Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition Letters 29 (2008) 712–723
Pattern Recognition Letters
www.elsevier.com/locate/patrec

Recovering the 3D shape and poses of face images based on the similarity transform

Hei-Sheung Koo, Kin-Man Lam *

Centre for Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong

Received 21 November 2006; received in revised form 6 November 2007
Available online 23 December 2007

Communicated by N. Pears

Abstract

In this paper, a new algorithm is proposed to derive the 3D structure of a human face from a group of face images under different poses. Based on the corresponding 2D feature points of the respective images, their respective poses and the depths of the feature points can be estimated based on measurements using the similarity transform. To accurately estimate the pose of and the 3D information about a human face, the genetic algorithm (GA) is applied. Our algorithm does not require any prior knowledge of camera calibration, and has no limitation on the possible poses or the scale of the face images. It also provides a means to evaluate the accuracy of the constructed 3D face model based on the similarity transform of the 2D feature point sets. Our approach can also be extended to face recognition to alleviate the effect of pose variations. Experimental results show that our proposed algorithm can construct a 3D face structure reliably and efficiently.

Keywords: 3D reconstruction; Similarity transform; Genetic algorithm; Face recognition

1. Introduction

Many face recognition methods have been developed over the past few decades. Most of those based on frontal views without expression and under controlled lighting can achieve a reasonably high performance level (Chellappa et al., 1995; He et al., 2005). However, face recognition techniques based on 2D images are strongly affected by variations in pose, which is the primary source of difficulties with face recognition. The performance of face recognition algorithms suffers dramatically when a large variation in pose is present in a query image, especially when the training data have few non-frontal images. A sensible way to improve the recognition performance for face images under arbitrary poses is to use multiple training images under different poses. However, face images under different poses may not be available in some applications, and the use of multiple faces will greatly increase both the size of a database and the computation required for matching. 3D deformable models (Lee and Ranganath, 2003; Ansari and Abdel-Mottaleb, 2005; Jiang et al., 2005) have therefore been applied for pose-invariant face recognition. 3D face models have also been adopted in face tracking and facial animation (Ahlberg and Forchheimer, 2003). However, for these applications, an accurate 3D face structure may not be necessary to achieve a good performance level.

Face modeling can be achieved by extracting the motion and shape information about a 3D face model from the face viewed at different times or from using multiple cameras at different angles (Jerian and Jain, 1991; Tomasi and Kanade, 1992; Huang and Netravali, 1994). In particular, the problem of extracting the shape and motion parameters of a moving 3D object from a 2D image sequence is known as the structure from motion problem

* Corresponding author. Tel.: +852 2766 6207; fax: +852 2362 8439.
E-mail address: enkmlam@polyu.edu.hk (K.-M. Lam).

0167-8655/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2007.11.018

(SfM). In SfM, the 3D information about a collection of discrete structures, such as lines, curves and points, is recovered from a 2D collection of such lines, curves and points. 2D images are formed by projections from the 3D world. SfM recovers the original 3D information by inverting the effect of the projection process. Two well-known projection models are the perspective model and the orthographic model. Perspective projection is a realistic model of the imaging process, whereas orthographic projection yields easy-to-solve models that are applicable in some simple cases. Orthographic projection is used primarily because it gives rise to mathematically tractable equations. It is a reasonable approximation for objects that subtend a small field of view and whose distance does not change dramatically.

Much research has been conducted on determining the motion and structure of moving rigid objects under orthographic projection. Ullman (1979) proved that four point correspondences over three views yield a unique solution to motion and structure. It is impossible to determine the motion and structure uniquely from two orthographic views no matter how many point correspondences one may have. Huang and Lee (1989) and Hu and Ahuja (1991) presented a linear algorithm to obtain the 3D motion and structure parameters. Shapiro et al. (1995) considered the affine epipolar line properties and solved the affine epipolar line equation, and then determined all the unknown camera motion parameters. Tomasi and Kanade (1992) and Morita and Kanade (1997) developed a factorization method to recover shape and motion under an orthographic projection model. They used the singular value decomposition technique to factorize the measurement matrix into two matrices, which represent the object shape and the camera motion, respectively. Xirouhakis and Delopoulos (2000) extracted the motion and shape parameters of a rigid 3D object by computing the rotation matrices via the eigenvalues and eigenvectors of appropriate defined 2×2 matrices, where the eigenvalues are the expression of four motion vectors in two successive transitions.

In our proposed algorithm, three or more face images of the same subject are used to construct a 3D face model. One of them is a frontal view, while the other images are under arbitrary poses. To recover the 3D face structure, the 2D frontal-view face image is adapted to the CANDIDE model. Then, the pose and the feature-point depths of the CANDIDE model are adjusted to fit the poses of the respective 2D non-frontal-view face images in such a way that the feature-point distance between the projected 3D model and the 2D face images under different poses is minimized under the similarity transform (Werman and Weinshall, 1995). However, searching for the best pose to provide the best alignment is so computationally intensive that an exhaustive search is impossible. Thus, the genetic algorithm (GA) is employed to search the optimal poses and depths of the feature points of the face model, which are computed iteratively so as to fit the face images accurately and efficiently. In addition, our method does not need any camera calibration. However, it requires that all the face images be of the same facial expression, and assumes that the heads are under rigid motion.

The similarity transform is also used to measure the accuracy of the constructed 3D face model. After constructing a face model, it can be compared to those training 2D face images used in the construction by means of the similarity transform. The Levenberg–Marquardt method is used to optimize the alignment of the face model to the respective face images. If the structure of the constructed face model is similar to that of the face image, the distance will be small. In summary, our algorithm can construct the 3D face model and estimate the poses of the respective face images, and in the meantime can provide a measurement of the accuracy of the model. The next section will give the details of our algorithm.

2. Construction of the 3D face model

Our algorithm can construct the 3D face structure of a person based on a set of face images which are under different viewing angles and have a fairly neutral expression. The 3D face structure is represented by the (x, y, z) coordinates of the important facial feature points. In our algorithm, each face is considered a rigid object, and the poses and sizes of the face images are unknown. Hence, besides the 3D coordinates of the feature points, our algorithm can also estimate the poses and scales of the respective face images.

2.1. The 3D face model

To construct the 3D face model, at least three images under different poses and with a neutral expression are required. Our 3D face model is represented by $n = 15$ feature points, as illustrated in Fig. 1a. These can be located automatically or manually. Ullman (1979) proved that four point correspondences over three views can yield a unique solution to motion and structure. Thus, three or more face images under different viewing angles are required to construct the 3D face model. The first image in our experi-

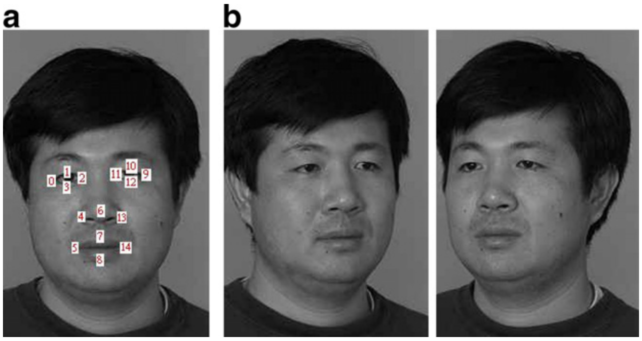

Fig. 1. (a) A frontal-view image with 15 landmark points, and (b) two more face images with different poses.

ments is a frontal-view image, and the poses of the other images are estimated with reference to the frontal-view image. Fig. 1b shows two other images of the same person under different poses.

In our proposed method, the CANDIDE model (Ahlberg, 2001) is employed. The CANDIDE model is used for initialization only in our iterative process because the 3D face structure is unknown in the first iteration. The definitions of the three axes are shown in Fig. 2. Based on the position of the important feature points, the CANDIDE model is first adapted to the frontal-view face image, as shown in Fig. 3a. Then, the CANDIDE model is rotated to the same poses as the non-frontal-view face images, and the depths of the feature points of the model are adjusted so that the feature points obtained by projecting the 3D model onto the 2D space can fit the corresponding feature points of the images accurately. Fig. 3b illustrates the adaptation of the 3D CANDIDE model to two other face images.

