The matching of a deforming contour to the object in a given image can be formulated as maximizing *a posteriori* (MAP) estimation. Denote the mean of the sample contours in the shape domain as $\bar{\mathbf{f}}_0$ (the mean contour), the deformed version of $\bar{\mathbf{f}}_0$ as $\bar{\mathbf{f}}$ (the prototype contour), and the deformable contour in the image domain as $\mathbf{f}$, where $\bar{\mathbf{f}}_0 \in R^{2N\times 1}$, $\bar{\mathbf{f}} \in R^{2N\times 1}$ and $f \in R^{2N\times 1}$ are the matrices representing the corresponding contours formed by the coordinates of N landmark/boundary points. According to the Bayesian estimation, the joint posterior distribution of $\mathbf{f}$ and $\bar{\mathbf{f}}$, $p(\mathbf{f}, \bar{\mathbf{f}}|d)$, is (14)

$$p(\mathbf{f}, \bar{\mathbf{f}}|d) = \frac{p(d|\mathbf{f})p(\mathbf{f}, \bar{\mathbf{f}})}{p(d)}, \tag{1}$$

where $p(d|\mathbf{f}) = p(d|\mathbf{f}, \bar{\mathbf{f}})$ is the likelihood of input image data d.

$$p(\mathbf{f}, \bar{\mathbf{f}}) = p(\mathbf{f}|\bar{\mathbf{f}})p(\bar{\mathbf{f}}) \tag{2}$$

is the joint prior distribution of $\mathbf{f}$ and $\bar{\mathbf{f}}$. For a given image d, the MAP estimates, $\mathbf{f}_{MAP}$ and $\bar{\mathbf{f}}_{MAP}$, can be defined as

$$\begin{aligned}\{\mathbf{f}_{MAP}, \bar{\mathbf{f}}_{MAP}\} &= \arg\max_{\mathbf{f},\bar{\mathbf{f}}} \{p(\mathbf{f}, \bar{\mathbf{f}}|d)\} \\ &= \arg\max_{\mathbf{f},\bar{\mathbf{f}}} \{\frac{p(d|\mathbf{f})p(\mathbf{f}|\bar{\mathbf{f}})p(\bar{\mathbf{f}})}{p(d)}\}\end{aligned} \tag{3}$$

Note the Bayesian framework is an MAP estimation of the joint prior distribution of $\mathbf{f}$ and $\bar{\mathbf{f}}$, which has some advantages than the classical MAP estimation. In the classic MAP (9), the objective is to find out a prototype contour $\bar{\mathbf{f}}$ in the shape domain, which matches the input image data best subjected to the transformations of translation, rotation and scaling. The disadvantage of this classic estimation is that some local or random shape variations can not be tolerated. In Eq.(3), both $\mathbf{f}_{MAP}$ and $\bar{\mathbf{f}}_{MAP}$ are estimated, and the result contour $\mathbf{f}_{MAP}$ can be different from its prototype $\bar{\mathbf{f}}_{MAP}$ in shape. Two kinds of shape variations are considered, $p(\bar{\mathbf{f}})$ models the prior distribution of the the shape variation and reflects major shape variations of the object shape, and $p(\mathbf{f}|\bar{\mathbf{f}})$ considers the local shape variations between $\mathbf{f}$ and $\bar{\mathbf{f}}$. Therefore, the joint MAP estimation can match object shapes more accurately than the classic MAP estimation.

When the densities can be modeled as Gibb's distribution, *i.e.*

$$\begin{aligned} p(\bar{\mathbf{f}}) &= Z_1^{-1} \exp\{-E_{con}(\bar{\mathbf{f}})\} \\ p(\mathbf{f}|\bar{\mathbf{f}}) &= Z_2^{-1} \exp\{-E_{int}(\mathbf{f}|\bar{\mathbf{f}})\} \\ p(d|\mathbf{f}) &= Z_3^{-1} \exp\{-E_{ext}(d|\mathbf{f})\} \end{aligned} \tag{4}$$

where Z_1, Z_2 and Z_3 are the partition functions, maximizing the posterior distribution is equivalent to minimizing the corresponding energy function of the contour:

$$\{\mathbf{f}_{MAP}, \bar{\mathbf{f}}_{MAP}\} = \arg\min_{\mathbf{f},\bar{\mathbf{f}}} \{E_{BSM}\} \tag{5}$$

where $E_{BSM} = \lambda_{con}E_{con} + \lambda_{int}E_{int} + \lambda_{ext}E_{ext}$. $E_{con} = E_{con}(\bar{\mathbf{f}})$ is the constraint energy term of the adjustable prototype contour $\bar{\mathbf{f}}$, which limits the variations of $\bar{\mathbf{f}}$ and ensures that $\bar{\mathbf{f}}$ is similar with $\bar{\mathbf{f}}$ in shape. $E_{int} = E_{int}(\mathbf{f}|\bar{\mathbf{f}})$ is the internal energy term that describes the global and local shape deformation between $\mathbf{f}$ and $\bar{\mathbf{f}}$. The external energy term $E_{ext} = E_{ext}(d|\mathbf{f})$ defines the degree of matching between $\mathbf{f}$ and the salient image features. λ_{con}, λ_{int} and λ_{ext} are the manually set weighting parameters to regularize the energy terms. They are consistent for all the tests in an experiment or application.

2.2 Constraint Energy of the Prototype Contour

The constraint energy term E_{con} of the prototype contour is caused by the prior distribution of the samples in the shape domain. The density of $\bar{\mathbf{f}}$, $p(\bar{\mathbf{f}})$, can be estimated by applying PCA to the sample contours. In cases where all the samples are aligned views of similar objects seen from a standard view, this distribution can be accurately modeled by a single Gaussian distribution (15):

$$p(\bar{\mathbf{f}}) = \frac{\exp(-\frac{1}{2}\sum_{i=1}^{M}\frac{w_i^2}{e_i})}{(2\pi)^{M/2}\prod_{k=1}^{M} e_k^{1/2}}, \tag{6}$$

where

$$\mathbf{w} = \Phi_M^T(\bar{\mathbf{f}} - \bar{\mathbf{f}}_0) \tag{7}$$

is the vector of the shape parameters, and $\bar{\mathbf{f}} - \bar{\mathbf{f}}_0$ is the deformation from $\bar{\mathbf{f}}_0$ to $\bar{\mathbf{f}}$. Φ_M is the matrix composed of the eigenvectors corresponding to the largest M eigenvalues e_i, $(1 \leq i \leq$

M) of the covariance matrix Σ, and Σ is calculated by

$$\Sigma = \frac{1}{D-1}\sum_{i=1}^{D}(\bar{\mathbf{f}}_i - \bar{\mathbf{f}}_0)\cdot(\bar{\mathbf{f}}_i - \bar{\mathbf{f}}_0)^T. \tag{8}$$

In Eq.(8), D is the number of sample data. Using the PCA, a prototype contour can be reconstructed from $\bar{\mathbf{f}}_0$ and a given shape parameter $\mathbf{w}$,

$$\bar{\mathbf{f}} = \bar{\mathbf{f}}_0 + \Phi_M \mathbf{w}. \tag{9}$$

The PCA representation preserves the major linear correlations of the sample shapes and discards the minor ones, hence provides an optimized approximation of $\bar{\mathbf{f}}$ in the sense of least squares error. This representation describes the most significant modes of the shape variations or the global shape deformations subjecting the prior distribution of the prototype contour. From Eq.(6), the corresponding constraint energy is denoted as

$$E_{con} = \frac{1}{2}\sum_{i=1}^{M}\frac{w_i^2}{e_i} \tag{10}$$

The variation of the prototype is limited by the plausible area of the corresponding shape parameter $\mathbf{w}$, which is defined as

$$\sum_{i=1}^{M}\frac{w_i^2}{e_i} \le M_t \tag{11}$$

The threshold, M_t, may be chosen using the χ^2 distribution (9). The constraint energy term ensures that the dynamically adjustable prototype contour remains similar with the mean shape during the matching process, and at the same time, large shape variations and deformations subject to the prior distribution of the samples can be tolerated.

