

Pose Invariant Face Recognition by Face Synthesis

Mun Wai Lee & Surendra Ranganath*
Department of Electrical and Computer Engineering
National University of Singapore
4 Engineering Drive 3
Singapore 117576
*elesr@nus.edu.sg

Abstract

This paper describes a face recognition system based on a recognition-by-synthesis approach. Given an image of an unknown face, the pose of the face is first estimated by matching it to a 3D deformable face model which can encode shape as well as texture. This model is a composite of three sub-models: edge model, color model and a wire frame model which jointly describe the shape of the face and various facial features. After estimating the pose of the unknown face, the system synthesizes face images of known subjects in the same pose as the unknown face. A least squares procedure is used for face synthesis to best approximate the lighting in the unknown face image in terms of the training face images. The unknown face is finally classified as the subject whose synthesized image is most similar. The novelty of this method lies in the use of the composite 3D deformable face model for face analysis to yield the 3D shape and texture face representation, and thereby facilitate face synthesis and pose invariant face recognition. Experimental results show that the method is capable of determining pose and recognizing faces accurately over a wide range of poses and with varying lighting conditions. Recognition rates of 92.3% have been achieved which is 18.4% higher than that of the PCA method in a comparative evaluation.

1 Introduction

In this paper, we address the problem of pose invariant face recognition. This is a difficult problem because the pose of the face greatly affects how the face appears in an image. Our system seeks to recognize faces over a wide range of poses using only a small number of training images.

In the literature, face recognition methods can generally be categorized into (1) geometric feature-based methods [6]; (2) statistical appearance-based methods [1,9]; and (3) neural network-based methods [4], etc. The above methods have their own merits, but they are generally 2D-based and therefore not inherently pose invariant.

In recent years, more attention has been focused on pose invariant face recognition. One approach is to use local feature extractors with convolution neural networks [8] or Gabor wavelets with the dynamic link architecture [7]. Another approach is based on the linear object class concept [2] where a linear combination of a number of 2D face images is used to synthesize new face images in different poses. Both approaches can

achieve some degree of pose invariance but they cannot handle large pose differences where appearance changes significantly and occlusions occur in parts of the face.

In computer graphics, the wire frame model (WFM) is commonly used to represent 3D objects for efficient 2D image rendering. The WFM has also been widely used to encode and synthesize human faces in model-based coding for low bit-rate communication. We therefore adopted a recognition-by-synthesis approach and exploited the use of image synthesis techniques from computer graphics and model-based coding to achieve pose invariant face recognition. Using a few training face images of each subject in different poses, the system extracts a 3D face representation which encodes 3D shape and texture. Using this 3D representation, the system is able to synthesize 2D appearances of a face in a wide range of poses. The synthesized images are then used for recognition.

In this system, a 3D face model is matched to face images. This model matching process yields estimates of face pose and is used to set up the 3D face representations during training. The face model acts as an important reference for establishing correspondences between faces in different images. This model is deformable to accommodate faces of different shapes. The deformation functions and the matching process are extensions of the method used in [5] for object matching.

In the following, Section 2 describes the 3D deformable face model and Section 3 describes the face recognition procedure. Experimental results are presented in Section 4 and Section 5 gives the conclusions.

2 3D Deformable Face Model

2.1 Face Model

A 3D deformable face model is used for matching an input face image. This model is a composite of three sub-models: edge model, color model and a wire frame model (Figure 1). The edge and color models are used for matching whereas the WFM is used for image synthesis.

The edge model defines the outlines of the face as well as various facial features such as the eyebrows, eyes, nose, mouth and ears. In this model, each edge is defined by a set of 3D coordinates equally spaced along the edge. During matching, these edge points will be influenced by an edge potential field so that the points tend to align along the edges in the image. The color model identifies facial regions of low intensity, high intensity or fairly homogeneous lip color. These regions are the eyebrows, eyes, nostrils or mouth. Each region is defined by a set of points uniformly distributed in the region. During matching, these model regions will be attracted to regions in the input image which have similar intensity/color characteristics. The points defining eyebrows and nostrils will be attracted towards regions of low intensity, the eyes towards regions either of low intensity (for iris) or high intensity (for whites of eyes), and the mouth towards regions that exhibit lip color. In the WFM, the face surface is divided into 100 triangles which are defined by 59 vertices. The WFM approximates the 3D structure of the face and can be used to synthesize face images efficiently and accurately.

