Facial modeling from an uncalibrated face image using a coarse-to-fine genetic algorithm

Shinn-Ying Ho*, Hui-Ling Huang

Department of Information Engineering, Feng Chia University, 100 Wen-hwa Road, Taichung, Taiwan 407, ROC

Received 2 August 1999; received in revised form 18 February 2000; accepted 18 February 2000

Abstract

This paper presents a genetic algorithm-based optimization approach for facial modeling from an uncalibrated face image using a flexible generic parameterized facial model (FGPFM). The FGPFM can be easily modified using the facial features as parameters of FGPFM to construct an accurate specific 3D facial model from only a photograph of an individual with a yawed face based on the projection transformation. The facial modeling problem is formulated as a parameter optimization problem and the objective function is also given. Moreover, a coarse-to-fine approach based on our intelligent genetic algorithm which can efficiently solve the large parameter optimization problems is used to accelerate the search for an optimal solution. Furthermore, sensitivity analysis and experimental results with texture mapping demonstrate the effectiveness of the proposed method. © 2001 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Facial modeling; Genetic algorithm; Generic facial model; Pose determination; Optimization; Computer vision

1. Introduction

Face images have received considerable attention, particularly in the fields of computer vision and signal processing communities. For instance, model-based image coding methods have been proposed for future videophone and video conference services. However, the images in these applications are complex and highly variable, even for a specific individual. An important problem is how to create a 3D model of a specific individual. Automatic creation of a 3D facial model of a specific individual plays an important role in many applications, such as model-based coding for narrow-band visual communication [1–4], view-independent face recognition tasks [5,6], and image synthesis problems in areas like virtualized reality [7] and synthesis of novel views [8,9].

3D facial models can be categorized into two classes: those based on the view-independent 3D facial structure and those considering only view-dependent facial models. The view-dependent facial model uses multiview representation in which a set of 2D image-based example facial models are combined into a flexible 3D facial model by a weighted sum of given example facial models [9,10]. The limitations of view-dependence narrow the scope of 3D facial model-related applications. The approaches used to automatically create 3D facial models with a view-independent 3D facial structure can be applied more extensively [3]. Approaches capable of creating a view-independent 3D facial model of a specific individual can be categorized into two groups: use of an actual 3D face of a specific individual and use of a generic facial model with 2D face images of an individual. The approaches that need an actual 3D face include active vision [2], 3D digitizer [8], and vision-based methods [11].

*This work was supported by the National Science Council, Taiwan, ROC, Grant No. NSC88-2213-E-035-012.

*Corresponding author. Tel.: + 886-4-4517250; fax: + 886-4-4516101.

E-mail address: syho@fcu.edu.tw (S.-Y. Ho).
Approaches belonging to the second group in which the 3D facial model of a specific individual is constructed consist of two steps. Firstly, select a generic model representing the topological structure of a typical face and typical 2D face images of the individual. Secondly, adjust the geometrical shape of the generic model to that of the actual face images through 3D transformation and modification according to the positions of some facial features such as eyes, mouth, nose, and facial contour. Akimoto et al. [1], Tang and Huang [12], and Ip and Yin [7] used two orthogonal face images of an individual’s head to acquire the features deemed necessary for fitting a generic model to an individual’s head. Pei et al. [4] used the transformation of the generic facial model with an affine mapping for model-based image coding by tracking 3D contour feature points. Aizawa et al. [14] used multiple face images to adjust a flexible three-dimensional facial model for a particular face. Eisert and Girod [15] changed the texture and control points’ position using information from 3D laser scans to adjust the generic 3D model to a specific individual.

Given only a photograph of an individual’s face shown in front-viewed or as a randomly yawed face when viewing the face from close range, can one automatically create an accurate 3D facial model of the individual? For this purpose, assessing the effectiveness of the above approaches is relatively difficult since no work has reported on the generalization performance to automatically create 3D facial models that takes the pose of the face and the death information in the fitting process into consideration from only a photograph.