2.3 Affine-Invariant Internal Energy

An affine-invariant internal energy term, $E_{int}(\mathbf{f}|\bar{\mathbf{f}})$, is defined and incorporated to deal with the affine transformations between the shape domain and the image domain. It describes the global and local shape deformations between $\bar{f}$ and f. Mathematically, $\bar{\mathbf{f}}$ and $\mathbf{f}$ are related by $\mathbf{f}_i = T(\bar{\mathbf{f}}_i) + \boldsymbol{\epsilon} = A\bar{\mathbf{f}}_i + \mathbf{t} + \boldsymbol{\epsilon}$, $(1 \le i \le N)$, where A is a 2×2 nonsingular matrix, $\mathbf{t}$ is a translation vector, and $\boldsymbol{\epsilon}$ represents the random deformation. Define the least squares objective function as

$$E(A, \mathbf{t}) = \sum_{i=1}^{N} [(\bar{\mathbf{f}}_i - A^{-1}(\mathbf{f}_i - \mathbf{t}))^T \cdot (\bar{\mathbf{f}}_i - A^{-1}(\mathbf{f}_i - \mathbf{t}))] \tag{12}$$

the affine transformational parameters can be estimated by

$$\hat{A} = [(F - F_{av})(\bar{F} - \bar{F}_{av})^T] \cdot [(\bar{F} - \bar{F}_{av}) \cdot (\bar{F} - \bar{F}_{av})^T]^{-1} \tag{13}$$
$$\hat{T} = [\hat{\mathbf{t}} \cdot [1, 1, ..., 1]]_{2\times N} = F_{av} - A\bar{F}_{av} \tag{14}$$

where F and $\bar{F}$ are the $2 \times N$ matrices corresponding to $\mathbf{f}$ and $\bar{\mathbf{f}}$ respectively, and $\bar{F}_{av} = [(\frac{1}{N}\sum \bar{\mathbf{f}}_i) \cdot [1, 1, ..., 1]]_{2\times N}$ and $F_{av} = [(\frac{1}{N}\sum \mathbf{f}_i) \cdot [1, 1, ..., 1]]_{2\times N}$. Matrix $(\bar{F} - \bar{F}_{av}) \cdot (\bar{F} - \bar{F}_{av})^T$ is always nonsingular provided there exist at least three points in $\bar{\mathbf{f}}$, which are not located in the same line.

Let $(A_1,\ \mathbf{t}_1)$ be the estimated transformational parameters between $\bar{\mathbf{f}}$ and $\mathbf{f}$, if $\mathbf{f}$ is affine-transformed to $\mathbf{f}'$, *i.e.* $\mathbf{f}_i' = A'\mathbf{f}_i + \mathbf{t}'$, $1 \leq i \leq N$, according to Eq.(13) and Eq.(14), the estimated transformational parameters between $\bar{\mathbf{f}}$ and $\mathbf{f}'$ will be $(A'A_1, A'\mathbf{t}_1 + \mathbf{t}')$. Therefore, the values of the objective function remain unchanged under affine transformations, and hence the global internal energy term of the deformable contour is designed as,

$$E_{gint}(\mathbf{f}|\bar{\mathbf{f}}) = \frac{1}{N}\sum_{i=1}^{N} [(\bar{\mathbf{f}}_i - \hat{A}^{-1}(\mathbf{f}_i - \hat{\mathbf{t}}))^T \cdot (\bar{\mathbf{f}}_i - \hat{A}^{-1}(\mathbf{f}_i - \hat{\mathbf{t}}))], \tag{15}$$

where the transformational parameters $(\hat{A}, \hat{\mathbf{t}})$ are calculated by Eq.(13) and Eq.(14). This internal energy indicates the global matching degree between the deformable contour $\mathbf{f}$ and the prototype contour $\bar{\mathbf{f}}$.

In addition, a local internal energy term is defined by affine invariants: the proportion of area (16),

$$E_{lint}(\mathbf{f}_i|\bar{\mathbf{f}}) = \frac{(S_1 + S_2)AREA_{proto}}{AREA_{aligned}} \tag{16}$$

$S_1 = S(\bar{\mathbf{f}}_{i-1}, \mathbf{f}_i^a, \bar{\mathbf{f}}_i)$ and $S_2 = S(\bar{\mathbf{f}}_i, \mathbf{f}_i^a, \bar{\mathbf{f}}_{i+1})$, where $S()$ is the area of the triangle formed by the three points inside the brackets. $AREA_{proto}$ and $AREA_{aligned}$ represent the interior areas formed by the hull of the prototype contour and the aligned deformable contour $(\mathbf{f}_i^a = \hat{A}^{-1}(\mathbf{f}_i - \hat{\mathbf{t}}))$ in the shape domain respectively. Fig.2 shows the geometrical relationship between S_1 and S_2. It can be seen from the figure that, when the areas of the triangles S_1 and S_2 are close to zero, the shape and position of $\bar{\mathbf{f}}$ and $\mathbf{f}$ will also be close. When the contour represents an open shape, the local internal energy at the end point of the contour will not be calculated.

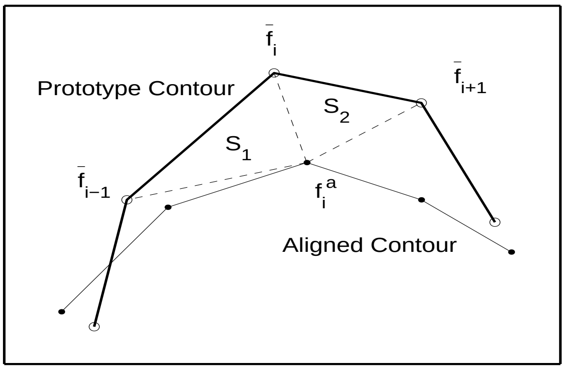


Fig. 2. A part of the prototype contour $\bar{\mathbf{f}}$ and the aligned deformable contour $\mathbf{f}^a$.

In summary, the affine-invariant internal energy is composed of both the global and local terms,

$$E_{int}(\mathbf{f}|\bar{\mathbf{f}}) = E_{gint}(\mathbf{f}|\bar{\mathbf{f}}) + \frac{1}{N}\sum_{i=1}^{N} E_{lint}(\mathbf{f}_i|\bar{\mathbf{f}}), \tag{17}$$

which reflects the degree of fitting between $\bar{\mathbf{f}}$ and $\mathbf{f}$.

2.4 External Image Constraint

The external energy term $E_{ext} = E(d|\mathbf{f})$ indicates the degree of matching between the deformable contour $\mathbf{f}$ and the salient image features. Minimizing E_{ext} adjusts $\mathbf{f}$ and moves it towards the object boundary in the image d. The external energy usually combines all the information of edge, texture, color and region, so that it can provide an effective description of the matching. For example, the color information can be combined into the edge detection process, so that the edge maps will accurately stand for the boundaries of the interested objects. Among various matching rules and external energy terms used in the literature, the energy term including both the gradient and directional edge information is utilized because of its simplicity and efficiency (6).

First, the image $d = \{d(x, y)\}$ is smoothed using Gaussian function, $G_\sigma(x, y)$, with the deviation σ:

$$d_\sigma(x,y) = G_\sigma(x,y) * d(x,y). \tag{18}$$

Second, the normalized gradient of the smoothed image d_σ at each pixel location (x,y), denoted as $\mathbf{d}^g_\sigma(x,y) = [d^g_{\sigma x}(x,y), d^g_{\sigma y}(x,y)]^T$, $(||\mathbf{d}^g_\sigma(x,y)|| \in [0,1])$, is computed.

At last, the constraint on $\mathbf{f}$ ensures that $\mathbf{f}$ moves towards the object boundaries: when the image pixels along the contour have high gradient magnitude, and the direction of the image gradient along the contour is perpendicular to the orientation of the contour, the external energy is small. Therefore, the external energy function can be defined as

$$E_{ext}(d \mid \mathbf{f}) = \sum_{i=1}^{N}(1 - ||\mathbf{d}^g_\sigma(x_i,y_i)||)|\mathbf{n}(x_i,y_i)\cdot\mathbf{h}(x_i,y_i)|, \tag{19}$$

where "·" is the dot product. $\mathbf{h}(x,y)$ is the direction of the gradient $\mathbf{d}^g_\sigma(x,y)$, $\mathbf{h}(x,y) = \mathbf{d}^g_\sigma(x,y)/||\mathbf{d}^g_\sigma(x,y)||$ and $||\mathbf{h}(x,y)|| = 1$. $\mathbf{n}(x_i,y_i)$ indicates the normal vector of the contour $\mathbf{f}$ at point $\mathbf{f}_i = (x_i,y_i)$, with $||\mathbf{n}(x_i,y_i)|| = 1$ and $\mathbf{n}(x_i,y_i) = [\begin{smallmatrix}0 & -1\\ 1 & 0\end{smallmatrix}]\mathbf{v}_i/||\mathbf{v}_i||$, where $\mathbf{v}_i = \frac{\mathbf{f}_{i+1}-\mathbf{f}_i}{||\mathbf{f}_{i+1}-\mathbf{f}_i||} + \frac{\mathbf{f}_i-\mathbf{f}_{i-1}}{||\mathbf{f}_i-\mathbf{f}_{i-1}||}$ is the tangent vector of contour $\mathbf{f}$ at point $\mathbf{f}_i$.