During model matching, the face model is projected onto a face image so that the various facial features are aligned. To match accurately, the model undergoes complex local deformation as well as geometric rigid body transformation. Correspondences are established among the three models, so that they are integrated into one composite face model and undergo the same complex deformation and geometric transformation. After matching is established using the edge and color models, the WFM can be readily adapted to the image and used for synthesis.

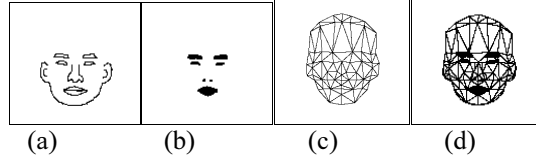


Figure 1: 3D Deformable Face Model. (a) Edge Model (b) Color Model (c) Wire Frame Model (d) Composite Model.

2.2 Complex Deformation

Different people have varying face shapes and features. Complex deformation allows the model to handle this shape variation. We use a modification of the deformation transformations used in [5], where a 2D displacement field was used to deform a 2D template derived from a hand drawn sketch of the object shape. Here, we extend this by using a 3D displacement field to deform our 3D face model. The deformation mapping used is

$$(x,y,z) \rightarrow (x,y,z) + (D^x(x,y,z), D^y(x,y,z), D^z(x,y,z)) \quad (1)$$

where $D^x(x,y,z)$, $D^y(x,y,z)$ and $D^z(x,y,z)$ are the displacement functions along the three coordinates axes. The space of the displacement functions is spanned by the following orthogonal bases:

$$\begin{aligned} e_{lmn}^x(x,y,z) &= (2 \sin(\pi lx) \cos(\pi my) \cos(\pi nz), 0, 0), \\ e_{lmn}^y(x,y,z) &= (0, 2 \cos(\pi lx) \sin(\pi my) \cos(\pi nz), 0), \\ e_{lmn}^z(x,y,z) &= (0, 0, 2 \cos(\pi lx) \cos(\pi my) \sin(\pi nz)) \end{aligned} \quad (2)$$

where $l, m, n = 1, 2, 3 \dots$, and the displacement function is given by

$$\begin{aligned} D(x,y,z) &= (D^x(x,y,z), D^y(x,y,z), D^z(x,y,z)) \\ &= \sum_{l=1}^L \sum_{m=1}^M \sum_{n=1}^N \frac{\xi_{lmn}^x \cdot e_{lmn}^x + \xi_{lmn}^y \cdot e_{lmn}^y + \xi_{lmn}^z \cdot e_{lmn}^z}{\lambda_{lmn}}. \end{aligned} \quad (3)$$

where $\lambda_{lmn} = \alpha \pi^2 (l^2 + m^2 + n^2)$ are normalizing constants and α is a scale constant. The deformation parameters are

$$\underline{\xi} = \left\{ \left(\xi_{lmn}^x, \xi_{lmn}^y, \xi_{lmn}^z \right) \mid l, m, n = 1, 2, \dots \right\} \quad (4)$$

In our implementation, we set L, M, N to 3 as a good trade-off between matching accuracy and computation. Figure 2 shows some examples of faces with different shapes in the frontal pose that can be generated by varying the deformation parameters.

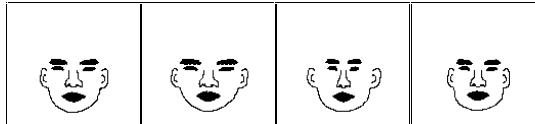


Figure 2: Complex Deformation.

2.3 Geometric Transformation

In geometric transformation, the 3D model is first projected onto a 2D view plane. Thereafter, the 2D face template undergoes scaling and translation. An orthographic projection is assumed because the distance between the face and the camera is generally large, relative to the size of the face. Figure 3 shows examples of 2D face templates generated for different views.

