Pose-normalized 3D Face Modeling for Face Recognition

Sunjin Yu*, Sangyoun Lee* ***Regular Members***

ABSTRACT

Pose variation is a critical problem in face recognition. Three-dimensional(3D) face recognition techniques have been proposed, as 3D data contains depth information that may allow problems of pose variation to be handled more effectively than with 2D face recognition methods. This paper proposes a pose-normalized 3D face modeling method that translates and rotates any pose angle to a frontal pose using a plane fitting method by Singular Value Decomposition(SVD). First, we reconstruct 3D face data with stereo vision method. Second, nose peak point is estimated by depth information and then the angle of pose is estimated by a facial plane fitting algorithm using four facial features. Next, using the estimated pose angle, the 3D face is translated and rotated to a frontal pose. To demonstrate the effectiveness of the proposed method, we designed 2D and 3D face recognition experiments. The experimental results show that the performance of the normalized 3D face recognition method is superior to that of an un-normalized 3D face recognition method for overcoming the problems of pose variation.

Key Words : 3D head pose normalization, 3D face modeling, face recognition

Ⅰ. Introduction

Although current 2D face recognition systems have reached a certain level of maturity, performance has been limited by external conditions such as head pose variation and lighting conditions[1,2]. To alleviate these factors, 3D face recognition methods have recently received significant attention, and the appropriate 3D sensing techniques have been highlighted[3-6]. Many 3D data acquisition methods that use general-purpose cameras and computer vision techniques have been developed[5,6]. Since these methods use general purpose cameras, they may be effective for low-cost 3D input sensors that can be used in 3D face recognition systems. Previous approaches in the field of 3D shape reconstruction in computer vision can be broadly classified into two categories; active and passive sensing.

Although the stereo camera, a kind of passive sensing device, infers 3D information from multiple images, the human face contains an unlimited number of features. Because of this, it is difficult to use dense reconstruction with human faces. There is also a degree of ambiguity when matching point features. This is known as the matching problem[7]. Therefore, passive sensing is not an adequate choice for 3D face data acquisition.

Active stereo sensing, however, uses CCD cameras to project a special pattern onto the subject and reconstruct shapes by using reflected pattern imaging[8]. Because active sensing is better at matching ambiguity and also provides dense feature points, it can act as an appropriate 3D face-sensing technique.

In general, a 3D face modeling device requires pose normalization in order to recognize faces after acquisition. Some face recognition systems using 3D face data assume that a normalized 3D database exists already. They may also carry out another pose

※ This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (KRF-2008-313-D00774)

* 연세대학교 전기전자공학과 영상인식 연구실 (biometrics@yonsei.ac.kr, syleee@yonsei.ac.kr)

논문번호 : KICS2010-11-559, 접수일자 : 2010년 11월 26일, 최종논문접수일자 : 2010년 12월 10일

normalization step after the initial acquisition step[9 - 13].

This represents a 3D head pose normalization problem in 3D face recognition systems. Solutions to this kind of problem can be roughly divided into feature-based and appearance based methods. Feature-based methods try to estimate the position of significant facial features such as the eyes, the nose or the mouth in the facial range or intensity image. Appearance-based methods consider the facial image as a global entity[14]. This paper presents a new method for automatic pose normalized 3D face modeling which is able to perform both 3D face data acquisition and pose normalization simultaneously. Table 1. shows our motivation for this research. In the proposed system, both binary and color time-multiplexing patterns are used in order to solve matching problem. Once the matching problem is solved, the 3D range data is computed by using triangulation. Triangulation is a well-established technique for acquiring range data with matching point information[5]. For 3D head pose normalization, we define a 3D head pose. Especially, we use 2D and 3D information fusion which is based on feature-based model. In 2D space, we use Active Appearance Model(AAM)[15] and in 3D space, we use 3D normal vector from a 3D facial plane which is made from 3D facial features. This paper is organized as follows. Section 2 presents background and section 3 discusses our 3D modeling system and generating patterns. In Section 4 proposed 3D head pose normalization method is explained and Section 5 shows the experimental results. Finally, section 6 concludes the paper.

Ⅱ. Background

2.1 Active Appearance Model

A statistical model of the shape and grey-level appearance of the face is trained using AAM. In these models, the face model is fit to the given image and the extracted model parameters are used in various applications such as tracking, recognition, pose estimation and expression recognition. Each shape sample can be expressed by means of a set of shape coefficients as shown in equation (1) and texture as shown in equation (2).

$$x = \bar{x} + P_s b_s \tag{1}$$

$$g = \bar{g} + P_g b_g, \tag{2}$$

where $\bar{x}$ is the mean shape, P_s is the matrix with eigenvectors on the shapes, b_s is a shape coefficient, $\bar{g}$ is the mean texture, P_g is the matrix with eigenvectors on the texture, and b_s is a texture coefficient. Since the shapes and gray-level textures can be correlated with each other, the shape and texture coefficients can be concatenated into a vector as shown in equation (3).

$$\mathbf{b} = \begin{pmatrix} W_s b_s \\ b_g \end{pmatrix} = \mathbf{Qc} \tag{3}$$

In equation (3), $\mathbf{Q}$ represents the eigenvectors and $\mathbf{c}$ is a vector of the appearance parameters which control both the shape and gray-levels. Thus all images are modeled with a vector $\mathbf{c}$ which describes both the shape and gray-level texture variations together.