3 Modeling the Full-Face Using Principle Component Analysis

3.1 The Full-Face Model and Its Prior Distribution

In this section the full-face model is proposed and the prior distribution of the whole face and the separate component models are estimated using PCA. As shown in Fig.1, a full-face model are formed by the boundary/landmark points of the face outline, mouth, nose, eyes and eyebrows, as well as other scattered or isolated points. These points of the full-face model can be divided into two sets:

(1) Contour points: the contour points of the face outline, mouth, nose, eyebrows and eyes; (88 points)

(2) Control points: the points that are very useful for face image warping, but correspond to lack image edge features. (18 points)

Fig.1 shows two examples of the marked face. 'o' is the contour point representing the outlines of the face, eyebrows, eyes, nose and mouth respectively, while '*' stands for the control point. Fig.3 plots several examples of the full-face model, where the contour points and control points are linked to formulate a number of triangles or face units. It can be seen that this plot is similar to the *CANDIDE* face model, but the later is a 3-D face model for plotting, animating and synthesizing faces in various applications such as multimedia.

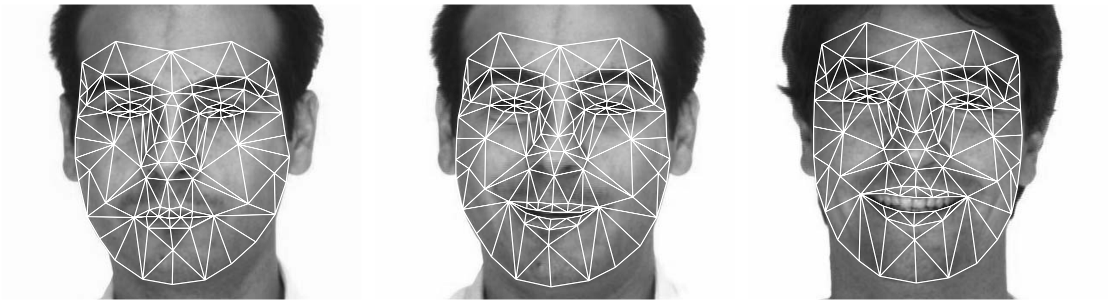


Fig. 3. Link the model points to triangles or face units.

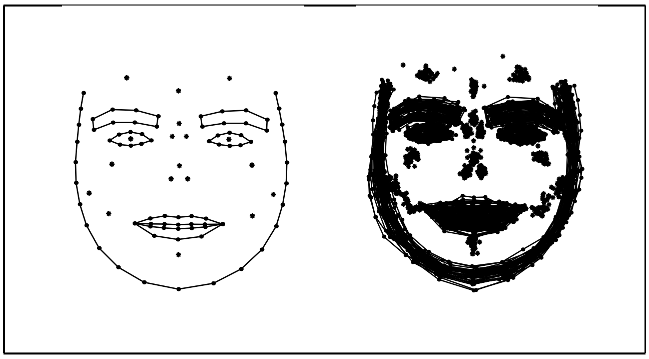


Fig. 4. The mean face model and all the 39 normalized face models.

The full-face model in the shape domain, $\bar{\mathbf{s}}$, can be represented as

$$\bar{\mathbf{s}} = \left[\bar{\mathbf{f}}^T \;\; \bar{\mathbf{c}}^T\right]^T, \tag{20}$$

where $\bar{\mathbf{f}}(1N_f \times 1)$ represents the contour points, $\bar{\mathbf{c}}(2N_c \times 1)$ is the control points, and $N_f + N_c = N$. The contour points of the face consist of 7 component contours of separate facial features: a face outline contour, a mouth, a nose, two eyes and two eyebrows (see Fig.4 and Table 1). These component contours are denoted as $\bar{\mathbf{f}}_i, 1 \leq i \leq 7$, and

$$\bar{\mathbf{f}} = \left[\bar{\mathbf{f}}_1^T \;\; \bar{\mathbf{f}}_2^T \;\; \cdots \;\; \bar{\mathbf{f}}_7^T\right]^T. \tag{21}$$

Table 1

Component contours of the face contour

Component contour	Facial feature	Number of points
$\mathbf{f}_1$	face outline	23
$\mathbf{f}_2$ and $\mathbf{f}_3$	eyebrows	2×8
$\mathbf{f}_4$ and $\mathbf{f}_5$	eyes	2×8
$\mathbf{f}_6$	nose	13
$\mathbf{f}_7$	mouth	20

The PCA is utilized to build the statistic model of the full-face model. First, all the sample face images are manually marked and normalized/aligned to a standard view by using least square errors method. Then, the distribution of the full-face model, the whole contour model as well as the separate component contour models of the face are established from these normalized sample data using PCA. These three kinds of shape models are summarized as follows,

(1) Full-face model ($\bar{\mathbf{s}}$): a hybrid overall distribution model for all the contour points and control points of the full-face in the shape domain.

Using PCA, given a shape parameter $\mathbf{w}_s$, the full-face model can be reconstructed using the following equations,

$$\bar{\mathbf{s}} = \bar{\mathbf{s}}_0 + \Phi_s \mathbf{w}_s \tag{22}$$

and the shape parameter of an input sample, $\bar{\mathbf{s}}$, can be determined by

$$\mathbf{w}_s = \Phi_s^T (\bar{\mathbf{s}} - \bar{\mathbf{s}}_0) \tag{23}$$

where Φ_s is the matrix composed of the eigenvectors corresponding to the largest M_s eigenvalues, which is computed from the covariance matrix of all of the sample data. The distribution of the full-face model can be reasonably modeled by a single Gaussian density (see Eq.(6) to Eq.(11) for detail).

The full-face model is very useful for determining the control points when the contour points are known.

(2) Whole contour model ($\bar{\mathbf{f}}$): the distribution model that takes all the contour points as a

whole.

The shape parameters of the whole contour model can be calculated by

$$\mathbf{w}_f = \Phi_f^T(\bar{\mathbf{f}} - \bar{\mathbf{f}}_0) \tag{24}$$

and also, the whole contour can be synthesized from a given shape parameter $\mathbf{w}_f$,

$$\bar{\mathbf{f}} = \bar{\mathbf{f}}_0 + \Phi_f \mathbf{w}_f \tag{25}$$

The whole contour model integrates all the component contours, which is useful for matching the overall face object.

(3) Component contour model ($\bar{\mathbf{f}}_i, \ \ 1 \leq i \leq 7$) : separate distribution model for each component contour.

Using PCA, the shape parameters of each component contour can be described as

$$\mathbf{w}_i = \Phi_i^T(\bar{\mathbf{f}}_i - \bar{\mathbf{f}}_{i0}), \qquad i = 1, ..., 7, \tag{26}$$

where $\Phi_i, \ (1 \leq i \leq 7)$ are the matrices composed of the eigenvectors corresponding to the largest M_i eigenvalues, which are computed from the covariance matrix of all of the sample data. Then, given the shape parameters $\mathbf{w}_i, \ (1 \leq i \leq 7)$, each component contour can be reconstructed by

$$\bar{\mathbf{f}}_i = \bar{\mathbf{f}}_{i0} + \Phi_i \mathbf{w}_i \tag{27}$$

Note in establishing the above component models, all the sample contours of each component are translated into a standard position before performing PCA, so that the model of each component contour reflects only their own shape variations.

To apply the BSM algorithm for facial feature extraction, first the BSM is used to match all the contour points $\mathbf{f}$ of the face model to the target face in the input image. During the matching procedure, both the whole and separate contour models are utilized to ensure that the facial contours are matched accurately. Finally, the control points are estimated from the matching result of the contour points based on the full-face model, to extract the face patch. Further processes like face patch warping, can be performed very easily.

The method for estimating the control points from the contour points using the prior distribution of the full-face model is proposed in the following section.

3.2 Estimation of the Control Points

Suppose the current deforming contour $\mathbf{f}$ in the image domain and its prototype contour $\bar{\mathbf{f}}$ are known, the corresponding transformational parameters A and T (between $\mathbf{f}$ and $\bar{\mathbf{f}}$, or between F and $\bar{F}$) can be estimated using alignment algorithms (Eq.(13) and Eq.(14)). Then, the contour points can be transformed into the shape domain using,

$$\bar{F}' = A^{-1}(F - T). \tag{28}$$

To estimate the control points corresponding to $\bar{\mathbf{f}}'$ using the full-face distribution model, in this section, a fast approximate algorithm is proposed. The idea is to estimate the shape parameter $\hat{\mathbf{w}}_s$, so that the contour points $\hat{\bar{\mathbf{f}}}$ of the reconstructed full-face model $\hat{\bar{\mathbf{s}}} = \begin{bmatrix}\hat{\bar{\mathbf{f}}}^T & \hat{\bar{\mathbf{c}}}^T\end{bmatrix}^T$ matches $\bar{\mathbf{f}}'$ closely, and hence the corresponding reconstructed control points $\hat{\bar{\mathbf{c}}}$ can be regarded as the estimate of the control points.