Luo and King [13] developed a facial-feature-extraction algorithm to automate the process of fitting the general facial wire frame model to the actual face image. Furthermore, by introducing face orientation detection during the fitting process, the facial model construction method is robust with respect to its ability to adjust the specific facial 3D model for arbitrary orientations. However, the depth values of the facial model were adjusted in the z-direction by an amount proportional to its change in 2D image plane [13]. A weak perspective imaging process is assumed valid when depth changes on the face are small compared with the long distance between the face and the camera. This is generally a good approximation, except when viewing the face from close range with a short focal length lens, which results in significant perspective distortion in the image [35]. To adjust the 3D control points of a generic facial model to fit the face image accurately, the pose of the yawed face must be determined and depth information of facial features must be taken into account simultaneously, especially when viewing the face from close image. How to accurately modify the generic facial model to fit the specific face image from an uncalibrated face image using the perspective imaging process is investigated in the work.

Two fundamental problems which must fully cooperate with each other are the model modification method and the establishment of the generic facial model, described as follows.

(a) Used as an optimization technique, genetic algorithms (GAs) have prove to be an effective way to search extremely large or complex solution spaces. Since genetic algorithms do not rely on problem-specific knowledge, they can be used to discover solutions that would be difficult to find by other methods. Due to the complexity and large variance of face images, many GA-based approaches are applied to the applications about human face [16–26]. Guarda et al. [16] used GA techniques to learn visual feature and proposed a program which combines and integrates the features in non-linear ways to design a face detector. Ohya and Kishino [17] detected deformations of facial parts from a face image regardless of change in the position and orientation of a face using a genetic algorithm. In addition, GAs are also used in the human face detection [18–20], facial feature extraction [21–23], human face location in images sequence [24], human posture estimation [29], and automated face recognition [26].

(b) An effective generic facial model is in general problem-dependent. In addition, the establishment of the generic facial model depends on the modification method for facial modeling. Parameterized facial model can produce realistic and manipulable face images with a small number of parameters [27,28]. Although parametric facial modeling has received considerable interest, the inverse problem of extracting parameters from face images has seldom been addressed. Therefore, developing a complete parameter set which can be automatically and easily manipulated is extremely difficult, especially from a monocular face image. Various kinds of the wire frame facial models are adopted in various applications [1,4,12,29,30]. The larger number of triangle elements implies a better quality of the synthesis image. However, the complexity of facial modeling grows.

In the light of the above two problems, this paper presents a GA-based optimization approach for facial modeling from an uncalibrated face image using a flexible generic parameterized facial model (FGPFM). The microstructural information can be expressed using the structural FGPFM with representative facial features that can be accurately found in the image. The reconstruction procedure can be regarded as a block function of FGPFM, and the input parameters are the 3D face-centered coordinates of control points. Once the control points are given, the desired 3D facial model is determined based on the topological and geometric
descriptions of FGPFM. How to reconstruct the 3D facial model is transformed into a problem of how to acquire the accurate 3D control points.

Since the solution space is large and complex considering the large number of control points in 3D space, the proposed coarse-to-fine approach based on our intelligent genetic algorithm IGA [31] is used to efficiently solve the optimization problem. IGA is an efficient general-purpose algorithm capable of solving large parameter optimization problem. Coarse-to-fine approach can efficiently adjust control points in 3D space. The fitness function takes into account the evidence from the face image and human perception. The proposed coarse-to-fine IGA can effectively construct an optimal facial model. Merits of the proposed method are summarized as follow. (1) FGPFM is presented so that the good parameters, the control points, of FGPFM can yield the good facial model for a specific person. (2) An analytic solution for the pose determination of human faces (PDF) [32] from a monocular image is applied to obtain the initial 3D control points and make coarse-to-fine IGA more efficient. (3) The reconstruction problem is formulated as a parameter optimization problem based on the ability of FGPFM and PDF. Furthermore, the coarse-to-fine IGA is also used to speed up the search for an optimal solution which is a set of optimal control points. The facial model construction method can obtain more accurate facial models with respect to its uses of the perspective fitting process and the coarse-to-fine IGA for adjusting the 3D control points of FGPFM, compared with that of the existing ones, e.g. Ref. [13].

The rest of this paper is organized as follows. Section 2 describes how to establish a flexible generic parameterized facial model. Section 3 formulates the facial modeling problem as an optimization problem and also outlines the reconstruction procedure. Section 4 summaries the sensitivity analysis and shows the experimental results with texture mapping. Conclusions are finally made in Section 5.

2. Establishment of FGPFM

The FGPFM consists of a topological structure and geometric knowledge of human faces. The topological description consists of a set of well-designed triangular polygons with a multi-layered elastic structure in which the microstructural information can be expressed without complicated facial feature. All the geometric values are obtained from a set of training facial models using statistical approaches and genetic algorithms.