With the given face image, it is necessary to search for a 2D face model that minimized the texture error[15]. After AAM matching, b_s, b_g and $\mathbf{c}$ are extracted and used as shape coefficients, $\bar{g}$ represented the mean texture, and P_g is the matrix with eigenvectors on the texture. All of these are used as features to identify faces.

2.2 Camera Calibration

Calibration is the process of estimating the

Table 1. Motivation for Research

	2D Face recognition	3D Face Modeling	Proposed Method
Problem	Pose Variation in 2D Space	Head Pose Normalization for Recognition	3D Modeling + 3D Pose-estimation
Solution	3D Face Modeling	3D Head Pose-estimation	Pose normalized 3D Face Modeling

parameters that determine a projective transformation from the 3D space of the world onto the 2D space of image planes. A set of 3D to 2D point pairs for calibration is obtained with a calibration rig. If we know 6 point pairs, calibration matrix is uniquely determined. However, in many cases, since they have errors, more than 6 point pairs are recommended, and it results in over-determined problem. In this paper, we use 96 point pairs for high accuracy. Then the stereo camera system is calibrated with the DLT(Direct Linear Transform) algorithm[5,6].

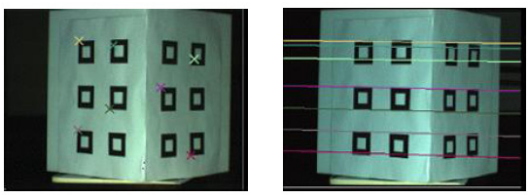
2.3 Epipolar Geometry

We use epipolar geometry in order to solve matching problem. Epipolar geometry refers to the intrinsic projective geometry between two given views. It is independent of scene structure, and depends only on the camera's internal parameters and relative pose[6]. The fundamental matrix $\mathbf{F}$ encapsulates this intrinsic geometry. If a point in 3D space, $\mathbf{X}$, is imaged as $\mathbf{x}$ in the first view, and $\mathbf{x}'$ in the second, then the image points can be said to satisfy the relation equation (4), (5).

$$\mathbf{x}'^{\mathbf{T}} F\mathbf{x} = 0 \tag{4}$$

$$l' = F\mathbf{x} \tag{5}$$

The fundamental matrix is also an algebraic representation of epipolar geometry. It derives the fundamental matrix from the mapping between a point and its epipolar line, and then specifies the properties of the matrix[6]. Using the normalized 8-point algorithm[6,16], we extract the fundamental matrix. Fig. 1. shows calibration patterns and examples of epipolar lines.



(a) points in left view (b) epipolar lines in right view

Fig. 1. Calibration patterns and examples of epipolar lines

2.4 3D Reconstruction

Stereo vision is based on capturing a given scene from two or more points of view and then finding the matching pairs between the different images in order to triangulate the 3D position. However, difficulties in finding the matching may arise, even when taking into account epipolar constraints. Active stereo vision using light can be divided into time-multiplexing, spatial neighborhood and direct coding[8]. It is possible to use both binary and color time-multiplexing coded patterns to solve the matching problem. The idea of these methods is to provide distinguishing features in order to find the matching pairs. To acquire 3D information, we use the triangulation method that is the process of obtaining a real 3D position from two intersecting lines[5]. These lines are defined by the matching pairs and information from each calibration. After camera calibration, the triangulation method is used to obtain the 3D depth information[6].

Ⅲ. 3D Face Modeling System

For the 3D face modeling system, we design coded patterns for solution of matching problem and proposed absolute code interpolation for dense reconstruction. We use 2 cameras and 1 projector.

3.1 3D Face Data Acquisition System

The simplest active technique is to use a single-line stripe pattern, which greatly simplifies the matching process, even though only a single line of 3D data points can be obtained with each image shot. To speed up the acquisition of 3D range data, it is necessary to adopt a multiple-line stripe pattern instead. However, the matching process then becomes much more difficult. One other possibility is to use color information to simplify this difficulty[8].

Furthermore, when using the single-camera approach, it is necessary to find the matching between the color stripes projected by the light source and the color stripes observed in the image. In general, due to the different reflection properties (or surface albedos) of varying object surfaces, the

color of the stripes recorded by the camera is usually different from that of the stripes projected by the light source (even when the objects are perfectly Lambertian)[17]. It is difficult to solve these problems in many practical applications.