2.2. Our algorithm

Our algorithm can recover the structure and poses of a face based on a number of 2D images by projecting its 3D model on to the 2D plane, i.e. a 2D to 3D problem. We assume that one frontal-view face image and N ($N \geqslant 2$) non-frontal-view face images are available. The poses and scales of the non-frontal-view images with respect to the frontal-view face image are all unknown. We also assume that the n feature points in the respective face images have all been located accurately. The 3D to 2D projection is performed using the following transformation:

$$\boldsymbol{p}_i = s_i \boldsymbol{R}_{i_{2\times3}} \boldsymbol{C} + \boldsymbol{T}_i \quad \text{for } i = 1, \ldots, N, \tag{1}$$

where N is the number of non-frontal-view face images, s_i, $\boldsymbol{T}_i = [t_{i1}, t_{i2}]^{\mathrm{T}}$ and $\boldsymbol{R}_i$ denote the scaling factor, the translation matrix and the rotation matrix between the frontal-view image and the ith non-frontal-view face image, respectively. $\boldsymbol{R}_i$ can be specified as three successive rotations around the x-, y-, and z-axes, by angles ϕ_i, ψ_i and θ_i, respectively, and can be written as the product of these three rotations as follows:

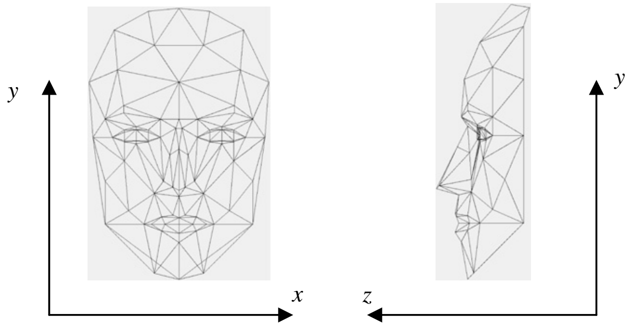

Fig. 2. The CANDIDE model in frontal view and profile view.

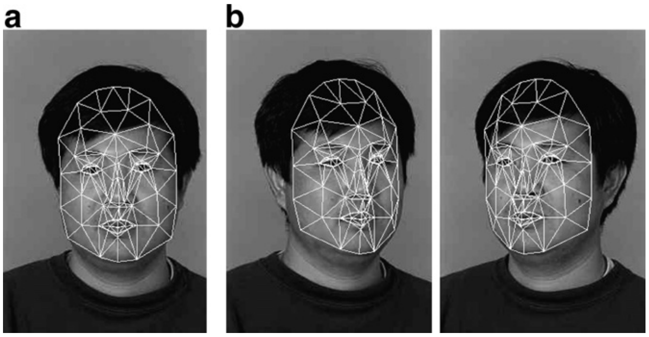

Fig. 3. (a) Face images with an adapted face model, and (b) face images under different poses adapted by the rotated face model.

$$\boldsymbol{R}_i = \begin{bmatrix} \cos\phi_i & \sin\phi_i & 0 \\ -\sin\phi_i & \cos\phi_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\psi_i & 0 & -\sin\psi_i \\ 0 & 1 & 0 \\ \sin\psi_i & 0 & \cos\psi_i \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_i & \sin\theta_i \\ 0 & -\sin\theta_i & \cos\theta_i \end{bmatrix} = \begin{bmatrix} r_{i_{11}} & r_{i_{12}} & r_{i_{13}} \\ r_{i_{21}} & r_{i_{22}} & r_{i_{23}} \\ r_{i_{31}} & r_{i_{32}} & r_{i_{33}} \end{bmatrix}. \tag{2}$$

$\boldsymbol{R}_{i2\times3}$ contains the first two rows of the 3×3 rotation matrix $\boldsymbol{R}_i$. Let n be the number of feature points in a face image. The matrix $\boldsymbol{C}$ $(=[\boldsymbol{X_C}, \boldsymbol{Y_C}, \boldsymbol{Z_C}]^{\mathrm{T}})$ is a $3 \times n$ matrix, which represents the 3D coordinates in the adapted face model. $\boldsymbol{X_C}$, $\boldsymbol{Y_C}$ and $\boldsymbol{Z_C}$ are three $n \times 1$ matrices, which are the x-, y- and z-coordinates, respectively, of the feature points in the adapted face model. $\boldsymbol{X_C}$ and $\boldsymbol{Y_C}$ are measured from the image being adapted, while $\boldsymbol{Z_C}$ is initially set at the default values of the CANDIDE model with a particular scale according to the size of the face image. $\boldsymbol{p}_i$ is a $2 \times n$ matrix which represents the 2D coordinates of the feature points in the ith non-frontal-view face images. Also, the first row and the second row of $\boldsymbol{p}_i$ represent the x- and y-coordinates, respectively.

If the pose of the face model and the depths of the feature points fit the ith non-frontal-view face images, the following equation will be a minimum:

$$\boldsymbol{DI}^2 = \frac{1}{N} \sum_{i=1}^{N} \left\| \boldsymbol{p}_i - s_i \boldsymbol{R}_{i_{2\times3}} \boldsymbol{C} - \boldsymbol{T}_i \right\|^2. \tag{3}$$

Before taking the norm of the difference between the face model and the images, we must remove the differences caused by irrelevant effects, such as the arbitrary image size under scaled orthography or the arbitrary location due to the translation and rotation of the face in the image. To remove the irrelevant effects, image alignment is performed. The alignment transformation, which is a series of transformations – including translation, scaling, and rotation – is applied to one image to obtain an optimal alignment to another image.

All the point sets to be compared are translated to their respective centroids so that the centroids become the origin of the coordinate system, and their first moments are zero. Let $\boldsymbol{M}$ $(=[\boldsymbol{X_M}, \boldsymbol{Y_M}, \boldsymbol{Z_M}]^{\mathrm{T}})$ be a $3 \times n$ matrix which represents the centered model point set. Similarly, suppose that $\boldsymbol{q}_i$ denotes a $2 \times n$ matrix which represents the centered point sets of the ith image. In other words, $\boldsymbol{q}_i$ and $\boldsymbol{M}$ are the centered point sets of $\boldsymbol{p}_i$ and $\boldsymbol{C}$, respectively, and (3) becomes

$$\boldsymbol{D2}^2 = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{q}_i - s_i \boldsymbol{R}_{i_{2\times3}} \boldsymbol{M}\|^2. \tag{4}$$

To accomplish the optimal alignment, we employ the genetic algorithm (GA) to search for the optimal solution, i.e. the optimal poses of the non-frontal-view images. GA is used because it can search for the optimal solution even in a large searching space. This approach can provide an accurate solution, although the computational time is a little bit longer. However, for many applications, such as face recognition, this 3D face reconstruction can be performed offline.

2.2.1. The chromosome

In the GA, the relative poses of the face model, adapted to the respective non-frontal-view face images, are randomly generated and evenly distributed to form the initial population. The fitness value of each candidate in a population is measured based on (4). When the population evolves, the number of candidates with the correct poses will gradually dominate. The iterative process will be stopped either when the fitness value of the population does not change significantly over a number of iterations or when a certain number of iterations have been done. Finally, the parameters of the best candidate in the population are used to represent the best poses of the face model adapted to the non-frontal-view face images.

The chromosome designed for the GA should be able to represent the solution effectively, and its length should be as short as possible. Fig. 4 illustrates the chromosome structure used in our algorithm for having N non-frontal-view face images, where ϕ_i, ψ_i and θ_i are the angles rotated about the z-, y- and x-axes, respectively, for adapting the face model to the ith face images. In our approach, the number of elements in the chromosome is therefore $3N$.