2.1. Topological description of FGPFM

The topological description of FGPFM consists of a set of well-designed triangular polygons that are constructed mainly by the control points and the interrelationship between the control points. The topological description consists of three parts and is denoted as $M = (V, U, T)$. $V = \{v_1, v_2, \ldots, v_n\}$ is the set of control points and vertices where $v_i = (x_i, y_i, z_i)$ is a face-centered coordinate. Let $V$ consist of three disjoint subsets, $V_1$, $V_2$, and $V_3$, where $V_i$ is the set of control points all in the foundation layer and functions as the parameters of FGPFM. The control points are easily extracted using automatic algorithms and can be differentiated between various 3D models. $V_2$ is the set of vertices in the foundation layer, and all vertices are determined by the set $U$ of interrelationship functions of control points. Let $U = \{ f_1, f_2, \ldots, f_m \}$. Every one of the $m$ vertices in $V_2$ can be derived from an individual function $f_i$. Function $f_i$ is the weighted linear combination of relative neighboring control points by interpolation/extrapolation algorithms and is obtained by averaging a set of training example face models. $V_3$ is the set of vertices in the refinement layers and all these $v_i$ vertices can be derived from the triangular polygon descriptor set $T$. Let $T = \{ t_1, t_2, \ldots, t_m \}$ where $t_j$ is the $j$th triangular polygon descriptor and $t_j = (v_p, v_q, v_r, h_j)$. Each triangle described by $t_j$ consists of three control points/vertices, $v_p$, $v_q$, and $v_r$. Let $v_h$ be the center of gravity of $\Delta v_p v_q v_r$ and $\overrightarrow{N_j}$ is the unit normal vector of $\Delta v_p v_q v_r$. Let $v_h$ be on the curvilinear surface of a facial model where $\overrightarrow{v_h v_h} = h_j \overrightarrow{N_j}$. Notably, $h_j$ reveals the curvature of the area described by $t_j$ and serves as a parameter of vertex generation function of $v_h$. If $h_j$ is greater than the threshold value, the new vertex $v_h$ is added to $V_3$. The renewed triangulation process for the neighboring triangles of all new vertices is proceeded as follows. For each triangle $\Delta v_p v_q v_r$ with a new vertex $v_h$, the three edges, $v_p v_h$, $v_q v_h$, and $v_r v_h$, form the new additional edges. Let $v_p v_q$ be the common edge between $\Delta v_p v_q v_h$ and some connected triangle $\Delta v_p v_q v_s$. If $v_h v_s$ is closer to the curvilinear surface than $v_p v_q$, then $v_h$ replaces $v_p v_q$. The curvature threshold value $H$ of FGPFM dominates the adaptive partition of triangles. When the curvature parameter $h$ of some triangular polygon is greater than $H$, the vertex $v_h$ in the next layer is added to FGPFM. Fig. 1 illustrates the foundation-layered and two-layered FGPFMs.

2.2. Geometric description of FGPFM

All the geometric values used in FGPFM are obtained by measuring some actual human faces using statistical approaches and genetic algorithms. Initially, we define a face-centered coordinate system (FCCS) used for FGPFM. Let the four far corners of the eyes and mouth in FGPFM be coplanar and the facial plane be defined as the $XY$ plane of the FCCS. Let the horizontal axis be the $X$-axis which is located on the center and is parallel to the eye-line and the mouth-line. Let the symmetric axis of
FGPFM be the Y-axis and the positive Z-axis be outward. The following geometric values are recorded in FGPFM.

(a) All the 3D coordinates \((x, y, z)\) of control points. The \(z\) value serves as the parameter for the depth recovery process.

(b) All of the curvature parameters for every vertex generation function.

(c) All the weights in the interrelationship functions of set \(U\).

(d) The statistical model ratios \(r_k\), \(k = 1, 2, \ldots, l\) \((l = 10\) in our system). These model ratios vary from face to face which are essential and helpful for fitting the FGPFM to a specific actual face image. We define the notations as follows, as illustrated in Fig. 2.

