3D Model Based Face Recognition by Face Representation Using PVM and Pose Approximation

Yang-Bok Lee[1], Taehwa Hong[2], Hyeon-Joon Moon[1], and Yong-Guk Kim[1]

[1] School of Computer Engineering, Sejong University, Seoul, Korea
yangbok@empal.com{hmoon,ykim}@sejong.ac.kr
[2] Samsung Electronics, Suwon, Korea

Abstract. Since a generative 3D face model consists of a large number of vertex points and polygons, a 3D model based face recognition system is generally inefficient in computation time. In this paper, we present a novel 3D face representation method to reduce the number of vertices and optimize its computation time and generate the 3D Korean face model based on the representation method. Also, a pose approximation method is described for initial fitting parameter. Finally, we evaluate the performance of proposed method with the face databases collected using a stereo-camera based 3D face capturing device and a web camera.

1 Introduction

Recently, a large number of researches on 3D based face recognition have been presented with various algorithms. It has several advantages over traditional 2D face recognition: First, 3D data provides absolute geometrical shape and size information of a face. Additionally, face recognition using 3D data is more robust to pose and posture changes since the model can be rotated to any arbitrary position. Also, 3D face recognition can be less sensitive to illumination since it does not solely depend on pixel intensity for calculating facial similarity. Finally, it provides automatic face segmentation information since the background is typically not synthesized in the reconstruction process [1].

V. Blanz, T. Vetter and S. Romdhani [2][3][4] proposed a method using 3D morphable model for face recognition robust to pose and illumination. They constructed a 3D morphable model with 3D faces acquired from a 3D laser-scanner, which are positioned in well calibrated cylindrical coordinates. Especially, texture information of 3D face is well defined in the reference frame where one pixel corresponds to one 3D vertex perfectly. Thereafter, they found appropriate shape and texture coefficient vectors of the model by fitting it to an input face using Stochastic Newton Optimization (SNO)[3] or Inverse Compositional Image Alignment (ICIA)[4][5] as a fitting. However, this approach has complexity and inefficiency problems caused by very large vertex number (about 80,000 ~ 110,000) despite of excellent performance.

C. Stephanidis (Ed.): Universal Access in HCI, Part II, HCII 2007, LNCS 4555, pp. 943–951, 2007.

In this paper we propose a novel 3D face representation method based on pixel-to-vertex map (PVM) and pose approximation method. Especially, through the representation method, it is possible to reduce the vertex number of each 3D face remarkably and all the 3D faces can be aligned with correspondence information based on the vertex number of a reference face simultaneously.

This paper is organized as follows. The PVM for 3D face representation and model generation are described in next section. Section 3 and Section 4 introduce the pose approximation process and the procedure of fitting the 3D model to an input image, respectively. Experimental results are presented in Section 5 based on a Korean database. Finally, conclusions and future work are discussed in Section 6.

2 3D Face Representation and Modeling

One of the most required parts in model-based face recognition system is clearly to generate a reliable face model. The reliability of the generated model has an important effect on the performance of the whole system. Then, model generation mainly depends on how well the face data are represented.

In most cases, a 3D face scan consists of texture intensity in 2D frame and geometrical shape in 3D. Especially, the shape information is represented by close connections of many vertices and polygons. Since the number of vertex points composing each of 3D face scans is different from each other, it is necessary to manipulate them to have the same number of vertices for consistent mathematical expression.

2.1 Shape Correspondence Using the Pixel-to-Vertex Map (PVM)

To achieve shape correspondence using the PVM, in the first place, we have manually registered three key features, which are left eye, right eye and mouth (from the 2D texture information of each 3D face scan). Thereafter, a fixed point for each of the features in texture coordinate is set and the features of each 3D face data are located in the fixed positions by rotating, translating and scaling. Then, the face region of a face data is separated from its background by adopting an elliptical mask.

PVM is a sort of binary image, which classifies pixels in the masked face region into ones mapped to a vertex and the opposite. We call the former active pixel (AP) and the latter inactive pixel (IP).