The detail procedure is described as follows,

(1) The estimation problem is to find out $\hat{\mathbf{w}}_s$, so that the result $\hat{\bar{\mathbf{f}}}$ matches $\bar{\mathbf{f}}'$ in the sense of least square errors. From Eq.(22) $\hat{\bar{\mathbf{f}}}$ can be calculated by

$$\begin{aligned}\hat{\bar{\mathbf{f}}} &= [I\ \mathbf{0}]\begin{bmatrix}\hat{\bar{\mathbf{f}}}^T & \hat{\bar{\mathbf{c}}}^T\end{bmatrix}^T\\ &= [I\ \mathbf{0}](\bar{\mathbf{s}}_0 + \Phi_s\hat{\mathbf{w}}_s)\\ &= [I\ \mathbf{0}]\bar{\mathbf{s}}_0 + [I\ \mathbf{0}]\Phi_s\hat{\mathbf{w}}_s\end{aligned} \tag{29}$$

where I is a unity matrix and $\mathbf{0}$ is a zero matrix. Noting $\bar{\mathbf{f}}_0 = [I\ \mathbf{0}]\bar{\mathbf{s}}_0$ and denoting $\Phi_f{}' = [I\ \mathbf{0}]\Phi_s$, from $\bar{\mathbf{f}}_0$, $\Phi_f{}'$, $\bar{\mathbf{f}}'$ and Eq.(23), the shape parameter $\hat{\mathbf{w}}_s$ can be estimated using

$$\hat{\mathbf{w}}_s = \Phi_f{}'^T(\bar{\mathbf{f}}' - \bar{\mathbf{f}}_0) \tag{30}$$

(2) The control points $\hat{\bar{\mathbf{c}}}$ can be calculated through $\hat{\mathbf{w}}_s$,

$$\hat{\bar{\mathbf{c}}} = [\mathbf{0}\ I](\bar{\mathbf{s}}_0 + \Phi\hat{\mathbf{w}}_s) \tag{31}$$

(3) The final full-face contour in the shape domain is represented by

$$\hat{\bar{\mathbf{s}}} = \begin{bmatrix}\bar{\mathbf{f}}'^T & \hat{\bar{\mathbf{c}}}^T\end{bmatrix}^T \tag{32}$$

thus its transformed counterpart in the image domain, $\mathbf{s} = \begin{bmatrix}\mathbf{f}^T & \mathbf{c}^T\end{bmatrix}^T$ can be obtained accordingly.

In this way, the extracted face not only consists of the contour points but also the control points, and now it is ready to be warped or normalized by linking these points to triangle face units and using the piece-wise affine warping algorithm.

The above estimation performs very fast, and its calculation begins from the face contour, $\mathbf{f}$, and ends at the control points, $\mathbf{c}$, as,

$$\mathbf{f} \to \bar{\mathbf{f}}' \to \hat{\mathbf{w}}_s \to \hat{\bar{\mathbf{c}}} \to \mathbf{c}. \tag{33}$$

The principle behind this approximate estimation is that the information provided by the contour and control points of a full-face model is redundant, *i.e.* using partial contour information, the whole shape of the full-face can be described approximately.

In order to further evaluate this estimation quantitatively, we test the algorithm using a number of, NUM, manually marked images. The average error or distance between the estimated control points (NUM_C is the number of the control points) and the marked points in the shape domain is used.

$$Error = \frac{1}{NUM \cdot NUM_C} \sum_{NUM} \sum_{i=1}^{NUM_C} ||\bar{\mathbf{c}}_i - \hat{\bar{\mathbf{c}}}_i|| \tag{34}$$

Table 2

Average error of the estimation.

Test images	Number of images	Average error	Error/Scale
Training samples	25	1.3	0.4%
Testing images	25	3.05	1.0%

In the experiment, 50 marked images with different face expressions are used, which is partitioned into two sets. The first set consists of 25 training sample images, while the other uses the marked images not used for training. It can be seen from Table 2 that although more accurate estimation can be obtained from the trained sample images, the average error is also small for the other image set, which are not in the training set. Moreover, it can be seen that the relative errors, *i.e.* the error/scale ratio, are very small with respect to the scale of mean face model in the shape domain, *i.e.* 360 × 294.

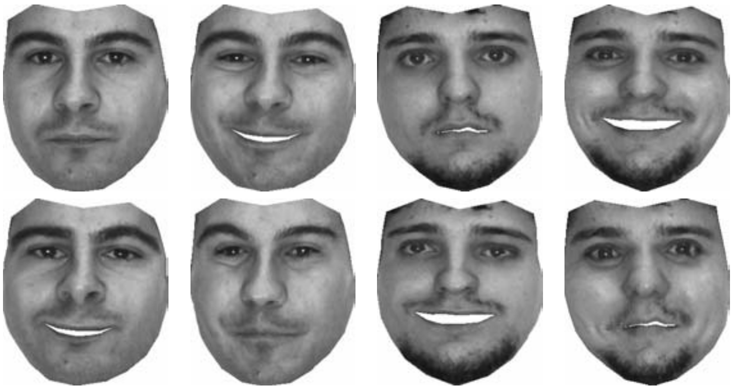


Fig. 5. The original (upper row) and the warped (lower row) face images.

Fig.5 shows several examples of the warping results, where the contour points are known and the control points are estimated by using the proposed algorithm. These full-face models are then utilized for face patch warping. In the figure, the upper row is the original face patches, while the bottom row corresponds to the warped images of the upper row, which changes the expressions of an extracted face patch. For example, the first column of the images illustrates how to change the expression to smile, and the second column is a reverse procedure. It can be seen from the figure that the proposed full-face model describes the face well, and through image warping, the expressions of face can be removed/changed, which is very useful for further face recognition, 2-D face animation, content based face image coding and retrieval in multimedia applications.

4 The Use of the BSM for Extracting Facial Features

The BSM algorithm for extracting facial features is summarized in this section. First we introduce the computation of BSM, and then describe the facial feature extraction algorithm in detail.

4.1 Computation of BSM

The computation of BSM is actually a solution finding procedure that minimizes the energy terms. Since the energy term of BSM is non-convex and very complex, in this study, in order to drive the BSM to the target object using energy minimization, an iterative procedure, incorporating the coarse-to-fine multiresolution searching techniques, is used. The major techniques used include i) iterative shape constraint, which ensure that the deforming contour is in the allowable area according to the prior distribution of object shapes; ii) matching in multiresolution framework, which first matches object in the coarsest image, and then refines the location of the model using a series of finer resolution images; iii) coarse searching, which matches only the contour points corresponding to salient image features, and only considers the global shape variations in the coarser resolution; iv) fine tuning, which matches all the contour points and considers not only the global but also local shape variations. In summary, the solution finding algorithm of BSM is described as follows.

Stage 1 (coarse searching):

Obtain the parameters of $\hat{A}$ and $\hat{T}$ using object detection and feature points alignment (manual initialization method is used in this thesis for simplicity), set the weighting parameters λ_{con}, λ_{int} and λ_{ext}, set the order K of the multi-resolution pyramid, $n = 0$, $\mathbf{w} = \mathbf{0}$, $\bar{\mathbf{f}}^{(n)} = \bar{\mathbf{f}}_0$, the contour $\mathbf{f}^{(n)}$ for object matching in the image domain is initialized as $F^{(n)} = \hat{A}(\bar{F}^{(n)}) + \hat{T}$, and calculate the multiresolution images and edge maps. $k = K - 1$, and $n = 1$.

(1) Examine the image region in a neighborhood of $\mathbf{f}^{(n-1)}$, and find out a new contour $\mathbf{f}^{(n)}$, which obtains minimum external energy (at resolution level k).

(2) Update the parameters of affine transform, $T(\hat{A}, \hat{\mathbf{t}})$, by aligning $\mathbf{f}^{(n)}$ and $\bar{\mathbf{f}}^{(n-1)}$. Calculate $\bar{\mathbf{f}}^{(n)'}$ using Eq.(13) and Eq.(14), and obtain a new prototype $\bar{\mathbf{f}}^{(n)}$ from $\bar{\mathbf{f}}^{(n)'}$, at the same time, make sure that the prototype is a plausible shape according to the prior distribution of object shapes.