However, this does not affect our system (which consists of two cameras and one projector) if the object is Lambertian, because the color observed by the two cameras will be the same, even though this observed color may not be exactly the same as the color projected by the light source. Therefore, by adding one more camera, the more difficult problem of lighting-to-image matching pair is replaced by an easier problem of image-to image stereo matching. Here, the stereo matching problem is also easier to solve than traditional stereo matching problems because an effective color pattern has been projected onto the object.

3.2 Pattern Design

We use both binary and color time-multiplexing coded patterns to solve the matching problem. The idea of these methods is to provide distinguishing features in order to find the matching pairs. The binary sequential pattern uses two illumination levels which are coded as 0 and 1 and the color sequential pattern uses color instead of these two illumination levels[8]. Fig. 2. shows coded binary patterns. Previous research has shown that the color model is an effective model for stereo matching[18-20]. We adopt the HSI model for color-coded sequential pattern generation. Using the line features and the HSI color model, a set of unique color encoded vertical stripes is generated. Each color-coded stripe is obtained as follows. The stripe color is denoted as shown in equation (6).

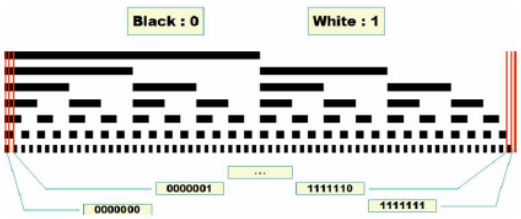


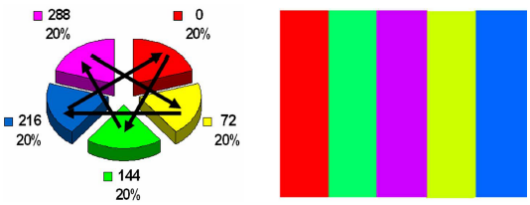
Fig. 2. Coded binary pattern

$$stripe(\rho,\theta)=\rho e^{j\theta}, \tag{6}$$

where ρ is the saturation value and θ is the hue value of the HS polar coordinate system. We use only one saturation value (saturation=1) because there is enough hue information to distinguish each stripe for the matching process. Finally, the stripe color equation is defined by using equation (7).

$$color(m,n)=e^{j(mH_{jmp}+\varepsilon n)}, \tag{7}$$

where m represents the set element, n is the set number, and ε is the hue perturbation. Next, the color-coded sequence is obtained as follows. We determine that the hue distance is $144°$. The next set elements are then use sequentially. In this paper, we generate codes in the face region. We used seven images for the binary pattern and three images for the color pattern. Fig. 3. shows designed color coded pattern. The binary pattern generates $2^7=128$ codes and the color pattern generates $5^3=125$codes.



(a) maximum hue distance (b) designed pattern

Fig 3. Color-coded pattern

3.3 Absolute Code Interpolation

After triangulation, we obtain the 3D reconstructed face data. However, this data is sparse, because we have to deal with a thinned code in each view. Thinning refers to the process of reducing the ambiguity of the matching problem. If we do not use the thinning step, the one-to many matching problem would have been encountered[21]. Fig. 4.

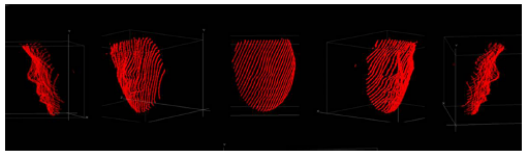


Fig. 4. Reconstructed sparse 3D face

shows the reconstructed sparse 3D face results. However, in order to recognize the images, dense reconstruction is required. Therefore, we use the absolute code interpolation method[22] to discover which human's face surface comprises the continuous region and when the neighbor region's depth is similar. The thinned codes in each view are interpolated by linear interpolation. Fig. 5. shows an example of code interpolation. Fig. 6. shows an example of reconstructed dense 3D face data using the proposed absolute code interpolation. The reconstructed 3D face data consists of a 3D point cloud and a texture map. Fig. 7. shows dense 3D face modeling procedure using absolute code interpolation.

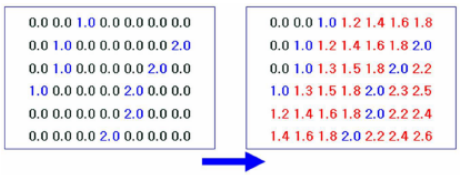


Fig. 5. Example of code Interpolation

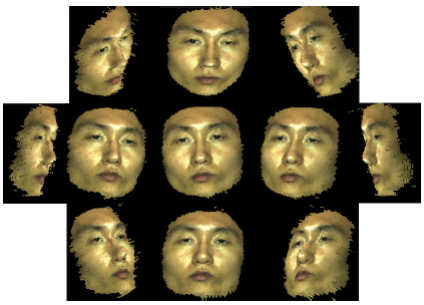


Fig. 6. Reconstructed dense 3D face data with texture using the proposed absolute code interpolation