When the number of face images used to construct the face model increases, the chromosome size will also increase. Then, increasing the population size and the maximum iteration numbers is also required because the chromosomes will form a much larger solution space.

2.2.2. The optimal depths of the feature points

To minimize the fitness function in the GA, the information provided from the chromosomes is insufficient because the depths of the feature points are unknown. The following equation shows the feature-point distance between the ith face image and the projected feature points of the face model:

$$\boldsymbol{D3}_i^2 = \|\boldsymbol{q}_i - s_i \boldsymbol{R}_{i_{2\times3}} \boldsymbol{M}\|^2, \quad i = 1,\ldots,N. \tag{5}$$

To compute the distance, ϕ_i,ψ_i and θ_i are substituted into (2) to calculate $\boldsymbol{R}_i$, and then $\boldsymbol{R}_{i2\times3}$ is obtained and substituted into (5). Since the depths of the features points are the default values of the CANDIDE model in the first iteration, the initial structure of the face model is an approximation only. Therefore, the z-coordinates in $\boldsymbol{M}$ are calculated by applying partial differentiation to (5) with respect to the z-coordinates. From (5), we can calculate N different z-coordinates for $\boldsymbol{M}$. There are N different combinations between the frontal-view image and each of the N non-frontal-view images. Let $\boldsymbol{Z_{Mi}}$ be the $n \times 1$ matrix which represents the z-coordinates in $\boldsymbol{M}$ constructed based on the frontal-view image and the ith non-frontal-view image. We also denote $\boldsymbol{r1}_i = \lfloor r_{i_{13}} \quad r_{i_{23}} \rfloor, \boldsymbol{r2}_i = \lfloor r_{i_{31}} \quad r_{i_{32}} \rfloor$, $\boldsymbol{r3}_i = \lfloor r_{i_{11}} \quad r_{i_{12}} \rfloor, \boldsymbol{r4}_i = \lfloor r_{i_{21}} \quad r_{i_{22}} \rfloor$ and $\boldsymbol{M}_{xy}$ (=$[\boldsymbol{X_M}, \boldsymbol{Y_M}]^{\mathrm{T}}$). Then, by applying partial differentiation to (5) with respect to $\boldsymbol{Z_{Mi}}$, we have

$$\boldsymbol{Z}_{\boldsymbol{M}i}^{\mathrm{T}} = \frac{\boldsymbol{r1}_i \cdot \boldsymbol{q}_i + s_i \cdot r_{i_{33}} \cdot \boldsymbol{r2}_i \cdot \boldsymbol{M}_{xy}}{s_i \cdot \boldsymbol{r1}_i \cdot \boldsymbol{r1}_i^{\mathrm{T}}}, \quad i = 1,\ldots,N. \tag{6}$$

Then, (6) is substituted into (5) and partial differentiation with respect to s_i is applied to (5). Denote $\boldsymbol{a}_i = \boldsymbol{r3}_i \cdot \boldsymbol{M}_{xy} + r_{i_{13}} \cdot \frac{r_{i_{33}} \cdot \boldsymbol{r2}_i \cdot \boldsymbol{M}_{xy}}{\boldsymbol{r1}_i \cdot \boldsymbol{r1}_i^{\mathrm{T}}}$ and $\boldsymbol{b}_i = \boldsymbol{r4}_i \cdot \boldsymbol{M}_{xy} + r_{i_{23}} \cdot \frac{r_{i_{33}} \cdot \boldsymbol{r2}_i \cdot \boldsymbol{M}_{xy}}{\boldsymbol{r1}_i \cdot \boldsymbol{r1}_i^{\mathrm{T}}}$, which are both $1 \times n$ matrices. Then, we have

$$s_i = \frac{\mathrm{tr}\left[\boldsymbol{q}_i \cdot \begin{bmatrix} \boldsymbol{a}_i \\ \boldsymbol{b}_i \end{bmatrix}^{\mathrm{T}}\right]}{\boldsymbol{a}_i \cdot \boldsymbol{a}_i^{\mathrm{T}} + \boldsymbol{b}_i \cdot \boldsymbol{b}_i^{\mathrm{T}}}, \tag{7}$$

where tr[ ] denotes the trace, which is the sum of the diagonal elements in a matrix.

From (6), there are N different z-coordinates for the face model. To find the optimal depths of the feature points in the face model, the z-coordinates in $\boldsymbol{M}$ are calculated by applying partial differentiation to (4) rather than (5), with respect to $\boldsymbol{Z_M}$:

$$\boldsymbol{Z}_{\boldsymbol{M}}^{\mathrm{T}} = \frac{\sum_{i=1}^{n}(s_i \cdot \boldsymbol{r1}_i \cdot \boldsymbol{q}_i + s_i^2 \cdot r_{i_{33}} \cdot \boldsymbol{r2}_i \cdot \boldsymbol{M}_{xy})}{\sum_{i=1}^{n} s_i^2 \cdot \boldsymbol{r1}_i \cdot \boldsymbol{r1}_i^{\mathrm{T}}}, \tag{8}$$

where the respective s_i are calculated using (7). Then this set of new z-coordinates replaces the original one. The proof of (6)–(8) has been included in Appendix 1.

To calculate the fitness of a chromosome, we first substitute its values, i.e. ϕ_i, ψ_i and θ_i, to (2) in order to calculate $\boldsymbol{R}_i$. Then, the corresponding scaling factor s_i is computed using (6) and (7), and the depths of the feature points in the adapted face model are calculated using (8). Finally, its fitness can be calculated by substituting all of the above parameters into (4), which consider its fitness to all the face images.

Chromosome: pose parameters

θ_1	ψ_1	ϕ_1	θ_2	ψ_2	ϕ_2	…	θ_N	ψ_N	ϕ_N

Fig. 4. Structure of a chromosome with N non-frontal-view face images.

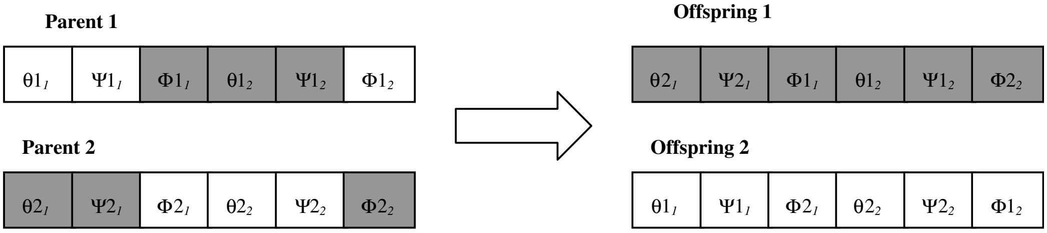

Fig. 5. An example of the crossover operation.

2.2.3. *The genetic operators*

Having defined the chromosome and the fitness function, the genetic operators – selection, crossover, and mutation (Goldberg, 1989) – which are performed to search the optimal poses of the face images and the optimal depths of the face model are described in this section. In our algorithm, the rank selection method is used to select two chromosomes to perform crossover and/or mutation. After selecting two chromosomes, two crossover points are selected randomly. The values between these two crossover points in the two chromosomes are exchanged to form a pair of new offspring. Fig. 5 illustrates the crossover operation.

Mutation is intended to prevent all the solutions in a population falling into a local minimum by exploiting new candidates randomly. In our algorithm, the number of elements in a chromosome being mutated depends on the number of face images. The N elements in each chromosome are randomly selected and replaced by N randomly generated numbers, where N is the number of non-frontal-view face images.

3. The similarity measure

After constructing the face model of a person, it can be adapted to any face image. If the face model is constructed from a particular subject, the feature-point distance between this face model and a face image of this particular subject should be smaller than that of another subject. Unlike other face model construction algorithms (Su et al., 2002; Ansari and Abdel-Mottaleb, 2005) our algorithm can evaluate the accuracy of the constructed 3D face model. The accuracy of the 3D face model can be determined by measuring the feature-point distance between the face model and the respective face images. This is especially useful since we do not usually have the exact data of the 3D face structure. This distance can also be applied to human face recognition.