(3) Update the current contour $\mathbf{f}^{(n)}$ to $T(\bar{\mathbf{f}}^{(n)})$ because the local shape variation is not taken into account in the coarse searching.

(4) Set $n = n + 1$ and go to step (1) unless the newly updated $\mathbf{f}^{(n)}$ is close to the old one $\mathbf{f}^{(n-1)}$ (the least square error is smaller than a preset threshold ξ.), or the maximum iterations I have been applied.

(5) Set $n = n + 1$ and $k = k - 1$. If $k > 0$, then go to step (i), else exit the coarse stage and go to the fine stage.

Stage 2 (fine matching):

$k = 0$

(1) Update the contour $\mathbf{f}^{(n)}$ by minimizing the internal and external energy terms.

(2) Update the parameters of affine transform $T(\hat{A},\ \hat{\mathbf{t}})$, and apply the shape constraint, *i.e.* update parameters $\mathbf{w}$, and the prototype contour $\bar{\mathbf{f}}^{(n)}$, according to $\mathbf{f}^{(n)}$ (9). The constraint of $\mathbf{w}$ is applied to ensure the prototype contour is in a plausible area.

(3) Repeat step (i) and (ii) until convergence.

4.2 BSM Algorithm for Facial Feature Extraction

In the previous section, we introduced the detailed computation procedure of BSM. In this section, we describe how to use the BSM for facial feature extraction by incorporating the full-face model. There are two stages for the solution finding procedure, ı.e. coarse matching and fine matching.

In the coarse searching stage, since the component contours are relatively far away from their target objects, only the whole contour model $\mathbf{f}$ is utilized. Moreover, the face outline contour is utilized for actively searching and updating the whole face contour.

Since the whole contour model consists of multiple component contour models $\mathbf{f}_i, (i = 1, 2, ..., 7)$, in the fine matching procedure, each component contour model is utilized to match facial features, *e.g.* eyes, mouth, and others separately. After all the newly updated contour points are obtained, a shape constraint is carried out using the whole contour model to ensure that the whole shape of the face is within an affordable area. The sketch of the fine matching is shown in Fig.6

Fig.7 plots an example of the BSM matching result. After the face patches are extracted, an image warping algorithm can be performed to warp the extracted face patch to the mean shape of the face model. The extracted face patch in the shape space from Fig.7 is also plotted in Fig.8, which also shows its warped version to the mean of the face contour. From Fig.8, it can be seen that the features of a face consists of two parts, the shape features and the appearance features, both of which play important role in describing the face. Since the sampling and warping of the appearance is depended on the shape information, accurate matching of the face contour is crucial in face matching and extraction.

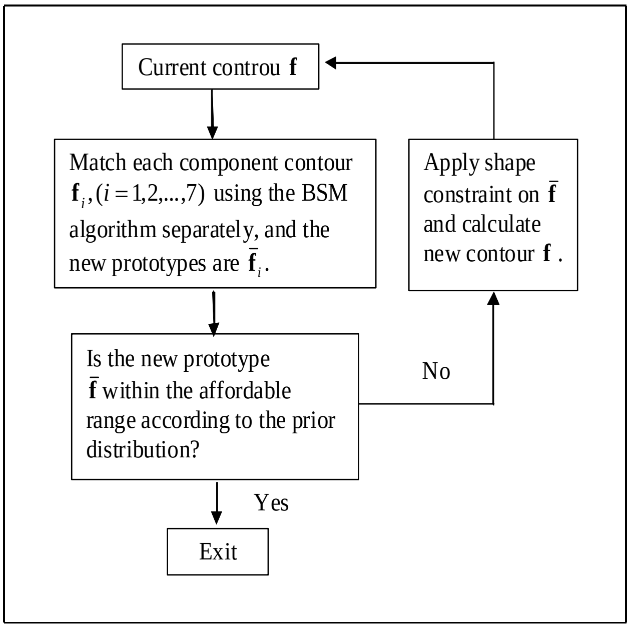
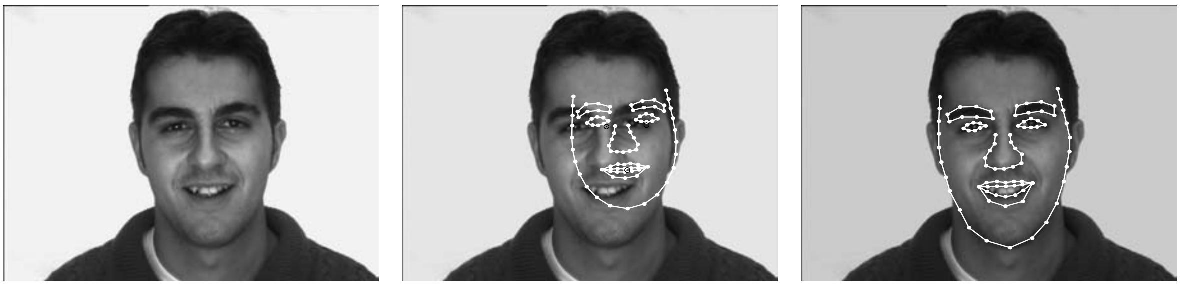

Current controu $\mathbf{f}$
Match each component contour $\mathbf{f}_i,(i=1,2,...,7)$ using the BSM algorithm separately, and the new prototypes are $\bar{\mathbf{f}}_i$.
Apply shape constraint on $\bar{\mathbf{f}}$ and calculate new contour $\mathbf{f}$.
Is the new prototype $\bar{\mathbf{f}}$ within the affordable range according to the prior distribution?
No
Yes
Exit

Fig. 6. The fine matching stage of BSM for facial feature extraction.



input image initial matching result

Fig. 7. Facial feature extraction using BSM.

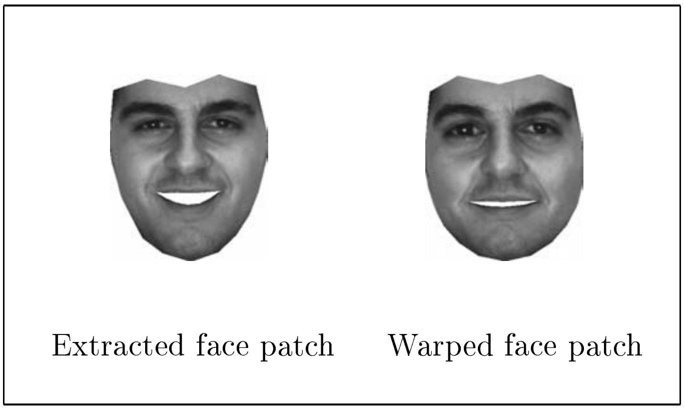


Fig. 8. The extracted face patch and its warped counterpart to the mean facial shapes (all patches are in the shape domain).

5 Experimental Results and Discussion

In this section, experiments are carried out to compare the ASM and BSM algorithms for facial feature extraction, and the comparison results are analyzed. More results using the BSM algorithm are further presented to demonstrate its effectiveness in facial feature extraction.

5.1 Performance Comparison Between ASM and BSM

For comparison purpose, the training sets and the initial positions of the face model used in the experiment are the same for both of the algorithms. The sample face images used for training are selected from the AR frontal face database (13) (http://rvl1.ecn.purdue.edu/ ∼aleix/aleix _face_DB.html), which consists lots of male and female frontal face images with various expression. In our experiment, 39 sample contours are manually marked, and aligned/normalized using least squares error method to constitute the shape space, and Fig.4 plots the normalized sample contours and the mean contour.

The face images used for testing include two sets, the first set is also chosen from the AR face database, which consists of the face images that are different from the training images, the second testing set are the face images with different and complex background from the training set.

In the comparative experiments, both of the ASM and BSM are used to match the selected sample face images. Altogether 20 different face images, each one has three different expressions, are recruited for testing purpose.

The initial contour for each image is set as the transformed version of the mean face contour. Suppose the transformation between the target face contour and the mean face contour is

$$\mathbf{f}^d = \mathbf{s}_0 \begin{bmatrix} \cos\theta_0 & -\sin\theta_0 \\ \sin\theta_0 & \cos\theta_0 \end{bmatrix} \bar{\mathbf{f}}_0 + \begin{bmatrix} t_{01} \\ t_{02} \end{bmatrix} \tag{35}$$

where $\mathbf{f}^d$ is the target face contour and $\bar{\mathbf{f}}_0$ represents the mean contour. The transformational parameters used to initialize the contour is chosen by randomly varying the parameters, *i.e.* $\mathbf{s}_0$, θ_0, t_{01} and t_{02}, by [-15%, 15%]. For each testing image, 10 random initial contours are generalized, and therefore altogether 600 tests are carried out using BSM and ASM respectively. In this paper, the results of ASM are obtained by using the ASM Toolkit (Version 1.0) of the Visual Automation Ltd (http://www.wiau. man.ac.uk/VAL).