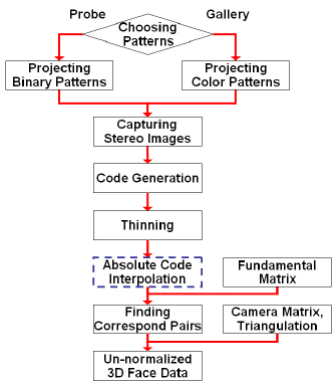
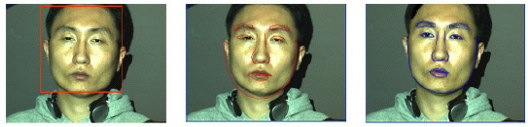


Fig. 7. Dense 3D face modeling procedure using absolute code interpolation



(a) face detection (b) initial shape (c) shape fitting

Fig. 8. Face geometric information using the adaboost algorithm and the AAM

Ⅳ. Proposed Head Pose Normalization Method

For 3D head pose normalization, firstly, we define the 3D head pose and then we fuse 2D and 3D information for head pose normalization. In 2D space we use AAM in order to extract facial geometric features. In 3D space, we calculate the 3D normal vector using relation between 2D and 3D facial features. Finally, we can normalize 3D head pose using 2D and 3D information fusion.

4.1 About 3D Head Pose

Head pose normalization researches are challenging issues in computer vision[9 - 13,22 - 25]. In the problems, there is no obvious choice of the head pose reference frame and its origin. Therefore, we can choose the 3D head pose using a 3D facial normal vector from a 3D facial plane which consists of 4 facial features. At first, we select a left end point of the left eye's region, a right end point of the right eye's region and two mouth points which are both ends of the mouth's region.

If we know geometric information of 4 facial points, we can generate a 3D facial plane. From the 3D facial plane, we can calculate a 3D facial normal vector. This 3D facial normal vector shows 3D head pose. To solve translation problem in 3D space, we translate nose peak point to origin.

4.2 Active Appearance Model Fitting

We apply a AAM model to the selected face region in order to extract the shape and texture features. Fig. 8. shows the way to get the geometric information of a face by using the AAM. AAM

fitting performance is depended on initial shape position. To find the initial shape position, we use the face detection method proposed by the Viola-Jones cascade detector[26]. Due to train, we use 75 pose-varied face images which consist of 5 poses.

4.3 3D Head Pose Normalization

We use a 3D normal vector which represents a head pose[27] and propose a 3D head pose normalization method that uses 2D and 3D information fusion. First, we find the left eye, right eye, nose and mouth's region in the 2D image by using the AAM. Second, we reconstruct the 3D face. Third, because we know the 2D and 3D relation equations, we are able to select 4 facial features in 3D space which are a left end point($\mathbf{P_1}$) of the left eye's region, a right end point($\mathbf{P_2}$) of the right eye's region, two mouth points($\mathbf{P_3}$ and $\mathbf{P_4}$) which are both ends of the mouth's region and the maximum depth point($\mathbf{N_{pp}}$) of the nose's region. In the presence of noise such as spike, hole and so on, we consider local standard deviation and threshold. The selected facial features are defined as shown in equation(8), (9), (10), (11) and (12).

$$\mathbf{N_{pp}} = \arg\max_{\mathbf{N}_n}(z_n), \tag{8}$$

where each $\mathbf{N}_n = [x_n, y_n, z_n]^T, n = 1, \cdots, v_n$ is a vertex of the nose region and v_n is the number of vertex points within the nose region.

$$\mathbf{P_1} = \arg\min_{\mathbf{L}_l}(x_l), \tag{9}$$

where each $\mathbf{L}_l = [x_l, y_l, z_l]^T, l = 1, \cdots, v_l$ is a vertex of the left eye region and v_l is the number of vertex points in the left eye region.

$$\mathbf{P_2} = \arg\max_{\mathbf{R}_r}(x_r), \tag{10}$$

where each $\mathbf{R}_r = [x_r, y_r, z_r]^T, r = 1, \cdots, v_r$ is a vertex of the right eye region and v_r is the number of vertex points in the right eye region.

$$\mathbf{P_3} = \arg\min_{\mathbf{M}_m}(x_m), \tag{11}$$

$$\mathbf{P_4} = \arg\max_{\mathbf{M}_m}(x_m), \tag{12}$$

where each $\mathbf{M}_m = [x_m, y_m, z_m]^T, m = 1, \cdots, v_m,$ is a vertex of the mouth's region and v_m is the number of vertex points in the mouth's region.