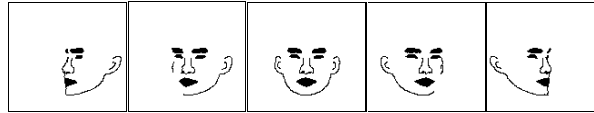


Figure 3: View Plane Projection.

The projection to the view plane is given by the transformation

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ -\sin \theta \sin \phi & \cos \phi & -\cos \theta \sin \phi \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (5)$$

where θ and ϕ are the slant and tilt angles of the face, respectively. Thereafter, the face is rotated in-plane by roll angle γ using

$$\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} \cos \gamma & -\sin \gamma \\ \sin \gamma & \cos \gamma \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (6)$$

Subsequently, scaling and translation are incorporated by

$$\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow s \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} d_x \\ d_y \end{bmatrix} \quad (7)$$

where s is the scale factor and d_x , d_y are the displacements along the two view plane axes.

3 Face Recognition

3.1 Overview of training and recognition procedures

During training, a number of training images of each subject in different orientations is used to extract a 3D shape and texture representation for that subject. The stages in the training procedure are (1) image pre-processing and model matching, and (2) setting up the 3D face representation. These 3D representations are central to face recognition.

A given image in which a face is to be recognized undergoes the following processing stages: (1) image pre-processing and model matching, (2) image synthesis, and (3) classification. The first stage is the same as in the training procedure, and is used to estimate the face pose. Using this information, synthesized faces of known subjects in the estimated pose are generated from their 3D face representations. Classification is based on similarity between these synthesized faces and the unknown face. The rest of this section describes the stages in more detail.

3.2 Image Pre-processing and Model Matching

Image pre-processing involves skin color segmentation and edge extraction. The segmentation uses simple thresholding and morphological operations. Using similar techniques, regions of low intensity, high intensity and lip color are also extracted. These are likely regions for the eyebrows, eyes, nostrils and mouth and can be matched to similar regions defined in the color model. Canny edges are extracted within the skin colour region, and model matching is confined to within this region.

During model matching, the 3D model is transformed so that it matches (aligns) with the face in the input image. To measure the quality of matching, an objective function is defined. This objective function consists of three weighted terms: edge energy, color energy and deformation cost. The edge and color energy describe the quality of matching for the edge and color features respectively, and the deformation cost penalizes modification to the shape of the face model.

For matching, edge and color *potential fields* are defined based on the edge and color features of the input image. These potential fields are used to exert forces on the face model and influence its transformation parameters during matching. The matching is implemented as an iterative search process which seeks to find a set of transformation parameters that minimizes the objective function. A multiresolution approach is used to achieve fast convergence. Figure 4 shows how the template changes at various iterations as it converges on a face image.

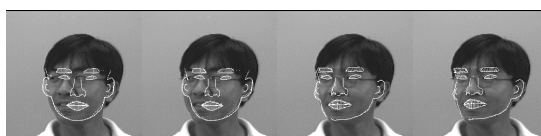


Figure 4: Model Matching at Iterations 0, 1, 3 and 10.

3.3 3D Face Representation

For each subject, a number of training face images in different poses is used to derive the 3D face representation. We first perform the two stages of image pre-processing and model matching on these images. During the matching process, the edge and color sub-models undergo transformation to align themselves with the features in the input face. Since the WFM is integrated with the edge and color sub-models, the WFM also undergoes the same transformation. Therefore, after matching, the WFM can be readily adapted to the input face image, as illustrated in Figure 5.

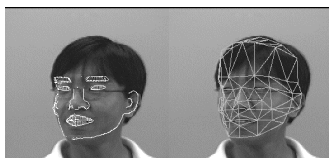


Figure 5: Wire Frame Model Adaptation After Matching.

After performing model matching on all the training images, the 3D face representation for each subject in the database is estimated. There are two aspects to this representation: shape representation and texture representation. The shape representation for a subject is defined by the positions of the WFM vertices in the 3D space. It is obtained as a weighted average of the positions of these vertices in the deformed WFMs after adapting to the training images of the subject.