To compute the feature-point distance between the 3D face model and a 2D face image, the Levenberg–Marquardt method (Levenberg, 1944; Marquardt, 1963) is used to optimize the following equation:

$$\boldsymbol{D}^2 = \min_{s,\boldsymbol{R}_{2\times3}} \frac{1}{n}\|\boldsymbol{u} - s\boldsymbol{R}_{2\times3}\boldsymbol{M}\|^2, \tag{9}$$

where $\boldsymbol{u}$ is the $2 \times n$ matrix representing the (x, y) coordinates of the feature points in a test face image, and $\boldsymbol{R}_{2\times3}$ and s are the rotation matrix and scaling factor, respectively, that can minimize the above equation. $\boldsymbol{R}_{2\times3}$ contains the first two rows of the 3×3 rotation matrix $\boldsymbol{R}$, which can be specified as the three successive rotations around the x-, y-, and z-axes, by an angle of ϕ, ψ and θ, respectively. This matrix can be written as the product of these three rotations by using (2). Having constructed the 3D face model of a face subject, the depths of the feature points are known. Hence, when a 2D face image is compared to the 3D face model for face recognition, a simpler optimization method, the Levenberg–Marquardt method instead of the GA, can be used to estimate the pose and scale of the query image.

For face recognition, the query or test face image can be compared to different face models using (9). The face model that results in the minimum feature-point distances should have the best representation of the query face image. However, not all the n feature points in the query face images are visible, because the query image may have an arbitrary pose. As a result, some modifications have to be made to (9). First, the columns of $\boldsymbol{M}$ corresponding to the invisible feature points are removed. Then, n is replaced by the number of visible feature points in the face image. Experiments in the next section will show the validate use of the 3D face model for face recognition.

4. Experimental results

A subset of the FERET database (Phillips et al., 2000) is selected for our experiments. This is a standard database for face recognition evaluation, which contains images in various poses. To construct different face models, 60 frontal face images, corresponding to 60 distinct subjects, were selected in our experiment: 13 of the subjects have 4 non-frontal-view face images, 12 have 3 non-frontal-view face images, and the remaining 35 have 2 non-frontal-view face images only. All the 15 feature points are visible in the selected non-frontal-view face images. However, in the face recognition experiment, face images with larger pose variations can be selected because not all the 15 feature points are required for the matching of a test image and the face model. In addition, the 15 feature points were selected manually in our experiments so that potential errors in the detection of the facial feature points can be eliminated.

4.1. *3D face model construction*

In this experiment, different numbers of face images of the same subject under different poses may be used to

Table 1
The parameters of the GA under different numbers of face images

Number of face images	Population size	Maximum iterations	Maximum runtime per model (s)
3	800	200	1.8
4	1200	300	2.6
5	1500	400	4.0

Table 2
Indices of the face models for (a) Example 1 and (b) Example 2

	Face model indices
(a) Example 1	
Model 1	Images 1, 2, 3
Model 2	Images 1, 2, 4
Model 3	Images 1, 2, 5
Model 4	Images 1, 3, 4
Model 5	Images 1, 3, 5
Model 6	Images 1, 4, 5
Model 7	Images 1, 2, 3, 4
Model 8	Images 1, 2, 3, 5
Model 9	Images 1, 2, 4, 5
Model 10	Images 1, 3, 4, 5
Model 11	Images 1, 2, 3, 4, 5
(b) Example 2	
Model 1	Images 1, 2, 3
Model 2	Images 1, 2, 4
Model 3	Images 1, 3, 4
Model 4	Images 1, 2, 3, 4

construct a face model. For example, suppose that one frontal face image and four images under different poses are available. Then, six different face models can be constructed when two of the different non-frontal-view face images and one frontal-view face image are considered in the construction. Similarly, there are four different models when three of the non-frontal-view images are used and only one when all the four non-frontal-view images are considered. Therefore, 11 different face models can be constructed. To construct the face models, the best poses of the face models were aligned using the GA. The respective ranges of the elements in the chromosomes, i.e. ϕ_i, ψ_i and θ_i, were set between −50° and 50°; this allows all 15 feature points to be visible after 2D projection.

Table 1 shows the population size and the maximum number of iterations for face model construction using different numbers of images under different poses. The maximum runtime required to generate a face model is about 1.8 s using 3 face images under different poses. This runtime is measured with a Pentium IV computer system with 2.3 GHz and 512 MB RAM. The crossover rate and the mutation rate were set at 80% and 20%, respectively.

Figs. 6 and 7 show the face images of two different subjects that were used to construct their face models. The leftmost image is the reference image, i.e. the frontal-view image, while the other images are under different poses. In Fig. 6, five images are available, so at most 11 different face models can be constructed; while in Fig. 7, four images are available, so at most 4 different face models can be constructed. To illustrate the estimation of the poses using

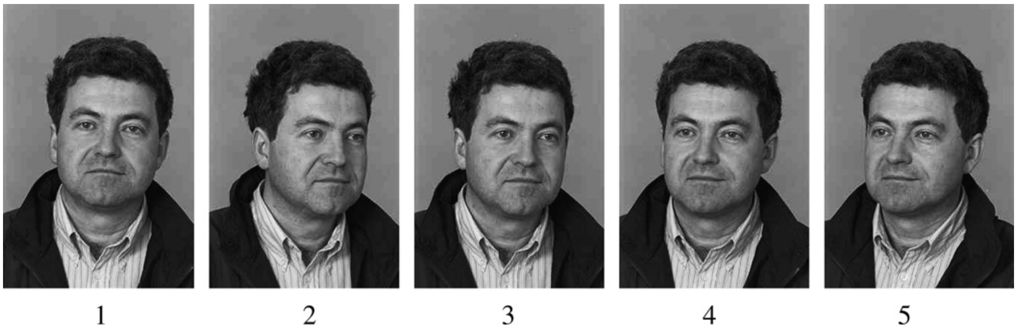

Fig. 6. Example 1 – five face images under different poses used to construct the face model.

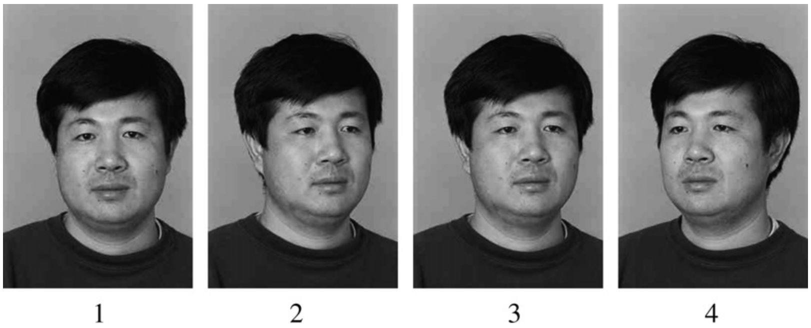

Fig. 7. Example 2 – four face images under different poses used to construct the face model.

different combinations of the front-view image and non-frontal-view images from Example 1 and Example 2, Table 2 tabulates the corresponding indices of the face models determined based on the different combinations of the face images.

Tables 3 and 4 tabulate the best poses of the adapted face models to the respective non-frontal-view face images from Example 1 and Example 2, respectively. The entries in these tables show the angles of the non-frontal-view face images about the x-, y- and z-axes. It can be observed that the estimated poses of the different models for the same face image are consistent. Fig. 8 shows the variance of the poses for 13 distinct subjects. The face images rotated around the y-axis result in the largest variance as the faces are mainly rotated around this axis.

Tables 5 and 6 tabulate the structure of the face models from Example 1 and Example 2, respectively, which are constructed using all the available face images. Each of the columns in these tables shows the (x, y, z) coordinates of each corresponding feature point, which have been defined as shown in Fig. 3a. Figs. 9 and 10 show the means and the standard deviations of the depths of the feature points from different face models in Example 1 and Example 2, respectively. The number of pixels is used to represent the depths of the face models because the models are derived from face images, which use pixels as the unit.