Fourth, using the selected eyes and mouth points in 3D space, we produce a 3D facial plane by a simple plane fitting algorithm which uses SVD as shown in equations (13) and (14).

$$\mathbf{P} = \begin{bmatrix} \mathbf{P_1^T} \\ \mathbf{P_2^T} \\ \mathbf{P_3^T} \\ \mathbf{P_4^T} \end{bmatrix}^{\mathbf{T}} = \begin{bmatrix} \mathbf{p_1} \\ \mathbf{p_2} \\ \mathbf{p_3} \end{bmatrix}$$

$$\mathbf{P_{mean}} = \begin{bmatrix} mean(\mathbf{p_1}) \\ mean(\mathbf{p_2}) \\ mean(\mathbf{p_3}) \end{bmatrix} = \begin{bmatrix} p_{m1} \\ p_{m2} \\ p_{m3} \end{bmatrix}, \tag{13}$$

$$\mathbf{P_{nor}} = \begin{bmatrix} \mathbf{p_1} - p_{m1} \\ \mathbf{p_2} - p_{m2} \\ \mathbf{p_3} - p_{m3} \end{bmatrix}$$

$$[\mathbf{U}\,\mathbf{D}\,\mathbf{V}] = SVD(\mathbf{P_{nor}})$$

where $\mathbf{P}$ is a matrix which consists of extracting facial features and $\mathbf{p_1}$, $\mathbf{p_2}$, and $\mathbf{p_3}$ are row vector of $\mathbf{P}$.

$$\mathbf{V} = [\mathbf{v_1}\,\mathbf{v_2}\,\mathbf{v_3}]$$

$$\mathbf{E} = \frac{-1}{\mathbf{v_3}(3)} \times \mathbf{v_3} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}, \tag{14}$$

$$A = e_1,\ B = e_2,\ C = -\mathbf{P_{mean}} \times \mathbf{E}$$

where $z = Ax + By + C$ is a plane equation and $\mathbf{v_1}$, $\mathbf{v_2}$, and $\mathbf{v_3}$ are column vector of $\mathbf{V}$ and $\mathbf{v_3}(3)$ is a last element of $\mathbf{v_3}$.

Fifth, using this plane, we are able to find a 3D facial normal vector($\mathbf{N_f} = [n_x, n_y, n_z]^{\mathrm{T}}$) by calculating a plane normal vector. Finally, the problem of DOF 3 against translation is solved by nose peak point rearrangement and the problem of DOF 3 against

$$\mathbf{R}_{\mathbf{N_f},\mathbf{S}}(\theta)=\begin{bmatrix} cos(\theta)+(1-cos(\theta))n_x{}^2 & (1-cos(\theta))n_xn_y-sin(\theta)n_z & (1-cos(\theta))n_xn_z+sin(\theta)n_y \\ (1-cos(\theta))n_xn_y+sin(\theta)n_z & cos(\theta)+(1-sin(\theta))n_y{}^2 & (1-cos(\theta))n_yn_z-sin(\theta)n_x \\ (1-cos(\theta))n_xn_z-sin(\theta)n_y & (1-cos(\theta))n_yn_z+sin(\theta)n_x & cos(\theta)+(1-cos(\theta))n_z{}^2 \end{bmatrix} \quad (15)$$

rotation is solved by the 3D facial normal vector rearrangement. It acquires and normalized the 3D face data at once. After extracting the 3D facial normal vector($\mathbf{N_f}$) from un-normalized 3D face, we can calculate the rotation matrix $\mathbf{R}_{\mathbf{N_f},\mathbf{S}}(\theta)$ from the 3D facial normal vector $\mathbf{N_f}$ to standard vector $\mathbf{S}=[0,0,1]^T$ via θ as shown in equation (15), where $\theta = arccos\left(\frac{\mathbf{N_f}\cdot\mathbf{S}}{\|\mathbf{N_f}\|\cdot\|\mathbf{S}\|}\right)$. Fig. 9. shows concept of defined 3D facial features, facial plane and vectors. The process of the proposed 3D head pose normalization method is as follows:

1. Find the 2D geometry features such as eyes, nose and mouth's region in the 2D image using AAM.
2. Reconstruct the 3D face data.
3. Find the 3D geometry features in the 3D space using the relation between the 2D geometry features and the 3D position.
4. Search the nose peak point($\mathbf{N_{pp}}$) under the nose region using equation (8).
5. Move the 3D face data based on the nose peak point($\mathbf{N_{pp}}$).
6. Calculate the 3D facial normal vector using the selected two eyes and the mouth points($\mathbf{P_1}$, $\mathbf{P_2}$, $\mathbf{P_3}$, and $\mathbf{P_4}$) based on equations (9), (10), (11) and (12).
7. Calculate the 3D facial plane by plane fitting algorithm as shown in equations (13) and (14).
8. Calculate the 3D facial normal vector($\mathbf{N_f}$) by the 3D facial plane normal.
9. Rotate the 3D facial normal vector ($\mathbf{N_f}$) with the 3D face data for the X and Y axis using equation (15).
10. Check the eyes' position and align the 3D face data for the Z axis horizontally.