The texture representation, on the other hand, is an N -dimensional vector of luminance values obtained at N points on the 3D face model. These luminance values are sampled from the training images after the model matching process, which specifies the model points to image correspondences. Let N_i^s represent the number of training images of subject s and let $\{t_i^s, i = 0, 1, \dots, N_i^s - 1\}$ represent the set of *texture vectors* for the subject, obtained by sampling the luminance values in each of the matched training images. A mean texture vector m^s , for subject s is obtained as

$$m_j^s = \frac{\sum_{i=0}^{N_i^s-1} a_{i,j}^s t_{i,j}^s}{\sum_{i=0}^{N_i^s-1} a_{i,j}^s} \quad (8)$$

where m_j^s is the j th element of m^s , $t_{i,j}^s$ is the j th texture element obtained from i th training image, and $a_{i,j}^s$ represents the projected area of the WFM plane in which the j th texture point lies, on the i th training image. If this WFM plane is hidden in the i th training image, which means that the j th texture point must be hidden, $a_{i,j}^s$ is set to zero.

The vector m^s represents the average face texture for subject s under different pose and lighting conditions. As the face texture appearance may change due to lighting conditions and lighting geometry, an average representation would be unable to characterize the variations adequately. We therefore use a set of vectors to encode texture deviations from the average. This set of vectors for subject s $\{b_i^s, i = 0, 1, \dots, N_i^s - 1\}$ is defined as

$$b_{i,j}^s = \begin{cases} t_{i,j}^s - m_j^s, & \text{if } j\text{th texture point is visible in image } i, \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

where $b_{i,j}^s$ is the j th element of b_i^s . Assuming that $\{b_i^s\}$ are linearly independent, we can use them to characterize the subspace of texture variations seen in the training set of subject s . With this basis set, any texture vector \tilde{y}^s in this subspace can be expressed as

$$\tilde{y}^s = m^s + \sum_{i=0}^{N_i^s-1} x_i b_i^s \quad (10)$$

By varying the coefficients $\{x_i\}$, we can synthesize multiple face images for subject s , simulating the effects of varying lighting conditions and geometry. This is useful during face matching, where a face image needs to be synthesized from the database, having the approximate lighting characteristics as the unknown face image. This serves to discount the effects of lighting while matching. A least squares procedure is described in Section 3.4 to incorporate this idea.

After obtaining the 3D face representation of a known subject from his training images, images of the face can be synthesized in different poses by applying view plane projection, scaling and translation. Figure 6 shows the synthesized images generated at different slant angles. As we have not included the necks and shoulders, the boundaries of the faces are sharp. But generally, the synthesized faces are realistic.



Figure 6: Face Synthesis with Different Poses.

3.4 Face Classification

Given an image of an unknown face, the pose is first estimated by model matching. Using the estimated pose, the texture points in the 3D face representation of subject s can be projected onto the same pose as the unknown face, thereby obtaining the locations of where these texture points should appear in the unknown face if the face belonged to subject s . The intensity values at these locations in the image are sampled to obtain the N -dimensional input texture vector called $\tilde{\mathbf{y}}^s$. Firstly, this vector may be different from the texture vector of the unknown face due to different lighting conditions. Secondly, if the unknown face does not belong to subject s , the projection of the texture points will be erroneous because of the differences in face shape, and hence cause the texture vector to be different. It is this error that needs to be used as a match measure, while discounting errors due to lighting.

A least squares strategy is used for this purpose. We first obtain an error measure for representing the input texture $\tilde{\mathbf{y}}^s$, by the texture subspace of subject s . To do this, a vector \mathbf{y}^s is defined as

$$\mathbf{y}_j^s = \begin{cases} \tilde{\mathbf{y}}_j^s - \mathbf{m}_j^s, & \text{if } j\text{th feature point is visible in input image} \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

and a basis matrix for subject s is constructed as $\mathbf{A}^s = \{\mathbf{b}_0^s \ \mathbf{b}_1^s \ \dots \ \mathbf{b}_{N_f^s-1}^s\}$.