The procedures for the vertex correspondence using PVM are as follows:

- Construct each PVM matrix of M+1 3D face scans and build the vertex position matrix by stacking the position vector of the vertex mapping to each AP in a PVM. If the resolution of the texture frame is C by R, the PVM matrix of the ith scan, denoted by $\mathbf{M}_i$ and the vertex position matrix of the ith scan, denoted by $\mathbf{P}_i$ are obtained as

$$\mathbf{M}_i = \begin{bmatrix} m_{11}^i & m_{12}^i & \cdots & m_{1C}^i \\ m_{21}^i & & & \\ \vdots & & \ddots & \vdots \\ m_{R1}^i & & \cdots & m_{RC}^i \end{bmatrix}, \qquad m_{rc} = \begin{cases} 0, \text{ if } \mathrm{p}_{\mathrm{rc}} \text{ is IP} \\ 1, \text{ if } \mathrm{p}_{\mathrm{rc}} \text{ is AP} \end{cases}. \tag{1}$$

$$\mathbf{P}_i = [\mathbf{v}_1^i \quad \mathbf{v}_2^i \quad \cdots \quad \mathbf{v}_{s(\mathbf{M}_i)}^i]. \tag{2}$$

where p_{rc} is the pixel positioned at (r, c) in the texture frame and $s(\mathbf{M}_j)$ is size of the PVM, the number of the APs in the PVM. Also, $\mathbf{v}_j^i = [x_j \ y_j \ z_j]^T$ is the 3D position vector of the vertex mapping to the j^{th} AP in the i^{th} scan.

- Select a reference pixel-to-vertex map(RPVM), denoted by $\mathbf{M}^R$, by maximizing this criterion.

$$\mathbf{M}^R = \arg \max_{\mathbf{M}_j} s(\mathbf{M}_j). \tag{3}$$

The size of the RPVM, $s(\mathbf{M}^R)$ means the vertex number of a reduced subset. Then, all scans will be in correspondence with the vertex number. Likewise, the vertex position matrix of RPVM are denoted by $\mathbf{P}^R$.

- Compute each modified vertex position matrix of all scans except one selected for the RPVM.

$$\hat{\mathbf{v}}_k^i = \begin{cases} \mathbf{v}_{p(k)}^i, \text{ if } \mathrm{m}_{\mathrm{p(k)}}^{\mathrm{i}} \text{ is AP} \\ \mathbf{v}^N, \ \text{ if } \mathrm{m}_{\mathrm{p(k)}}^{\mathrm{i}} \text{ is IP} \end{cases}. \tag{4}$$

$$\mathbf{v}^N = \sum_{q=1}^{8} w_q \mathbf{v}_q. \tag{5}$$

$$w_j = \frac{\frac{1}{d_j}}{\sum_{j=1}^{8} \frac{1}{d_j}} \tag{6}$$

where $\hat{\mathbf{v}}_k^i$ is a modified vertex position vector, which is the same to the position of the original vertex if mapped to AP, otherwise should be acquired by an interpolation method. And, the subscripted $p(k)$ means the position of the pixel mapped to the vertex related to kth column in the $\mathbf{P}^R$. We have to seek an appropriate 3D position $\mathbf{v}^N$ for a vertex mapped to IP using linear combinations of the positions of vertices mapped to 8 nearest neighbor APs in the PVM of the target scan as defined in eq. (5) and (6).

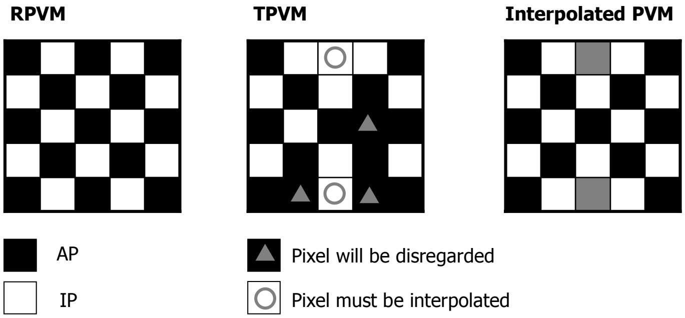

Fig. 1. Vertex correspondence by PVM

2.2 Face Model Generation

Through the PVM, it is possible that all 3D face data in our database are expressed with the same number of vertex points. Assume that there are N 3D face data in the database and they are expressed with V vertex points, shape and texture information can be written as eq.(7).