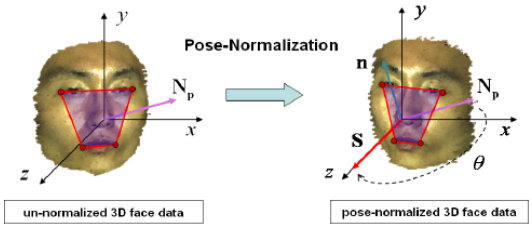


Fig. 9. Defined 3D facial features, 3D facial plane and vectors

4.4 Pose-normalized 3D Face Modeling Procedure

3D face modeling can be divided into an off-line and an on-line process. Calculating the camera matrix and the fundamental matrix is an off-line process. Because the camera matrix and the fundamental matrix are calculated before the on-line process, they are not related to running-time. After the off-line process, it is possible to carry out the on-line process, which is related to running time. The on-line process can be roughly divided into 2D and 3D processing. 2D processing consists of capturing the image pairs, detecting the face region, extracting the 2D face geometry features, thinning, absolute code interpolation and finding matching pairs. 3D processing consists of triangulation as well as the automatic pose normalized face modeling step. Fig. 10. shows automatic pose-normalized 3D face modeling full procedure.

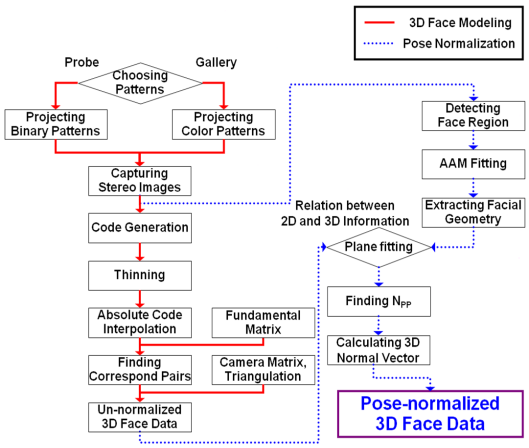


Fig. 10. Automatic pose-normalized 3D face modeling full procedure

Ⅴ. Experimental Results

To evaluate accuracy of proposed 3D face modeling system, we measure length and angle of a skin-colored box. And then, we make two recognition experiments in order to evaluate performance of the proposed pose normalized method.

5.1 Experiment Environments

The proposed algorithm is tested with one projector, two cameras, and a PC. A sharp PG-M25X DLP projector (1024 by 768 resolution) is used for illuminating the binary and color-coded pattern to the object, and two JAI CV-M77 cameras with color CCDs are used as the color cameras. These cameras are able to produce a progressive scan at 25 frames per second, and the size of each frame is 1024 by 768. A simulation code is written in Visual C++ 6.0 and run under a Pentium 4-2.4 [GHz] environment. Fig. 11. shows the overall 3D reconstruction environments.

We test accuracy to evaluate the performance of our system. To test accuracy, we use a skin-colored box. Fig. 12. shows the 3D reconstruction results using our system. We estimate the width, height and degree of the box. Table 2. shows the obtained results. From Table 2., we can see our system's accuracy when compared to the real values.

Fig. 11. The composition of the stereo cameras and the projector

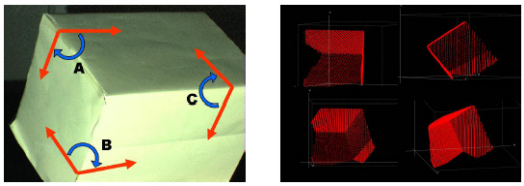


Fig. 12. 3D reconstruction results from different viewing points

Table 2. Accuracy test results

	Width	Length	Height	Degree A	Degree B	Degree C
Real value	15	13	10	89	90	91
Reconstruction result	14.91	12.82	10.18	89.22	89.54	90.32

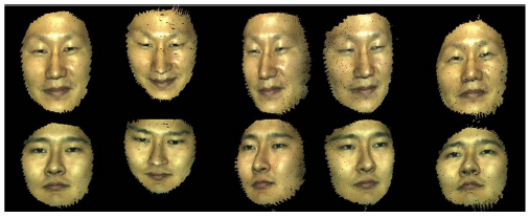
5.2 Face Database

We create a 3D face database of fifteen persons. Each person is captured at five different pose variations: front, up, down, left and right. Also, each person is subjected to binary sequential and color sequential reconstruction. The color sequential reconstructed data is used for a AAM train and a gallery, while the binary sequential reconstructed data is used for a query.

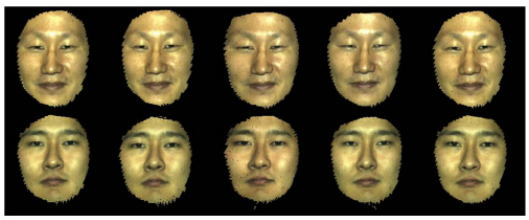
5.3 Recognition Experiments

Due to evaluate performance of the proposed pose normalized method, we compare three types of recognition experiments. The first experiment is done in order to recognize the reconstructed 3D face data without pose normalization using projected 2D face data. The second experiment is done to recognize the reconstructed 3D face data with pose-normalization using projected 2D face data. The recognition method is Principal Component Analysis(PCA)[28] with Euclidean distance. The third experiment is done in order to recognize the reconstructed 3D face data with pose-normalization using 3D curvatures. The last experiment is done in order to increase recognition performance. In this case, we use score level fusion method. After proposed processing, we obtain the normalized 3D face data. Fig. 13. shows comparing results between before and after. In the first and second experiments, 3D face data projects the 3D to 2D space with texture. This method can apply the existing 2D face recognition algorithm easily. In the third experiment, we use Dynamic Programming(DP) using 3 facial curvatures. Because of proposed modeling method, we can use eyes and nose peak information in 3D space easily. Used facial curvatures consist of them.