From Figs. 9 and 10, we see that feature point 7 has the largest z-coordinate value, as this point represents the nose tip, which is the outermost part in a face. In addition, feature points 8 and 9 have larger z-coordinate values than the other feature points because these two points represent the lips, which protrude more than all other feature points except the nose tip. Feature points 1–6 have very similar z-coordinate values to feature points 10–15, as the structure of a face is usually quite symmetrical. These show that the structure of the constructed face models conforms to the structure of the human faces. Figs. 11 and 12 show the models adapted to the non-frontal-view face images in Example 1 and Example 2 after the optimal poses of the face images and their 3D face models have been determined.

4.2. Evaluation of the accuracy of the face models

To evaluate the accuracy of the constructed 3D face models, the similarity transform described in Section 3 was used. With the various face models generated using Example 1 and Example 2, the smallest distances between a number of face images and the respective face models are measured, as shown in Figs. 13 and 14. Some of the test images are of the same person as was used for the face model, while the others are of other subjects. The similarity distances between the face images and the face models of the same subject are small when compared to those between the face images and the face models of different subjects. In addition, the similarity distances between the face images and the face models of the same subject are similar irrespective of the images used to construct the face

Table 3
The best estimated poses of the non-frontal-view images from Example 1

	Poses of image 2			Poses of image 3			Poses of image 4			Poses of image 5		
	Tilt (X degs)	Pan (Y degs)	Roll (Z degs)	Tilt (X degs)	Pan (Y degs)	Roll (Z degs)	Tilt (X degs)	Pan (Y degs)	Roll (Z degs)	Tilt (X degs)	Pan (Y degs)	Roll (Z degs)
Model 1	0	−16	1	1	−10	5						
Model 2	0	−22	3				2	11	8			
Model 3	0	−24	3							2	24	5
Model 4				1	−14	8	0	16	8			
Model 5				1	−15	7				1	31	9
Model 6							2	13	8	2	25	6
Model 7	0	−22	3	1	−14	8	2	11	9			
Model 8	0	−24	3	2	−16	10				2	25	6
Model 9	0	−23	4				2	11	7	2	24	6
Model 10				3	−14	8	3	9	4	3	24	5
Model 11	0	−23	3	2	−13	8	2	13	8	2	25	6

Table 4
The best estimated poses of the non-frontal-view images from Example 2

	Poses of image 2			Poses of image 3			Poses of image 4		
	Tilt (X degs)	Pan (Y degs)	Roll (Z degs)	Tilt (X degs)	Pan (Y degs)	Roll (Z degs)	Tilt (X degs)	Pan (Y degs)	Roll (Z degs)
Model 1	−1	−20	−5	−1	−15	−6			
Model 2	−1	−18	−5				−1	16	−4
Model 3				−1	−14	−6	−1	15	−4
Model 4	−1	−19	−4	−1	−15	−5	−1	15	−4

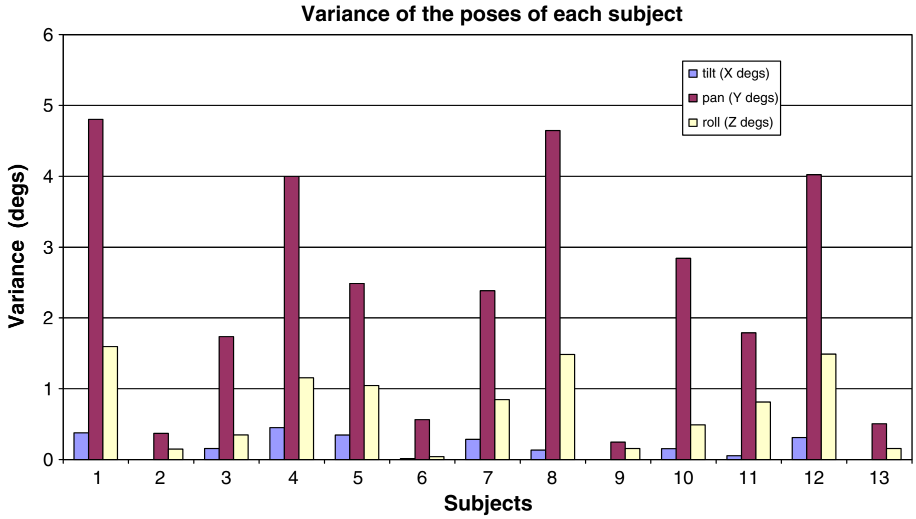

Fig. 8. Variance of the poses for 13 distinct subjects.

Table 5
The structure of the face model constructed from five images in Example 1

Example 1	Indices of the feature points (pixels)														
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
x	98	125	111	112	112	121	140	137	137	190	163	163	177	174	157
y	164	165	227	159	168	202	197	223	230	171	170	229	164	175	203
z	0	7	14	6	10	21	45	29	33	3	4	15	6	6	24

Table 6
The structure of the face model constructed from four images in Example 2

Example 2	Indices of the feature points (pixels)														
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
x	98	124	120	111	113	126	145	144	144	186	162	168	175	174	162
y	180	178	249	174	184	223	212	241	259	176	177	248	171	181	223
z	0	3	2	9	5	12	31	24	25	7	8	8	8	9	21

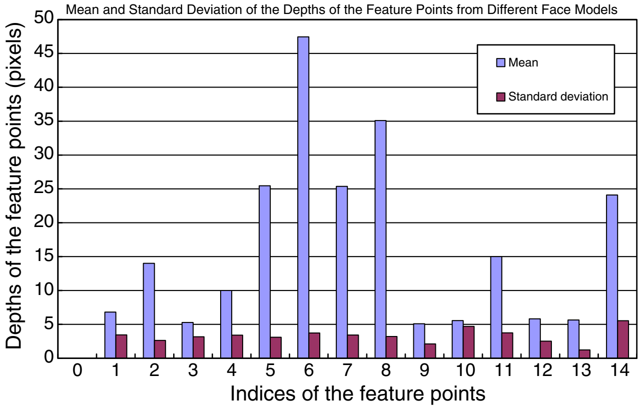

Fig. 9. The mean and standard deviation of the depths of the feature points from different face models in Example 1.

models. Therefore, this method can also be used as a face recognition algorithm, which can alleviate the effect of perspective variations.

4.3. Face recognition using the 3D face models

After the face models for each subject have been constructed, the feature-point distance can also be used for face recognition. In this experiment, each face model was constructed using 3 different face images under different poses (one of which is frontal-view). Each distinct subject is represented by a corresponding 3D face model, so 180 face images were used to construct 60 distinct face models.

To perform face recognition, other face images which have not been used to construct the face models are used as testing face images, and are compared to the different face models using the similarity transform. The face images used to construct the face models are the training images for PCA and LDA, while other face images are used as testing images. If the similarity distance between a face model and a testing face image is a minimum, this face image will then be classified as the subject of the face model. Therefore, for each testing face image, its similarity

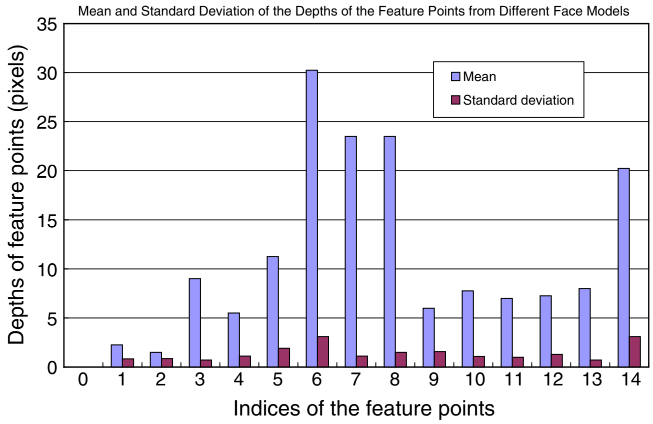

Fig. 10. The mean and standard deviation of the depths of the feature points from different face models in Example 2.

distances to all the face models are computed, and the faces are listed in ascending order according to these distances. As described in Section 3, not all the feature points are needed to calculate the feature-point distance. Consequently, face images with large pose variations can be recognized, even though not all the feature points are visible in these face images.