Then, the vector $\mathbf{A}^s \mathbf{x}$ generates all possible lighting variations contained in the subspace of subject s . Hence solving $\mathbf{A}^s \mathbf{x} = \mathbf{y}^s$, in the least squares sense, yields a synthesized face for subject s , that matches the lighting conditions in the unknown face as best as possible. The least squares solution is given as $\hat{\mathbf{x}} = (\mathbf{A}^{sT} \mathbf{A}^s)^{-1} \mathbf{A}^{sT} \mathbf{y}^s$, and the least squares error is

$$e^s = \|\mathbf{A}^s \hat{\mathbf{x}} - \mathbf{y}^s\|^2 \quad (12)$$

If the lighting conditions are well approximated, this error must contain a substantial component that is related to the identity mismatch of subject s and the unknown face. The least squares procedure is repeated for every subject in the database, and the identity of the unknown face is assigned to the subject with the smallest least squares error.

4 Experimental Results

4.1 Image Database

A database of face images was acquired for training and testing the face recognition system. The database consisted of 130x120 color images of 15 subjects. These images were divided into 11 subsets according to imaging conditions (orientation, lighting or size). The exemplar images from the subsets are shown in Figure 7. Some of the images in the database were used for training. Of the remainder, 4 images of each subject from each subset was used for testing. There were a total of 660 test images (15 subjects \times 11 subsets \times 4 images/subject/subset).

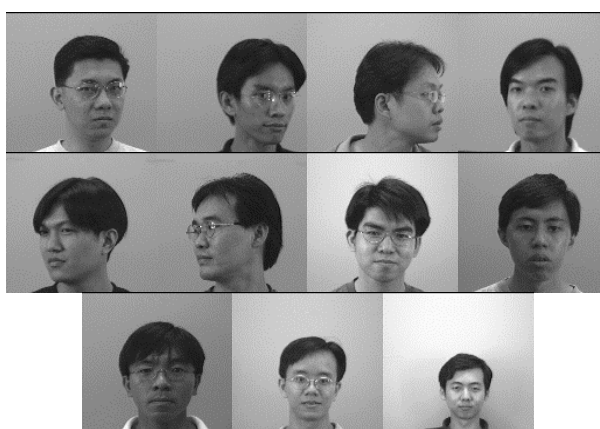


Figure 7: Example Images of Subsets Showing Pose and Lighting Variations

4.2 Pose determination

To examine the accuracy of pose determination, the system was tested on a number of face images of different persons in varying orientations. Figure 8 shows a few example images to which the model was matched. The result shows that our method is able to match the model to the face and determine the orientation accurately over a wide range of poses (from frontal to profile views). Most of the visible facial features, such as the eyebrows, eyes, noses, mouths and ears are accurately located in all the images as is evident from the superimposed model. The matching is also tolerant to scale and varying illumination.

It is insightful to obtain an objective measure for the accuracy of pose determination. As the true poses of the input faces are not available, these were manually estimated. The 3D model was used for this purpose. For each image, the model was rotated with an interactive program (with angular resolution of 1°), such that the orientation of the model was visually perceived to be the same as that of the input face. The final orientation of the model is used as the true orientation of the input face (see third column of Figure 8) and the error in orientation estimates is measured with respect to this. The absolute error of pose determination, on average, was less than 5° . In most cases, the error was within $\pm 10^\circ$. For profile or near profile poses, such as in the third image in Figure 8, the error in the slant angle may be large. However, at these extreme orientations, appearance changes due to slant angle variation is not significant.

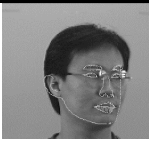

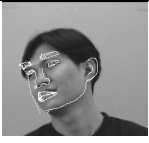



| Input Face | Simulated 3D Model | True Orientations in Degrees (and pose estimation errors) |
|---|---|---|
|  |  | slant : -30 (2.1) tilt : 0 (0.6) roll : 0 (0.0) |
|  |  | slant : 23 (6.8) tilt : -12 (0.5) roll : -12 (2.9) |
|  |  | slant : 55 (13.7) tilt : 0 (2.1) roll : 0 (0.0) |

Figure 8: Accuracy of Orientation Estimation. The first column shows faces with the matched model superimposed. The second column shows simulated 3D faces in true orientations. The third column shows true orientation angles and orientation estimation errors in parentheses.