First facial curvature line is facial profile which is pass through nose peak vertically and second facial



(a) Before applying the proposed method



(b) After applying the proposed method

Fig. 13. Comparing results between before and after

curvature line is linked between two eyes horizontally. Third facial curvature line is horizontal line including nose peak. In the last experiment, we consider score level fusion method as shown in equation (16).

$$Score = w \cdot S_{PCA} + (1 - w) \cdot S_{DP} \qquad (16)$$

where w is the weight, S_{PCA} is the score using PCA and S_{DP} is the score using DP. Table 3. shows the used sets. To recognize them at the first and second experiments, we use the AAM shape information and Euclidean distance. In the third experiment, we use DP distance and in the last experiment, we use both of Euclidean and DP distances. Table 4. shows the identification results.

Table 3. Used Sets And Types.

Used Set	Type	Number of data
Color Sequential	Gallery	$5 \times 15 = 75$
Binary Sequential	Query	$5 \times 15 = 75$

Table 4. Recognition results.

Set	Normalization	Methods	Recognition rate
Projected 2D data with texture	Off	PCA	60%(45/75)
Projected 2D data with texture	On	PCA	94.67%(71/75)
3D Curvatures	On	DP	84%(63/75)
2D Texture + 3D Curvatures	On	Score Level Fusion	100%(75/75)

Ⅵ. Conclusions

This paper presents the pose-normalized 3D face modeling for face recognition. We focus on 3D face modeling and 3D head pose normalization at once and assume that the user's pose variation range is $\pm 35^{\circ}$ in each axis in this paper but this is not critical limitations in practical. Because stereo vision based 3D face modeling system can generate only captured region of the face in front of cameras, users who want to identify have to gaze at the cameras practically.

Identification system is trained for only resisted people using AAM. In this paper, we use AAM in order to extract facial features, but it is not requirement. If we have other strong facial feature extraction methods, we can exchange facial feature extraction methods.

For 3D face modeling, we use two cameras and one projector, with expansion capabilities. Because the projector provides the features, we are easily able to change features such as lasers, infra-red systems and so on. To solve matching problem, we use epipolar constraint and design coded patterns. In order to make dense reconstruction, we use the absolute code interpolation method.

To normalize the pose factor in 3D space, we use the 2D and 3D information. Both 2D and 3D information can present both advantages and disadvantages. Therefore, we use the 2D and 3D information to provide advantages and to offset disadvantages. We calculate a 3D facial normal vector for rotation problem, and searched nose peak point for translation problem.

We produce a full face recognition system from 3D data acquisition to recognition so that we produce 3D modeling which uses a pose-normalized method. The proposed 3D head pose-normalization method shows robust results against pose variation. Experimental results show that the normalized recognition rate has improved about 35% when compared with un-normalized recognition in Euclidean distance. In 3D recognition, we use DP method. Because, we use only 3 curvatures, experimental result shows 2D information is better

than 3D information. In future work, we will adapt our pose normalization method to general 3D face database which is acquired different 3D face modeling system, firstly. Secondly, we will adapt various 3D face recognition methods to our system such as 3D PCA, Iterative Closest Point (ICP) and etc. Finally, we hope to adapt proposed head pose-normalization method to different sensors.

References

〔1〕 R. Chellappa, C.L.Wilson and S. Sirohey, "Human and Machine Recognition of Faces: A Survey," *Proc. IEEE*, Vol.83, No.5, pp.705-740, 1995.

〔2〕 W. Zhao, R. Chellappa, P.J. Phillips and A. Rosenfeld, "Face Recognition: A Literature Survey," *ACM Computing Surveys*, Vol.35, No.4, pp.399-458, 2003.

〔3〕 H.S. Yang, K.L. Boyer and A.C. Kak, "Range data extraction and interpretation by structured light," *Proc. IEEE Conference on Artificial Intelligence Applications*, pp.199-205, 1984.

〔4〕 K.L. Boyer and AC. Kak, "Color-encoded structured light for rapid active ranging," *IEEE Trans Pattern Analysis and Machine Intelligence*, pp.14-28, 1987.

〔5〕 E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vi-sion*, Prentice Hall, 1998.

〔6〕 R. Hartley and A. Zisserman, *Multiple view Geometry in computer vision*, Cambridge University Press, 2000.

〔7〕 D. Scharstein and R. Szeliski, "Stereo Matching with Nonlinear Diffusion," *International Journal of Computer Vision,* Vol.28, No.2, pp.155-174, 1998.