In this experiment, 72 testing face images of 28 different subjects were selected. All these subjects have their own face models stored in the face model database. These images are divided into two sets. The first set includes those face images under large pose variations in which the absolute angle rotated around the y-axis is larger than 50°. The second set contains those face images under small pose variations in which the absolute angle rotated around the y-axis is smaller than 50°. There are 45 testing face images in the first set, and the remaining face images are in the other set.

Fig. 15 shows the recognition rates when the correct face models of the testing images are in the top k of the list according to the similarity distances, where $k = 1, \ldots, 10$. Our method is also compared to two other face recognition techniques: PCA and LDA. These two methods can achieve better performances when the top 3 in the list are considered. Nevertheless, the recognition rate of our method is about 80%, whereas PCA and LDA have a similar recognition rate of about 60% when the top 10 on the list are considered. Figs. 16 and 17 show the face recognition rates of the testing images under small and large pose variations, respectively. Fig. 16 shows that PCA and LDA outperform our algorithm up to the first nine most similar faces. The reason for this is that the images are those face images under small pose variations. Our algorithm has a similar recognition performance when the top 10 on the list are considered. Fig. 17 shows that the recognition rates using PCA and LDA are lower than with our algorithm. These results show that our algorithm outperforms PCA and LDA when the testing face images have large pose variations. This face recognition algorithm is based on the facial-feature points only, which is not sufficient to achieve a high recognition rate. However, for a large face database, our algorithm can be used to select a subset of face images from the database for further analysis. The problem due to pose variation can be alleviated, and the computational processing time required for comparing the feature points is much lower than with other advanced face recognition techniques.

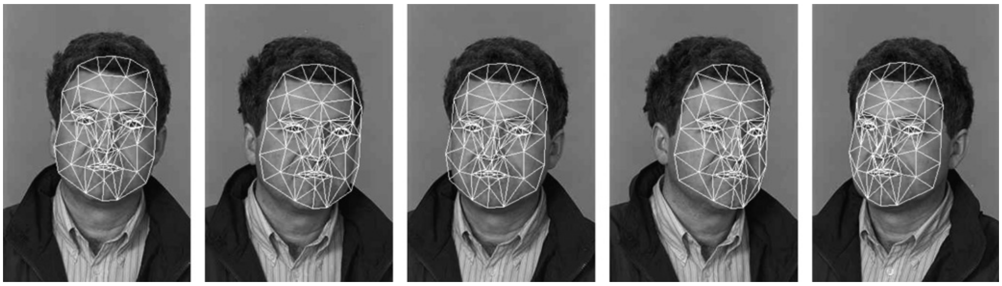

Fig. 11. Adaptation of face model to the non-frontal-view face images of Example 1.

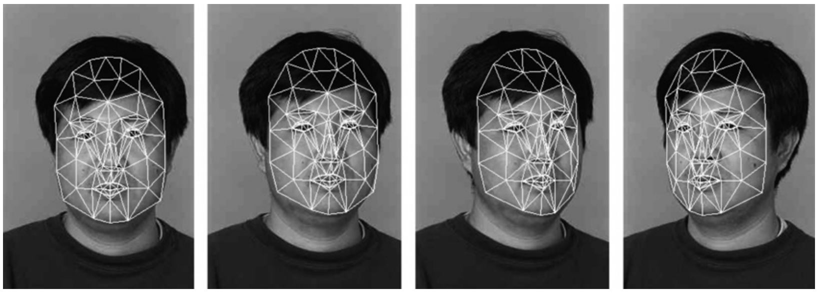

Fig. 12. Adaptation of face model to the non-frontal-view face images of Example 2.

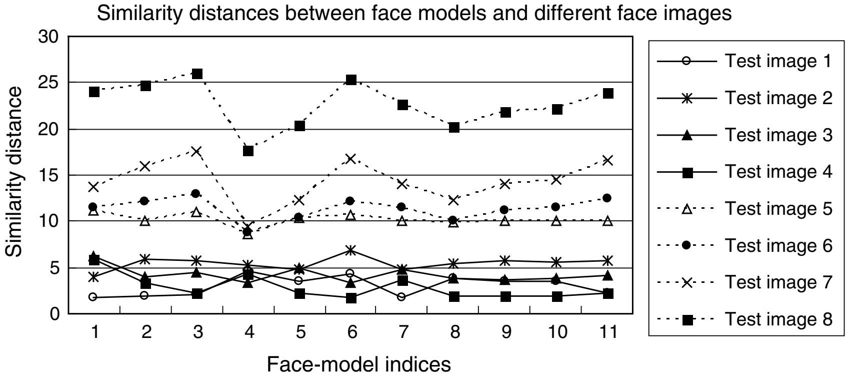

Fig. 13. The similarity distances between a number of test images and each of the face models generated from Example 1. Test images 1–4 are the same subject as the face model, while the others are different subjects to the face model.

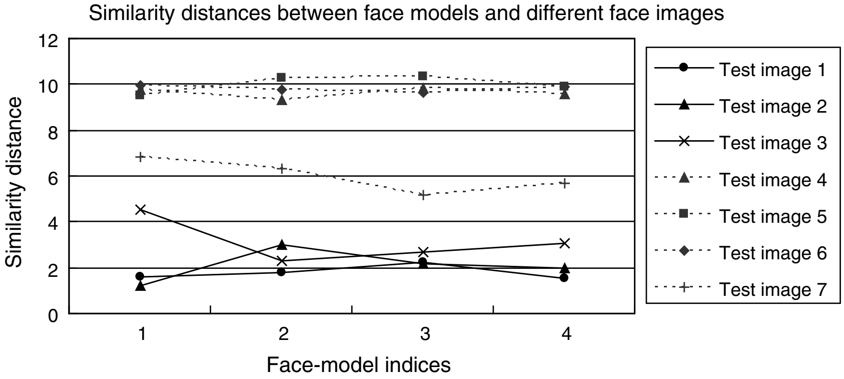

Fig. 14. The similarity distance between a number of images and each of the face models generated from Example 2. Test images 1–3 are the same subject as the face model, while the others are different subjects to the face model.

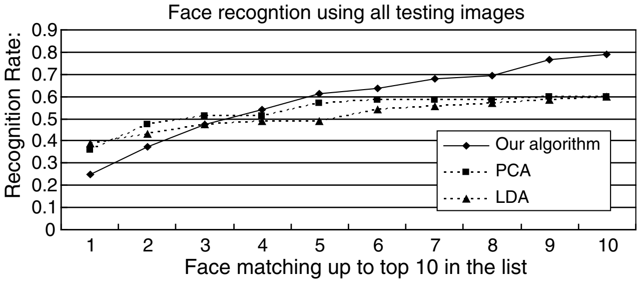

Fig. 15. The face recognition rates of different face recognition techniques using all testing images.

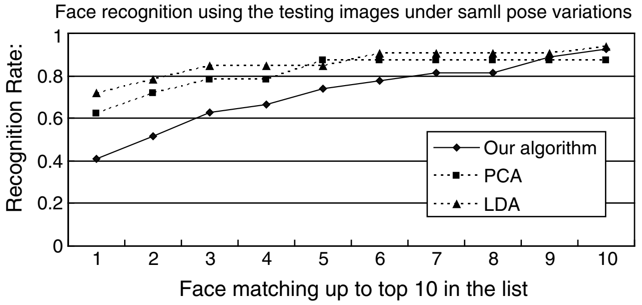

Fig. 16. The face recognition rates of different face recognition techniques using the testing images under large pose variations.