Table 3

Average errors of matching results of ASM and BSM (Unit in pixel)

Algorithm	ASM	BSM
Average error of initial	29.3	29.3
Average error of results	4.8	2.1
Error/scale	1.7	0.7

Some examples of the initial positions and the matching results using ASM and BSM are demonstrated in Fig.9. It can be visually seen from the figures that the matching results of BSM is more closer to the boundaries of the facial features.

To study the matching result quantitatively, the distance or error between the deformable contour $\mathbf{f}$ to its target position (marked manually), $ERROR$, is defined in the shape domain,

$$ERROR = \frac{1}{N}\sum_{i=1}^{N} ||T^{-1}(\mathbf{f}_i) - \bar{\mathbf{f}}_i^d|| \tag{36}$$

where $T(\cdot)$ is the transformation of the current contour between the shape domain and the image domain, and $\bar{\mathbf{f}}_i^d$ is the target object marked manually. Note the $ERROR$ is transformational invariant because it is defined in the shape domain, and hence it is reasonable to use $ERROR$ to justify the matching accuracy of different input images. The values of $ERROR$ for the matching results are listed under each face image in Fig.9. It is clear that although using either ASM or BSM, the target face contour can be matched, the BSM obtains better performance than ASM, *i.e.* less matching error. For example, it can be seen from the first row of Fig.9 that using ASM, the $ERROR$ is 2.1, while using BSM, the $ERROR$ is only 1.3, which is much less than that of ASM.

Table 3 shows the average initial error and the average matching errors of the ASM and BSM for all the 600 tests. It can be seen from the table that BSM obtains a better accurate matching results than the ASM, *i.e.* with less matching $ERROR$.

In real applications of face matching, the target objects may exist in different backgrounds. Therefore, the second set of the experiment is carried out to compare the performance of ASM and BSM for matching the images whose background different from that of the training samples. It worth noting that since the image background is quite different from case to case, the ASM often obtains diverse matching results because of the background dependence. Fig.10 illustrates one example of ASM when the testing image background is different from that of the training sample. Fig.10(a) plots the initial contour, and in the experiment we found that it is quite difficult for the ASM to obtain satisfied result. One of the matching results of ASM is given in 10(b), it is seen that the matching result actually diverges. To avoid the divergence, we tried another way by fixing three face outline points manually onto the target position, *i.e.* the three points can not move during the matching procedure. However, experiment shows the ASM still can not obtain a satisfied result (see 10(c)). After a number of iteration, it can not converge to the target contour and instead it intends to move to other places of the image. On the contrary, since BSM has an advantage of background independence, it is more adaptable for the applications of facial feature extraction.

Finally, it worth noting that because BSM considers the shape deformation between the deforming contour and the prototype contour, its computational speed is not as fast as that of the ASM. However, the experiments show that the speed of BSM is still acceptable, which performs at around 0.136 seconds per iteration for the 88 points' face model (the contour points only), and generally the algorithm converges after around 15 iterations using the PC with 450MHz Intel Pentium processor.

5.2 More Experimental Results Using BSM Facial Feature Extraction Algorithm

In practice, before matching and extracting the face features, the users are expected to detect the face from the input image. For example, in (17), a template matching strategy is used, and the rotations are dealt with by enumerations of various rotated versions of the template. To further demonstrate the effectiveness of BSM, in this section, BSM are used to match the frontal faces with different rotation angles, *i.e.* the face images with different expressions, different rotation angles, and complex image background are utilized as the input images. Some of these images are chosen from the database of the Vision and Autonomous System Center (VASC) of CMU(http://www.cs.cmu.edu/afs/cs.cmu.edu/user/har/Web/faces.html), while the others are captured by ourselves using USB camera in the laboratory environment.

Fig.11 shows several matching results using BSM. The extracted face patches and their warped versions to the mean facial shape are shown in Fig.12. It can be seen from Fig.12 that besides the shape information of a face, the interior gray level information is also very important for face recognition, face animation, and even multimedia applications. In the warped face patches, even there is no difference in shape, from the appearance information alone, one can recognize different person easily. As a conclusion, using the BSM facial feature extraction algorithm, not only the shape of the face, but also the face patch can be extracted.

From experiments we found that good results are observed for face matching when the resolution is larger or around 150×150 points for a face. A simple facial feature extraction system is implemented to demonstrate the performance of BSM. The system uses the proposed full-face model to describe the face contour, and the distribution parameters of the model, *i.e.* the mean face contour and the shape parameters of the shape variation, are trained offline, and the training samples are the images from the AR face database.

The operational sequence of the system is: first capture an image from the USB video camera or load an existing face image file from disk, and manually initialize the face contour so that it is close to the target face, then the final matching result of the input face image is iteratively obtained by the software, finally, the face patch is extracted and its warped version to the mean face contour in the shape domain is also calculated. Fig.13 shows the dialog to capture an image from the input USB video, and the matching results are given in Fig.14.

In this system, the initialization of the contour is manually set for simplicity. The initialization is done by manually select three points in the input image (the three points that we select are the center points of the left, right eyes and the mouth). Then the corresponding transformational parameters of the initial contour can be found by aligning the selected three points with their corresponding three points in the shape domain. At last, the initial contour is set as the transformed version of the mean face contour.

Fig.15 and Fig.16 plot some matching results of the face images, including the initial contours, the matching results and the extracted and warped/normalized face patches. The resolution of these images is 320×240, and the size of the face is approximately half of the image size. It can be seen from the figures that good matching and extraction results are obtained. The good matching results using BSM on the face images captured from USB camera further indicate the potential applications of the BSM facial feature extraction algorithm. For example, in face recognition system, BSM can be used to accurately match and extract the face patches, then the classification based on the shape and the appearance information of the extracted face patch, such as EigenFace algorithm (11) can be performed. In face image indexing, coding and multimedia systems, BSM facial feature extraction algorithm can also be utilized to extract the interested face patches accurately as long as the prior face model is trained and the face can be detected, located properly close to the target face position.

6 Conclusion

The facial feature extraction using the Bayesian Shape Model is studied in this paper. A full-face model consisting the contour points and the control points are developed to describe the shape of a face. The full-face model is suitable for extracting face patch warping, as well as simplifying sample of the appearance. Experimental results demonstrate that as compared with ASM, more accurate matching result can be achieved by using BSM. Therefore, BSM is suitable for the face matching applications where the matching accuracy is crucial. The potential applications of BSM facial feature extraction algorithm include face recognition, face video coding and retrieval, face animation and multi-media.

References

[1] Z. Xue, S.Z. Li, and E.K. Teoh, "Ai-eigensnake: An affine-invariant deformable contour model for object matching," *Image and Vision Computing*, vol. 20, no. 2, pp. 77–84, 2002.

[2] Z. Xue, S.Z. Li, and E.K. Teoh, "Facial feature extraction and image warping using pca based statistic model," in *IEEE International Conference on Image Processing (ICIP2001), Thessaloniki, Greece*, 2001.

[3] A.K. Jain, Y. Zhong, and S. Lakshmanan, "Object matching using deformable templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 3, pp. 267–278, 1996.

[4] W. Zhao and R. Chellappa, "Sfs based view synthesis for robust face recognition," in *4th IEEE Conference on Automatic Face and Gesture Recognition, Grenoble, France*, 2000, pp. 285 –292.

[5] S. Lakshmanan and D. Grimmer, "A deformable template approach to detecting straight edges in radar images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 438–443, 1996.

[6] A.L. Yuille, P.W. Hallinan, and D.S. Cohen, "Feature extraction from faces using deformable templates," *International Journal of Computer Vision*, vol. 8, no. 2, pp. 99–111, 1992.

[7] L.H. Staib and J.S. Duncan, "Boundary finding with parametrically deformable models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1061–1075, 1992.

[8] K.F. Lai and R.T. Chin, "Deformable contours: modeling and extraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 11, pp. 1084–1090, 1995.

[9] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham, "Active shape models - their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[10] T. Cootes, G.J. Edwards, and C.J. Taylor, "Active appearance models," in *Proceeding of 5th European Conference on Computer Vision*, 1998, vol. 2, pp. 484–498.

[11] A.T. Matthew and A.P. Pentland, "Eigenfaces for recognition," *Journal of Congnitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[12] M. Rydfalk, *CANDIDE, a parameterized face*, Report No. LiTH-ISY-I-866, Dept. of Electrical Engineering, Linkping University, Sweden, 1987.