〔8〕 J. Salvi, J. Pags and J. Batlle, "Pattern Codification Strategies in Structured Light Systems," *Pattern Recognition,* Vol.37, No.4, pp.827-849, Apr. 2004.

〔9〕 S. Malassiotis and M.G. Strintzis, "Pose and illumination compensation for 3D face recognition," *Proc. ICIP*, pp.91-94, 2004.

〔10〕 S. Malassiotis and M.G. Strintzis, "Robust Face Recognition Using 2D and 3D Data: Pose and Illumination Compensation," *Pattern Recognition*, Vol.38, No.12, pp.2536-2548, Dec. 2005.

〔11〕 K.I. Chang, K.W. Bowyer and P.J. Flynn, "Multi-Modal 2D and 3D Biometrics for Face Recognition," *Proc. IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, pp.187-194, 2003.

〔12〕 C. Xu, Y. Wang, T. Tan and L. Quan, "A New Attempt to Face Recognition Using Eigenfaces," *Proc. ACCV*, pp.884-889, 2004.

〔13〕 H. Song, S. Lee, J. Kim and K. Sohn, "3D sensor based face recognition," *Applied Optics*, Vol.44, No.5, pp.677-687, Feb. 2005.

〔14〕 H. Song, "3D Head Pose Estimation and Face Recognition Using Range Image," *Ph.D Thesis,* Yonsei University, South Korea, 2005.

〔15〕 T. F. Cootes, G. Edwards and C. Taylor, "Active Appearance Models," *IEEE Trans. On Pattern Analysis and Machine Intelligence*, Vol.23, No.6, pp.681-685, 2001.

〔16〕 R. Hartley, "In Defence of the 8-points Algorithm," *Proc. Fifth International Conference on Computer Vision*, Jun. 1995.

〔17〕 D. Shin and J. Kim, "Point to Point Calibration Method of Structured Light for Facial Data Reconstruction," *Proc. ICBA 2004,* pp.200-206, Jul. 2004.

〔18〕 C.H. Hsieh, C.J. Tsai, Y.P. Hung and SC. Hsu, "Use of chromatic information in region-based stereo," *Proc. IPPR Conference on Computer Vision, Graphics, and Image Processing*, pp.236-243, 1993.

〔19〕 R.C. Gonzales and R.E. Woods, *Digital Image Processing*, Addison-Wesley, 1992.

〔20〕 C. Chen, Y. Hung, C. Chiang and J. Wu, "Range data acquisition using color structured lighting and stereo vision," *Image and Vision Computing,* pp.445-456, 1997.

〔21〕 B. Kim, S. Yu, S. Lee, J. Kim, "Rapid 3D Face Data Acquisition Using a Color-Coded Pattern and a Stereo Camera System," *Proc. ICB* 2006, pp.26-32, Jan. 2006.

〔22〕 S. Yu, K. Choi and S. Lee, "Automatic

Pose-Normalized 3D Face Modeling and Recognition Systems," *Proc. IEEE Pacific-Rim Symposium on Image and Video Technology*, pp.652-661, Dec. 2006.

〔23〕 H. Kim, D. Kim and S. Bang, "A PCA mixture model with an efficient model selection method," *Proc. International Joint Conference on Neural Networks*, 2001 Vol.1, pp.430-435, July 2001.

〔24〕 S. Kim, H. Lee, S. Yu and S. Lee, "Robust Face Recognition by Fusion of Visual and Infrared 10 Cues," *Proc. IEEE Conference on Industrial Electronics and Applications*, pp.804-808, 2006.

〔25〕 X. Chai, S. Shan and W. Gao, "Pose normalization for robust face recognition based on statistical affine transformation," *Proc. Joint Coference on International Conference on Information, Communications and Signal Processing and Pacific Rim Conference on Multimedia,* Vol.3, pp.1413-1417, Dec. 2003.

〔26〕 P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, Vol.57, No.2, pp.137-154, 2004.

〔27〕 S. Yu, J. Kim and S. Lee, "Iterative three dimensional head pose estimation using a face normal vector," *Opt. Eng.,* Vol.48, 037204 (2009).

〔28〕 M. Turk And A. Pentland, Eigenfaces for recognition," *J. Cogn. Neurosci.,* Vol.3, pp.72-86, 1991.

유 선 진 (Sunjin Yu) 정회원

2003년 8월 고려대학교 전자 및 정보공학과 학사

2006년 2월 연세대학교 생체인식 협동과정 석사

2006년 3월~현재 연세대학교 전기전자공학과 박사과정

<관심분야> 3D vision, HCI, 얼굴 인식

이 상 윤 (Sangyoun Lee) 정회원

1987년 2월 연세대학교 전자공학과 학사

1989년 2월 연세대학교 전자공학과 석사

1999년 2월 Georgia Tech. 전기 및 컴퓨터공학과 박사

1989년~2004년 KT 선임연구원

2004년~현재 연세대학교 전기전자공학부 부교수

<관심분야> 생체인식, 컴퓨터비전, 영상부호화