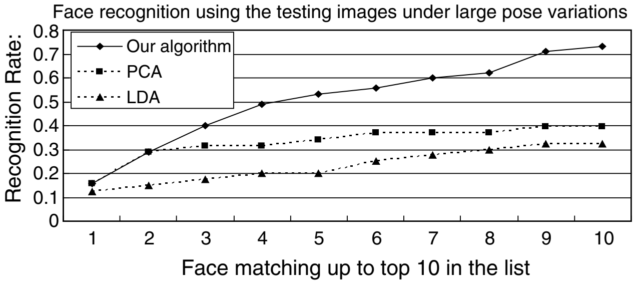

Fig. 17. The face recognition rates of different face recognition techniques using the testing images under small pose variations.

5. Conclusion

In this paper, a 3D face reconstruction method is proposed to estimate the depth information about a human face based on face images under different poses. Our method does not require any camera calibration. In order to estimate the poses and the depths of the face model efficiently, the genetic algorithm is applied to minimize the similarity distance between the adapted face model and the faces under different poses.

Since the 3D information about human faces is not available in most applications, a measurement to assess the accuracy of the constructed face model has been proposed, which is based on the similarity transform and the Levenberg–Marquardt method to find the optimal solution. With our proposed algorithm, both the poses of the face images and their 3D structure can be determined. In addition, experiments have shown that the estimation of the poses is consistent, and that the estimated 3D face models can be used for face recognition.

Acknowledgement

The work was supported by the Centre for Signal Processing, The Hong Kong Polytechnic University.

Appendix 1. Proof of the similarity transform

Assume that there are n points in two different point sets, and $(M_{x_i}, M_{y_i}, M_{z_i})$ are the 3D coordinates of the ith feature point in the adapted face model in which all the feature points have been centered. Similarly, (q_{x_i}, q_{y_i}) are the 2D coordinates of the ith feature point in the non-frontal-view face image in which the feature points have also been centered. Then, the similarity distance of the ith feature point between the face model and the non-frontal-view face image is

$$D^2 = \|q_{x_i} - s(r_{11}M_{x_i} + r_{12}M_{y_i} + r_{13}M_{z_i})\|^2 + \|q_{y_i} - s(r_{21}M_{x_i} + r_{22}M_{y_i} + r_{23}M_{z_i})\|^2. \tag{A1.1}$$

By applying partial differentiation to (A1.1) with respect to M_{z_i}

$$\frac{\partial D^2}{\partial M_{z_i}} = [q_{x_i} - s(r_{11}M_{x_i} + r_{12}M_{y_i} + r_{13}M_{z_i})](-sr_{13}) + [q_{y_i} - s(r_{21}M_{x_i} + r_{22}M_{y_i} + r_{23}M_{z_i})](-sr_{23}) \quad \text{i.e.} \ = 0,$$

$$M_{z_i} = \frac{r_{13}q_{x_i} + r_{23}q_{y_i} - s(r_{11}M_{x_i} + r_{12}M_{y_i})r_{13} - s(r_{21}M_{x_i} + r_{22}M_{y_i})r_{23}}{s(r_{13}^2 + r_{23}^2)}. \tag{A1.2}$$

Since

$$\begin{aligned} r_{11}r_{13} + r_{21}r_{23} + r_{31}r_{33} &= 0, \\ r_{12}r_{13} + r_{22}r_{23} + r_{32}r_{33} &= 0. \end{aligned} \tag{A1.3}$$

(A1.2) can be rewritten as follows:

$$M_{z_i} = \frac{r_{13}q_{x_i} + r_{23}q_{y_i} + sr_{33}(r_{31}M_{x_i} + r_{32}M_{y_i})}{s(r_{13}^2 + r_{23}^2)}. \tag{A1.4}$$

For simplicity's sake, we rewrite (A1.4) as follows:

$$M_{z_i} = \frac{m}{s} + n, \tag{A1.5}$$

where $m = \frac{r_{13}q_{x_i} + r_{23}q_{y_i}}{s(r_{13}^2 + r_{23}^2)}$ and $n = \frac{r_{33}(r_{31}M_{x_i} + r_{32}M_{y_i})}{r_{13}^2 + r_{23}^2}$.

Let $\boldsymbol{r1} = [r_{13} \quad r_{23}], \boldsymbol{r2} = [r_{31} \quad r_{32}], \boldsymbol{r3} = [r_{11} \quad r_{12}]$ and $\boldsymbol{r4} = [r_{21} \quad r_{22}]$, and $\boldsymbol{X_M}$, $\boldsymbol{Y_M}$ and $\boldsymbol{Z_M}$ are the three $n \times 1$ matrices, which represent the x-, y- and z-coordinates of the centered feature points in the adapted face model. Let $\boldsymbol{M}_{xy} = [\boldsymbol{X_M}, \boldsymbol{Y_M}]^{\mathrm{T}}$, and $\boldsymbol{q}$ be the $2 \times n$ matrix, which represents the centered image point set. Then, rewrite (A1.4) into matrix form as follows:

$$\boldsymbol{Z}_{\boldsymbol{M}}^{\mathrm{T}} = \frac{\boldsymbol{r1} \cdot \boldsymbol{q} + s \cdot r_{33} \cdot \boldsymbol{r2} \cdot \boldsymbol{M}_{xy}}{s \cdot \boldsymbol{r1} \cdot \boldsymbol{r1}^{\mathrm{T}}}. \tag{A1.6}$$

Substituting (A1.5) into (A1.1), we have

$$D^2 = \|q_{x_i} - r_{13}m - s(r_{11}M_{x_i} + r_{12}M_{y_i} + r_{13}n)\|^2 + \|q_{y_i} - r_{23}m - s(r_{21}M_{x_i} + r_{22}M_{y_i} + r_{23}n)\|^2 \tag{A1.7}$$

and then differentiating (A1.1) with respect to s

$$\begin{aligned} \frac{\partial D^2}{\partial s} &= [q_{x_i} - r_{13}m - s(r_{11}M_{x_i} + r_{12}M_{y_i} + r_{13}n)](r_{11}M_{x_i} \\ &\quad + r_{12}M_{y_i} + r_{13}n) + [q_{y_i} - r_{23}m - s(r_{21}M_{x_i} + r_{22}M_{y_i} \\ &\quad + r_{23}n)](r_{21}M_{x_i} + r_{22}M_{y_i} + r_{23}n) \\ &= 0, \end{aligned}$$

i.e.

$$s = \frac{(q_{x_i} - r_{13}m)(r_{11}M_{x_i} + r_{12}M_{y_i} + r_{13}n) + (q_{y_i} - r_{23}m)(r_{21}M_{x_i} + r_{22}M_{y_i} + r_{23}n)}{(r_{11}M_{x_i} + r_{12}M_{y_i} + r_{13}n) + (r_{21}M_{x_i} + r_{22}M_{y_i} + r_{23}n)}. \tag{A1.8}$$

We can also write s in matrix form, let

$$\boldsymbol{a}_i = \boldsymbol{r3}_i \cdot \boldsymbol{M}_{xy} + r_{i_{13}} \cdot \frac{r_{i_{33}} \cdot \boldsymbol{r2}_i \cdot \boldsymbol{M}_{xy}}{\boldsymbol{r1}_i \cdot \boldsymbol{r1}_i^{\mathrm{T}}} \quad \text{and}$$

$$\boldsymbol{b}_i = \boldsymbol{r4}_i \cdot \boldsymbol{M}_{xy} + r_{i_{23}} \cdot \frac{r_{i_{33}} \cdot \boldsymbol{r2}_i \cdot \boldsymbol{M}_{xy}}{\boldsymbol{r1}_i \cdot \boldsymbol{r1}_i^{\mathrm{T}}},$$

which are $1 \times n$ matrices, then

$$s_i = \frac{\mathrm{tr}\left[\boldsymbol{q}_i \cdot \begin{bmatrix} \boldsymbol{a}_i \\ \boldsymbol{b}_i \end{bmatrix}^{\mathrm{T}}\right]}{\boldsymbol{a}_i \cdot \boldsymbol{a}_i^{\mathrm{T}} + \boldsymbol{b}_i \cdot \boldsymbol{b}_i^{\mathrm{T}}}, \tag{A1.9}$$

where tr[] of a matrix is the sum of its diagonal elements.