$$\mathbf{S}=\begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^N \\ y_1^1 & y_1^2 & \cdots & y_1^N \\ z_1^1 & z_1^2 & \cdots & z_1^N \\ \vdots & \vdots & \vdots & \vdots \\ & & & \\ x_V^1 & x_V^2 & \cdots & x_V^N \\ y_V^1 & y_V^2 & \cdots & y_V^N \\ z_V^1 & z_V^2 & \cdots & z_V^N \end{bmatrix},\quad \mathbf{T}=\begin{bmatrix} t_1^1 & t_1^2 & \cdots & t_1^N \\ t_2^1 & t_2^2 & \cdots & t_2^N \\ \vdots & \vdots & \ddots & \vdots \\ t_V^1 & t_V^2 & \cdots & t_V^N \end{bmatrix} \tag{7}$$

We constructed separate models from shapes and textures of 100 Korean people by applying PCA [9][10] independently. The separate models are generated by linear combination of the shapes and textures as given by eq. (8).

$$\mathbf{S}=\mathbf{S}_0+\sum_{j=1}^{N_S}\alpha_j\mathbf{S}_j, \qquad \mathbf{T}=\mathbf{T}_0+\sum_{j=1}^{N_T}\beta_j\mathbf{T}_j. \tag{8}$$

where $\boldsymbol{\alpha}=[\alpha_1\,\alpha_2\cdots\alpha_{N_S}]$ and $\boldsymbol{\beta}=[\beta_1\,\beta_2\cdots\beta_{N_T}]$ are the shape and texture coefficient vectors (should be estimated by a fitting procedure). Also, $\mathbf{S}_0$ and $\mathbf{S}_j$ are the shape average model and the eigenvector associated with the jth largest eigenvalue of the shape covariance matrix, $\mathbf{T}_0$ and $\mathbf{T}_j$ in textures likewise.

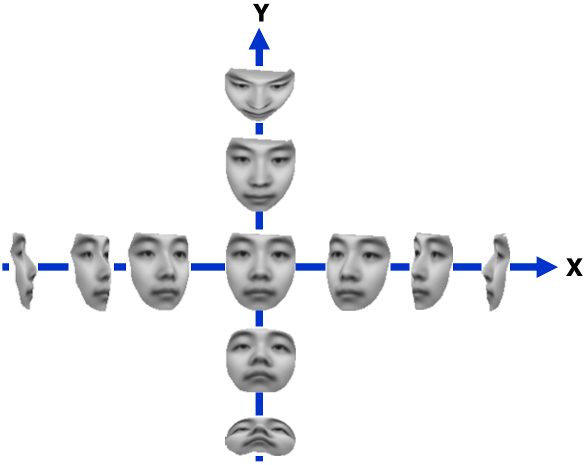

Fig. 2. A generative 3D average face model

3 Head Pose Analysis

Pose estimation is inevitably essential for implementing the face recognition system in pose varying environment. However, it is well known that it is not easy to determine accurately the pose of a face in 2D space, due to the absence of depth information to the camera axis (usually z-axis).

Our system also needs a pose estimation method for the robustness to pose variation certainly. Needless to say, the goal of our pose estimation is to compute not the accurate pose of the given face by an explicit mathematical expression but only the approximately estimated pose.

The proposed method is based on a topology of facial key features extracted from a detection algorithm or a manual registration, and a fixed marginal pose established by statistical analysis of the 3D data. Some assumptions must be preceded for the pose approximation as follows:

Assumption1. There is no variation to the z axis.

Assumption2. The pose variation to the y axis is considered in a selected marginal pose.

3.1 Marginal Pose

From the registered feature points, we can draw a graph named by feature graph as shown in Fig. 3.

Based on the feature graph, we define a marginal pose as a pose that allows the nose point N to be on the $\overline{LM}$ or $\overline{RM}$, which is the connected line of eye and mouth in right and left direction. We set up the statistical mean of 100 sample faces used for the generative model as the marginal pose.

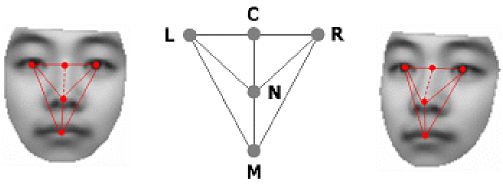

Fig. 3. Feature graph drawn from five detected feature points

3.2 Pose Approximation Procedure

To initialize the fitting parameters, we approximately perform pose estimation to a given input image by going through the following steps.