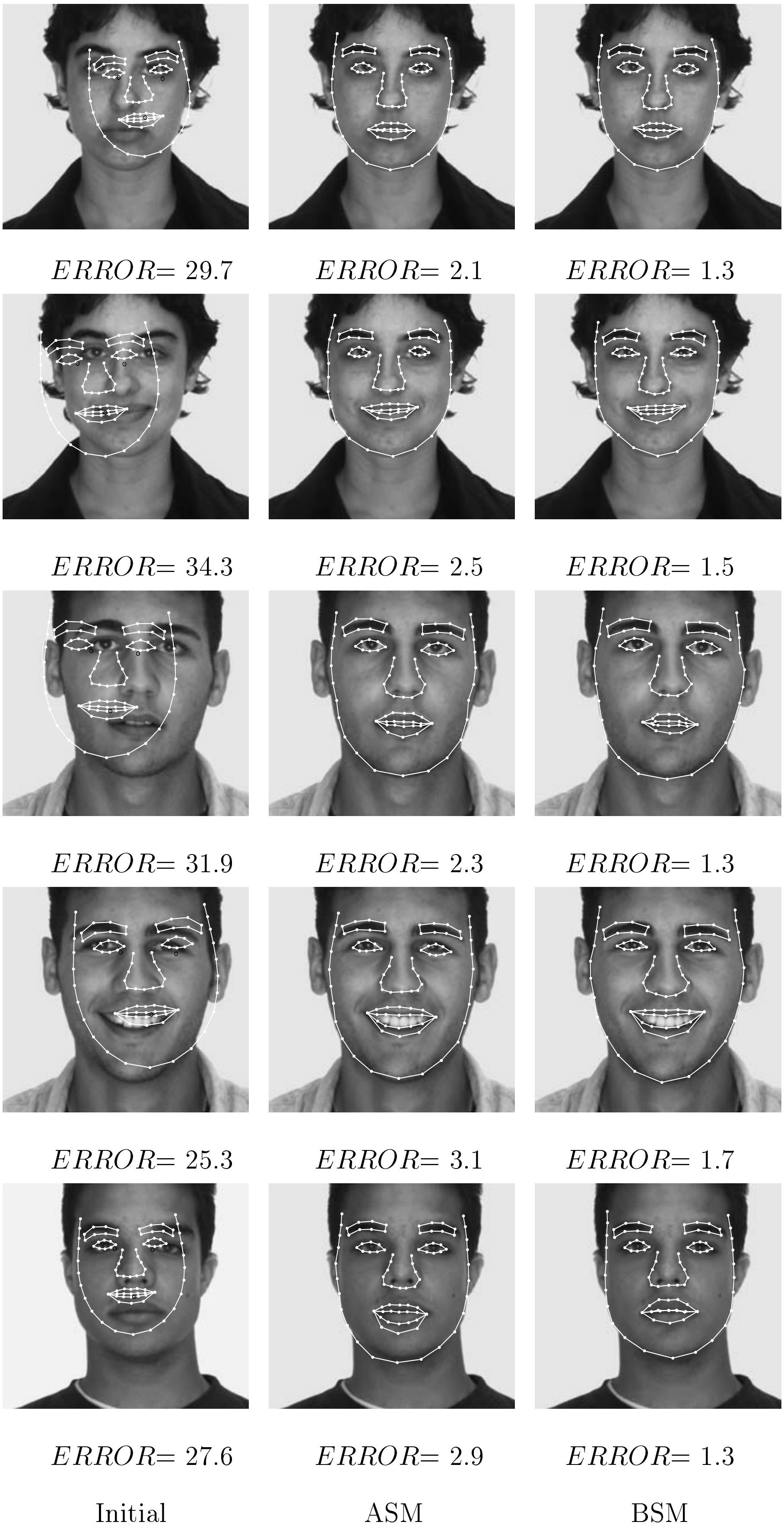
[13] A. Martinez and R. Benavente, "The AR face database," *CVC Technical Report #24, June*, 1998.

[14] Stan Z. Li and J. Lu, "Modeling Bayesian estimation for deformable contours," in *Proceedings of 7th IEEE International Conference on Computer Vision, Kerkyra, Greece.*, 1999, pp. 991–996.

[15] B. Moghaddam and A. Pentland, "Probabilistic visual learninig for object representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696–710, 1997.

[16] H.H.S. Ip and D. Shen, “An affine-invariant active contour model (AI-Snake) for model-based segmentation,” *Image and Vision Computing*, vol. 16, no. 2, pp. 125–146, 1998.

[17] H.A. Rowley, A. Baluja, and T. Kanade, “Rotation invariant neural network-based face detection,” in *International Conference on Computer Vision and Pattern Recognition (CVPR’98)*, 1998, pp. 38–44.



$ERROR = 29.7$ $ERROR = 2.1$ $ERROR = 1.3$

$ERROR = 34.3$ $ERROR = 2.5$ $ERROR = 1.5$

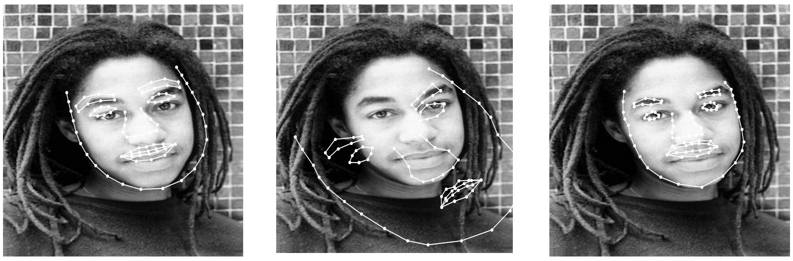
$ERROR = 31.9$ $ERROR = 2.3$ $ERROR = 1.3$

$ERROR = 25.3$ $ERROR = 3.1$ $ERROR = 1.7$

$ERROR = 27.6$ $ERROR = 2.9$ $ERROR = 1.3$

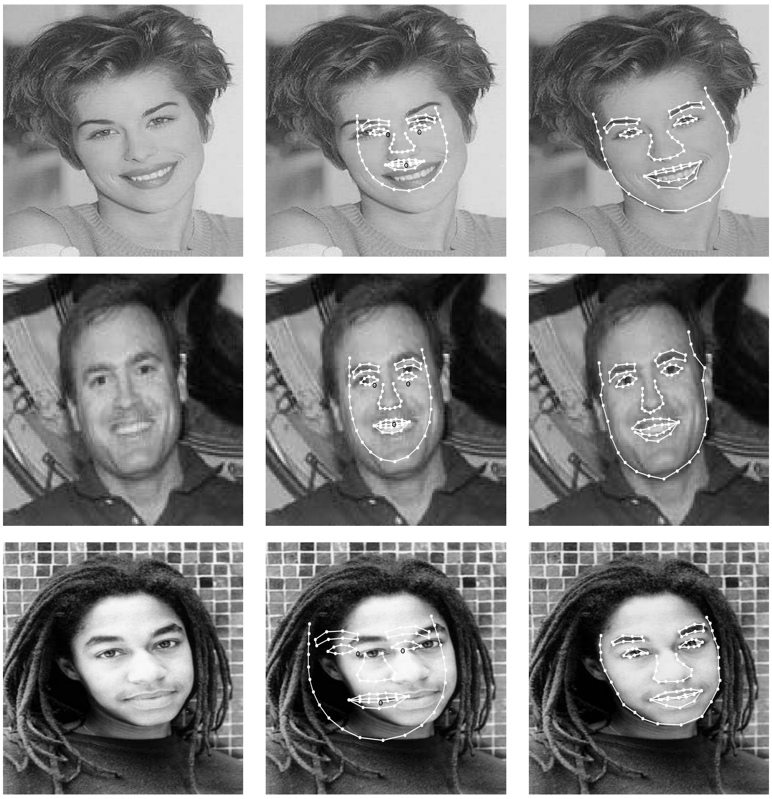
Initial ASM BSM

Fig. 9. Some Matching Results of ASM and BSM.



(a) (b) (c)

Fig. 10. When matching the faces in background different with that of the training samples, the ASM often obtains diverse results. (a) initial position. (b) the diverse result of ASM. (c) the result of ASM when manually fix three points to the image.



input image initial matching result

Fig. 11. Facial feature extraction using BSM.