Appendix 2. The Levenberg–Marquardt method

The Levenberg–Marquardt (LM) method searches the parameters $\boldsymbol{x}$, which will minimize (9), where $\boldsymbol{x} = [\theta \quad \varphi \quad \phi \quad s]^{\mathrm{T}}$. Let

$$f(\boldsymbol{x}) = \begin{bmatrix} q_{x_1} - s(r_{11}M_{x_1} + r_{12}M_{y_1} + r_{13}M_{z_1}) \\ \vdots \\ q_{x_n} - s(r_{11}M_{x_n} + r_{12}M_{y_n} + r_{13}M_{z_n}) \\ q_{y_1} - s(r_{21}M_{x_1} + r_{22}M_{y_1} + r_{23}M_{z_1}) \\ \vdots \\ q_{y_n} - s(r_{21}M_{x_n} + r_{22}M_{y_n} + r_{23}M_{z_n}) \end{bmatrix} \tag{A2.1}$$

and (9) can be rewritten as follows:

$$F(\boldsymbol{x}) = \frac{1}{n} f(\boldsymbol{x})^{\mathrm{T}} f(\boldsymbol{x}). \tag{A2.2}$$

We will compute $\boldsymbol{x}$ such that

$$\boldsymbol{x}^* = \arg\min_{x}\{F(\boldsymbol{x})\}, \tag{A2.3}$$

i.e. to minimize (A2.2).

To find the solution of (A2.3), Levenberg and Marquardt suggested using a damped Gauss–Newton method. Assume $\boldsymbol{J}$ is the Jacobian of $f(\boldsymbol{x})$, which is a matrix containing the first partial derivatives of $f(\boldsymbol{x})$, i.e.

$$(\boldsymbol{J}(\boldsymbol{x}))_{ij} = \frac{\partial f_i}{\partial x_j}(\boldsymbol{x}).$$

Then, solve

$$(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{J} + \mu\boldsymbol{I})\boldsymbol{h}_{\mathrm{lm}} = -\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}, \tag{A2.4}$$

where $\boldsymbol{J} = J(\boldsymbol{x})$ and $\boldsymbol{f} = f(\boldsymbol{x})$, μ is the damping parameter and $\boldsymbol{h}_{\mathrm{lm}}$ is a descent direction.

The steps in the Levenberg–Marquardt method are shown as follows:

1. Initialize the damping parameter μ related to the size of the elements $A_0 = \boldsymbol{J}(\boldsymbol{x}_0)^{\mathrm{T}}\boldsymbol{J}(\boldsymbol{x}_0)$. For example, let $\mu_0 = 10^{-3} \cdot \max_i\{a_{ii}^{(0)}\}$.
2. Solve (A2.4) to find $\boldsymbol{h}_{\mathrm{lm}}$.
3. $$\boldsymbol{x}_{\mathrm{new}} = \boldsymbol{x} + \boldsymbol{h}_{\mathrm{lm}}. \tag{A2.5}$$
4. Substitute (A2.5) back to (A2.4).
5. During iteration, the size of μ is controlled by the gain ratio
$$\varsigma = \frac{F(\boldsymbol{x}) - F(\boldsymbol{x} + \boldsymbol{h}_{\mathrm{lm}})}{\frac{1}{2}\boldsymbol{h}_{\mathrm{lm}}^{\mathrm{T}}(\mu\boldsymbol{h}_{\mathrm{lm}} - \boldsymbol{J}^{\mathrm{T}}\boldsymbol{f})}. \tag{A2.6}$$
6. The stopping criteria indicate that, at a global minimizer, $F(\boldsymbol{x}^*) = g(\boldsymbol{x}^*) = 0$, so $\|\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}\|_\infty \leqslant \varepsilon_1$. Another relevant stopping criterion is that the change in $\boldsymbol{x}$ is small, i.e. $\|\boldsymbol{x}_{\mathrm{new}} - \boldsymbol{x}\| \leqslant \varepsilon_2(\|\boldsymbol{x}\| + \varepsilon_2)$. In our algorithm, ε_1 and ε_2 are both set at 10^{-8}.

References

Ahlberg, J., 2001. CANDIDE-3 – Updated Parameterised Face. Linkoping University, Lysator LiTH-ISY-R-2325.

Ahlberg, J., Forchheimer, R., 2003. Face tracking for model-based coding and face animation. Internat. J. Imaging Systems Technol. 13 (1), 8–22.

Ansari, A.N., Abdel-Mottaleb, M., 2005. Automatic facial feature extraction and 3D face modeling using two orthogonal views with application to 3D face recognition. Pattern Recognition 38 (12), 2549–2563.

Chellappa, R., Wilson, C.L., Sirohey, S., 1995. Human and machine recognition of faces: A survey. Proc. IEEE 83 (5), 705–741.

Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading, MA.

He, X.F., Yan, S.C., Hu, Y.X., Niyogi, P., Zhang, H.J., 2005. Face recognition using Laplacian faces. IEEE Trans. Pattern Anal. Machine Intell. 27 (3), 328–340.

Hu, X., Ahuja, N., 1991. Motion estimation under orthographic projection. IEEE Trans. Rob. Autom. 7 (6), 848–853.

Huang, T.S., Lee, C.H., 1989. Motion and structure from orthographic projections. IEEE Trans. Pattern Anal. Machine Intell. 11 (5), 536–540.

Huang, T.S., Netravali, A.N., 1994. Motion and structure from feature correspondences: A review. Proc. IEEE 82 (2), 252–268.

Jerian, C.P., Jain, R., 1991. Structure from motion – A critical analysis of methods. IEEE Trans. Systems Man Cybernet. 21 (3), 572–588.

Jiang, D.L., Hu, Y.X., Yan, S.C., Zhang, L., Zhang, H.J., Gao, W., 2005. Efficient 3D reconstruction for face recognition. Pattern Recognition 38 (6), 787–798.

Lee, M.W., Ranganath, S., 2003. Pose-invariant face recognition using a 3D deformable model. Pattern Recognition 36 (8), 1835–1846.

Levenberg, K., 1944. A method for the solution of certain problems in least squares. Quart. Appl. Math. 2, 164–168.

Marquardt, D., 1963. An algorithm for least squares estimation on nonlinear parameters. SIAM J. Appl. Math. 11 (2), 431–441.

Morita, T., Kanade, T., 1997. A sequential factorization method for recovering shape and motion from image streams. IEEE Trans. Pattern Anal. Machine Intell. 19 (8), 858–867.

Phillips, P.J., Hyeonjoon, M., Rizvi, S.A., Rauss, P.J., 2000. The FERET evaluation methodology for face-recognition algorithms. IEEE Trans. Pattern Anal. Machine Intell. 22 (10), 1090–1104.

Shapiro, L.S., Zisserman, A., Brady, M., 1995. 3D motion recovery via affine epipolar geometry. Internat. J. Comput. Vision 16 (2), 147–182.

Su, M.S., Chen, C.Y., Cheng, K.Y., 2002. An automatic construction of a person's face model from the person's two orthogonal views. In: Proc. Geometric Modeling and Processing, pp. 179–186.

Tomasi, C., Kanade, T., 1992. Shape and motion from image streams under orthography: A factorization Method. Internat. J. Comput. Vision 9 (2), 137–154.

Ullman, S., 1979. The Interpretation of Visual Motion. MIT Press, Cambridge, Mass.

Werman, M., Weinshall, D., 1995. Similarity and affine invariant distances between 2D point sets. IEEE Trans. Pattern Anal. Machine Intell. 17 (8), 810–814.

Xirouhakis, Y., Delopoulos, A., 2000. Least squares estimation of 3D shape and motion of rigid objects from their orthographic projections. IEEE Trans Pattern Anal. Machine Intell. 22 (4), 393–399.