- Compensate the rotation angle to z-axis. An idea for achieving it is that each y coordinate of left and right eye in the image frame must be corresponded.
- Normalize the range of face region. Since each the face region of people is positioned at different range with each other, it is necessary for it to be ranged in same scope. We solve this problem by being identical the distance between C and M in y direction to the generative model' s.
- Decide if the given input face is frontal view or not. Inequality (9) shows the criterion we use.

$$\left|(\left|L_x - R_x\right|/2) - M_x\right| \le T_f \tag{9}$$

where T_f is a threshold value.

- Decide if the given input face is in range of the marginal pose or not. We can make a criterion for deciding it as the following:

$$f(x,y) = y - \left(\frac{L_y - M_y}{L_x - M_x}\right)x - \left(\frac{L_x M_y - L_y M_x}{L_x - M_x}\right) \tag{10}$$

- Decide if the given input image is right view or left view. The decision criterion of view is obtained from the fact that $\angle LCN$ may be an obtuse angle or an acute angle with respect to direction of view. By applying the second law of cosine in a triangle, $\angle LCN$ can easily be expressed.
- Estimate the pose of the given input image approximately. Assume that the input image is left view and its pose angle is smaller than the marginal pose. Then, our idea to estimate the pose of the input image is originated from the fact that the area of ΔLNM is also linearly changed when the pose is varied from frontal to the marginal pose. If the area of ΔLNM in the generative model with frontal pose is denoted by A_m and the marginal pose of the generative model is denoted by ϕ_m, the pose ϕ of an input image satisfying with the above conditions can be approximately estimated as defined in the following:

$$\phi = \frac{(A_m - A)}{A_m} \phi_m \tag{11}$$

where A is equal to the area of ΔLNM in the input image.

4 Fitting the 3D Model to a 2D Image

Shape and texture parameters of the generative 3D model are estimated by fitting it to a given input face. This is performed iteratively as close as possible to the input face. Fitting algorithms, called stochastic Newton optimization (SNO) and inverse compositional image alignment (ICIA) [5] were utilized in [3] and [4], respectively. It is generally addressed that SNO is more accurate but computationally expensive and ICIA is less accurate but more efficient in computation time [4].

We also explore the ICIA algorithm as a fitting method to guarantee the computational efficiency. When an input image is selected, the initial coefficients of shape and texture are set to zero and the projection parameters are appropriately decided by the pose approximation method introduced in previous section. Then, fitting steps are iterated until convergence to a given threshold value, minimizing the texture difference between the projected model image and the input image. During the fitting process, texture coefficients are updated without an additive algorithm at each iteration. But in case of shape coefficients, their updated values are not acquired with ease because of the nonlinear problem of structure from motion (SFM) [11]. To solve it, we recover the shape coefficients using SVD based global approach [12] after the convergence.

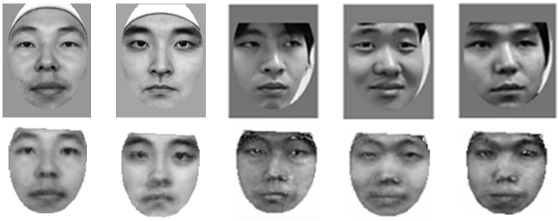

Fig. 4. Fitting results

5 Experimental Results

We evaluate our face recognition system based on a 3D model generated using proposed representation algorithm and pose approximation method in identification scenario [13]. For this, we constructed two databases from the Korean. One is composed of 3D face scans collected by a Geometrix Facevision 200, which is a stereo-camera based capturing device offering a 3D face scan including approximately 30,000 ~ 40,000 vertices and corresponding 2D texture image, the other is composed of 2D face images acquired from a general web camera. The data in the former have only frontal pose, therefore, are used for model generation and a few experiments with respect to the frontal poses. On the contrary, the latter was collected from 21 subjects during 3 sessions and contains 63 images with frontal pose and 51 images with profile pose ranged from 15°to 45°.

The experimental results to frontal and profile faces are depicted in Table 1 and 2. They report average recognition rate with rank 1 and fitting time. In both experiments, we utilized the L2 norm as a distance metric [13] and applied 50% as basis selection rate in texture and shape space.