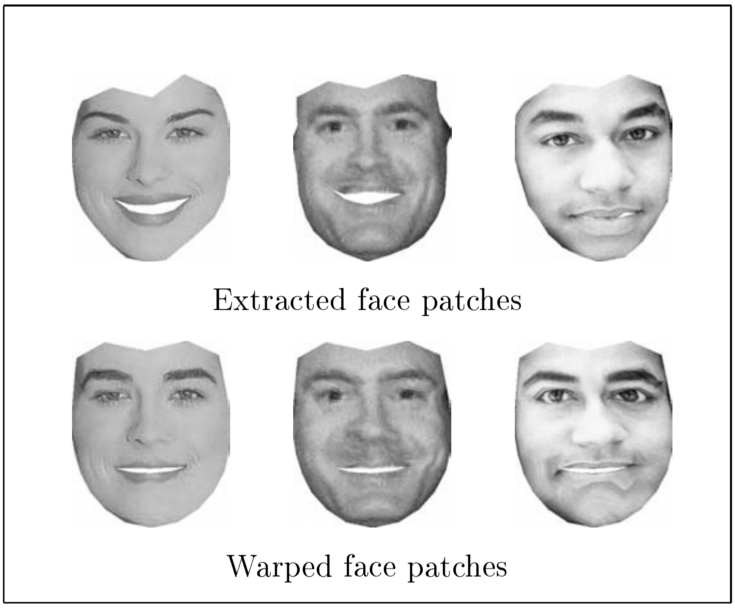


Fig. 12. The extracted face patches and their warped counterparts to the mean facial shapes (all patches are in the shape domain).

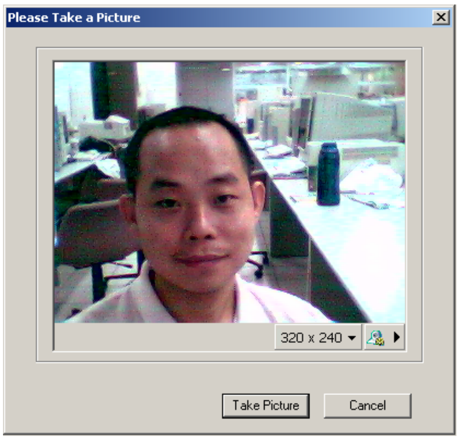


Fig. 13. Capture the face image from USB video camera.

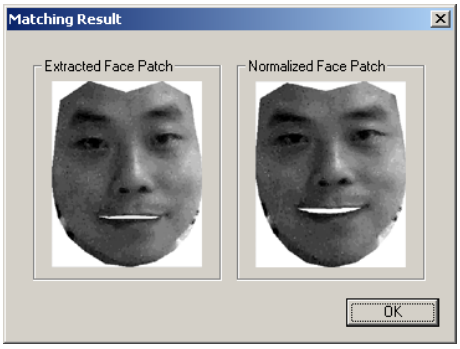


Fig. 14. The extracted and normalized face patches.

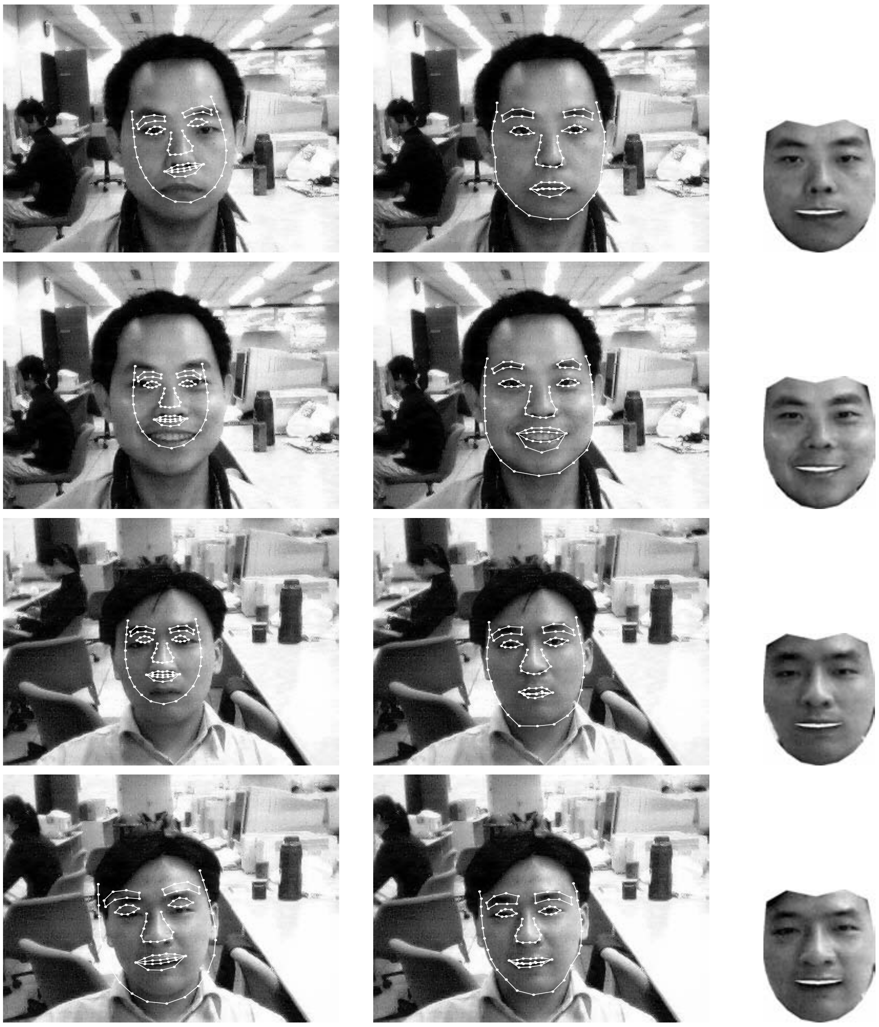


Fig. 15. The extracted and normalized face patches (I).

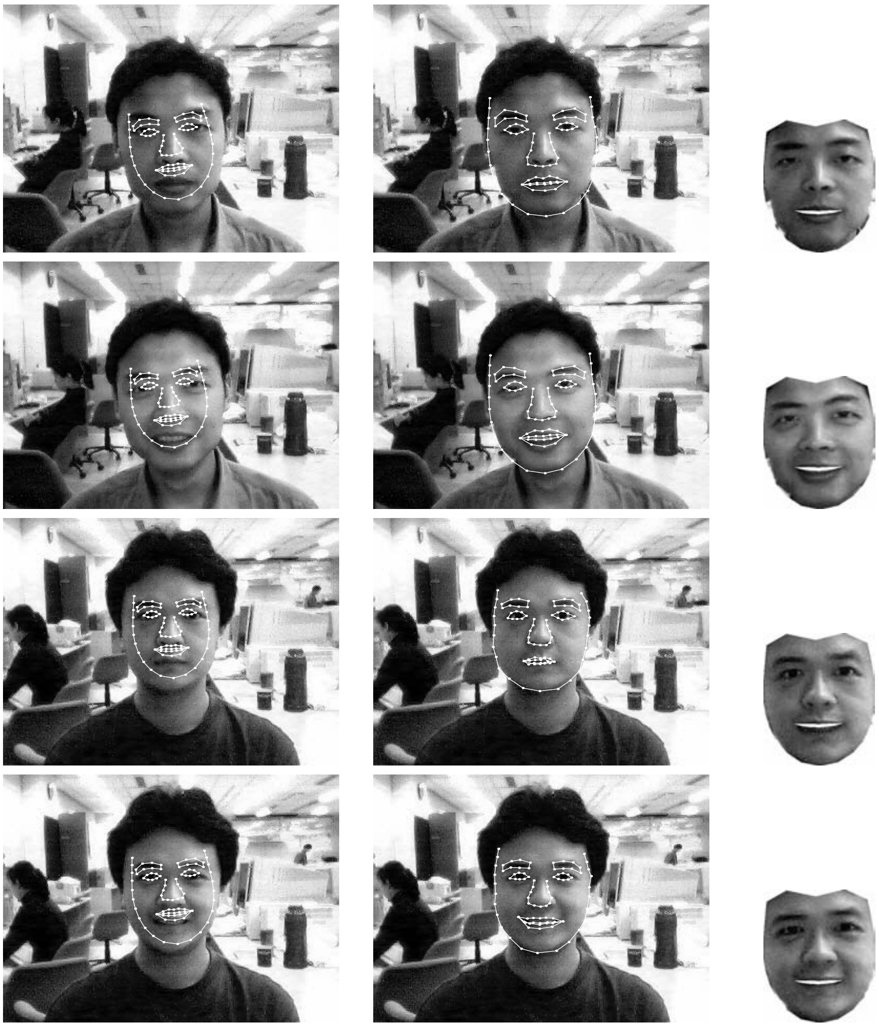


Fig. 16. The extracted and normalized face patches (II).