4.3 Face Recognition

To examine the face recognition performance of the method, five experiments were conducted with the number of training images for each subject set to 1, 3, 7, 9 or 10. The test set of 660 images did not include the training images. Results are shown for three tests:

Test A. Recognition by synthesis method as described in this paper.

Test B. Recognition by synthesis method, where face synthesis is performed only with the mean vector in equation (8), without using the texture subspace.

Test C. Recognition by PCA method [9]. In this test, the input images (both training and testing) were first pre-processed by skin color segmentation to estimate the centroids and sizes of the faces. By appropriate scaling and translation, the faces were normalized to the same scale and location. The normalized faces were used to construct the eigenfaces. A single eigenspace was constructed from all the training images. Classification was based on the nearest neighbor method using the PCA coefficients. There was no dimensionality reduction, and all PCA coefficients were used.

The results of the three tests are shown in Table 1. The recognition rates improve with increasing number of training images for all the three tests. With 10 training images per subject, the recognition rate of the recognition-by-synthesis method (Test A) is 92.3%. The method performed consistently better than the PCA, especially with small number of training images. Comparison of results from Test A and Test B shows significant improvement made with the use of texture basis vectors for face synthesis. This indicates that the use of basis vectors makes better use of available information in the training images by encoding the texture variations in these images.

| Experiments | Training images per subject | | | | |
|-------------|-----------------------------|-------|-------|-------|-------|
| | 10 | 9 | 7 | 3 | 1 |
| TestA | 92.3% | 89.5% | 80.2% | 69.4% | 56.2% |
| TestB | 69.4% | 69.1% | 68.3% | 65.6% | 56.2% |
| TestC | 73.9% | 67.9% | 61.8% | 43.2% | 21.8% |

Table 1: Recognition rates for various experiments and number of training images.

5 Conclusions

Recent works on face recognition consider the difficult problem of achieving pose invariance where appearance variations due to pose are often more significant than variations due to identity.

We have implemented a pose invariant face recognition method using a recognition-by-synthesis approach. It uses a 3D deformable face model which consists of edge and color features as well as a wire frame model. By matching this model to training face images, we were able to construct 3D shape and texture representations for each subject's face. For recognizing an unknown face, 2D appearance of faces of all known subjects were generated in the same pose as the unknown face, while approximating the lighting characteristics in the unknown face image as best as possible. Classification was then based on the similarities between these synthesized faces and the unknown face. The novelty of this method lies in the use of the 3D deformable face model for face analysis and the 3D face representation scheme for synthesis and classification.

Experimental results show that the method, which is fully automatic, can achieve a high recognition rate. The advantages of the method include invariance to translation, scale, orientation and lighting condition, and can work with a small number of training images. The model matching process is however computationally expensive, and makes the method unsuitable for real time applications.

References

- [1] P.N. Belhumeur, J.P. et. al., "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Trans. PAMI*, v 19 n 7 Jul 1997. p 711-720.
- [2] D. Beymer, T. Poggio, "Face recognition from one example view," *ICCV*, 1995, pp. 500-507.
- [3] J. Canny, "A computational approach to edge detection," *IEEE Trans. PAMI*, v 8, n 6, pp. 679-698, Nov 1986.
- [4] A. J. Howell, H. Buxton, "Towards unconstrained face recognition from image sequences," *2nd Int. Conf. on Automatic Face and Gesture Recognition*, 1996, pp. 224-229.
- [5] A.K. Jain, Y Zhong, S Lakshmanan, "Object matching using deformable templates," *IEEE Trans. PAMI*, vol. 18, no. 3, pp. 267-277, March 1996.
- [6] T. Kanade, *Computer recognition of human faces*, Birkhäuser Verlag, Stuttgart Germany, 1977.
- [7] M. Lades, et. al., "Distortion Invariant object recognition in the dynamic link architecture," *IEEE Trans. Comp.*, vol. 42, no. 3, pp. 300-311, 1993.
- [8] S. Lawrence, C. L. Giles, A. C. Tsoi, "Convolutional neural networks for face recognition," *CVPR* 1996, PP. 217-222.
- [9] M. Turk and A. Pentland, "Eigenfaces for recognition." *Journal of Cognitive Neuroscience*, vol. 3(1), pp. 71-86, 1991.