The aim of the first experiment is to verify the influences of shape and texture parameters used as an identity parameter with respect to the system performance. Total 20 trials were done by randomly selecting 21 out of 62 subjects in 3D face database a trial. From the report of Table 1, we could find that shape parameter does not have great effects on system performance and model fitting time to only texture parameter is more efficient than to both shape and texture parameters on 1.73GHz Pentium-M and 1GB RAM. Also, we conducted the tests on 2D database containing the frontal and profile face images captured by a web camera. In this test, we selected 21 images and 51 images as probe set for frontal and profile pose, respectively.

Table 1. Recognition accuracy with rank 1 to frontal faces in 3D face database

	Only Texture	Texture & Shape
Average Recognition Rate (%)	92.1	93.3
Average Fitting Time (s)	0.9	5.6

Table 2. Recognition accuracy with rank 1 to frontal and profile faces in 2D face database

	Frontal	Profile
Average Recognition Rate (%)	89.9	83.2

6 Conclusion

A 3D model-based face recognition system was presented in this paper. A Korean 3D face model was generated from 100 3D face data. We introduced a method reducing and being identical with the number of vertex points and polygons constituting 3D shape for face representation. Practically, our 3D face model was represented with 4014 vertex points and 7980 polygons.

In the mean time, ICIA algorithm was adopted to fit the model to an input image. Especially, to initialize the fitting parameter, we proposed pose approximation method based on a marginal pose and the topology of some key features.

We found some facts from the experiments on the 3D and 2D databases. First, the shape parameters as identity parameter did not meaningfully improve the system performance in spite of increased computation time. Second, the experimental results showed that 3D model based face recognition system using the proposed representation method was further more efficient in computation time than the works in [4] even if it is somewhat less accurate. Lastly, we confirmed that in some measure

In future, we will plan to research solutions on the illumination problem, which is one of the most important issues in face recognition.

Acknowledgement. This research was performed for the Promotion and Support Program of Strategic Industry Innovation Cluster, one of the Collaboration Programs for Industry, Academy and Research Institute funded by City of Seoul, Korea.

References

1. Papatheodorou, T., Rueckert, D.: Evaluation of 3D Face Recognition Using Registration and PCA. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546, pp. 997–1009. Springer, Heidelberg (2005)
2. Blanz, V., Vetter, T.: A Morphable Model for the Synthesis of 3D Faces. In: Computer Graphics, Annual Conference Series(SIGGRAPH), pp. 187–194 (1999)
3. Blanz, V., Vetter, T.: Face Recognition Based on Fitting a 3D Morphable Model. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(9), 1063–1074 (2003)
4. Romdhani, S., Vetter, T.: Efficient, Robust and Accurate Fitting of a 3D Morphable Model. In: IEEE International Conference on Computer Vision (2003)
5. Baker, S., Matthews, I.: Equivalence and Efficiency of Image Alignment Algorithms. In: CVPR (2001)
6. Horn, B.K.P., Hilden, H.M., Negahdaripour, S.: Closed-Form Solution of Absolute Orientation Using Orthonormal Matrices. Journal of the Optical Society of America A 5, 1127–1135 (1988)
7. Besl, P.J., Mckay, N.D.: A Method for Registration of 3D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2), 239–256 (1992)
8. Lu, X., Jain, A., Colbry, D.: Matching 2.5D Face Scans to 3D Models. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(1), 31–43 (2006)
9. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cognitive Neuroscience 3, 71–86 (1991)
10. Vetter, T., Poggio, T.: Linear Object Classes and Image Synthesis from a Single Example Image. IEEE Transactions on Pattern Analysis and machine Intelligence 19(7), 733–742 (1997)
11. Bascle, B., Blake, A.: Separability of pose and expression in facial tracking and animation. In: Sixth International Conference on Computer Vision, pp. 323–328 (1998)
12. Romdhani, S., Canterakis, N., Vetter, T.: Selective vs. Global Recovery of Rigid and Non-rigid Motion. Technical report, CS Dept., Univ. of Basel (2003)
13. Moon, H., Phillips, P.: Computational and Performance Aspects of PCA-Based Face-Recognition Algorithms. Perception. 30, 303–321 (2001)