1. \(L_u\): the distance between two far corners of the eyes.
2. \(L_f\): the distance between the mouth-line and eyes-line.
3. \(L_m\): the distance between the nose base and the mouth-line.
4. \(L_w\): the width of the mouth.
5. \(L_{e1}\): the width of one eye.
6. \(L_{e2}\): the width of the nose.
7. \(L_{31}, L_{32}\) and \(L_{33}\): the distances between the \(YZ\) plane and the 3D points \(Q_1, Q_2,\) and \(Q_3\) which are parallel projected on the intersections of the eye-line, \(X\) axis, and mouth-line, and the contour of the 2D full-front face, respectively.
8. \(L_{34}\): the distance between the \(XZ\) plane and the 3D point \(Q_4\) which is parallel-projected on the intersection of the mouth-corner-line which is parallel to the \(Y\)-axis and the contour of the 2D full-front face.

9. \(L_5\): the distance between the \(XZ\) plane and the 3D point \(Q_5\) which is parallel-projected on the intersection of the \(Y\)-axis and the contour of the 2D full-front face.

The ten model ratios \(r_k\) recorded in FGPFM are defined as follows: \(r_1 = L_x/L_e, r_2 = L_u/L_e, r_3 = L_m/L_e, r_4 = L_y/L_e, r_5 = L_n/L_e, r_6 = L_{11}/L_e, r_7 = L_2/L_e, r_8 = L_3/L_e, r_9 = L_4/L_e,\) and \(r_{10} = L_5/L_e\).

2.3. Automatic learning for geometric values

All the model ratios and robust weights in the interrelationship functions of the set \(U\) for FGPFM can be measured manually from two orthogonal face images of the representative persons and the average measurements are recorded. Two sets of geometric values, the depth values of the control points and the curvature parameters, are not easily estimated manually. These sets are derived from a set of the individuals by an automatic machine learning approach. The training images with yawed angles, 0, 30, and 90°, from its front-viewed position are used. The training images are normalized by centering the face images on the image plane and by using the same camera parameters. The same normalization process is also applied to the reconstruction of a facial model.

The optimum geometric values of FGPFM result in full coincidence between superimpositions of the transformed FGPFM and those facial contours of training images. Geometric values of FGPFM are established using the profile matching technique for silhouette of the training image and FGPFM with the known/estimated view directions. A set of Fourier features of profiles is used to measure the similarity of profiles. Let the NFDs of three projected facial contours with yawed angles, 0, 30, and 90°, of an FGPFM be concatenated to one feature vector \(B\). Let the representative feature vector be \(\bar{B}\) from average measurements of a set of the individuals’ face images.

In this study, we apply the genetic algorithm (GA) to obtain a good set of geometric values of FGPFM. The basic procedure of GA can be found in Ref. [34]. To do
so, we have to design the encoding scheme of chromosomes and the fitting function for GA. An individual in a population generated by some encoded schemes is defined as a chromosome. The value of \( i \)th position in the chromosome is termed a gene which represents one geometric value. In our system, the feature vector \( X_f \) consists of 14 depth values for control points and 44 curvature values based on the symmetry of the face. We want to find an optimal \( X_f \) such that the various profiles of the representative FGPFM and the training face images have maximal similarity. The fitting function is defined as

\[
\text{Max. Similar } (X_f) = \text{Dist}(B, \tilde{B}),
\]

where Dist() is the Euclidean distance. \( B \) is obtained using the following procedure.

Step 1: Decode a chromosome to obtain the feature vector \( X_f \).

Step 2: Construct the 3D facial model using \( X_f \) and the known information of the representative FGPFM.

Step 3: Project the 3D facial model to obtain three normalized facial profiles using the specified view directions.

Step 4: Compute NFDs \( B \) of the projected facial profiles.

Notably, the above large parameter optimization problem is difficult to be solved for deriving an accurate solution using the simple genetic algorithm. Ho et al. [31] proposed a novel intelligent genetic algorithm (IGA, see the appendix) which is superior to the conventional genetic algorithms and, therefore, is used to derive a better solution \( X_f \).

3. Facial modeling as an optimization problem

As widely recognized, accurate 3D control points based on FGPFM can lead to an accurate 3D facial model of a specific individual. Herein, the reconstruction problem is formulated as a parameter optimization problem as follows.

Find a set of control points \( V_i \),

such that \( F(V_i) = \text{Min. } F(V_i) \),

where \( V_i \) is the set of control points of the FGPFM. The two major problems are:

(a) How to construct the fitness function \( F(V_i) \)?

(b) How to search for the optimal solution \( V_i \)?

A coarse-to-fine IGA is presented to cope with the above two problems.

3.1. Fitness function

The formulation of fitness function \( F(V_i) \) closely corresponds to the quality of the constructed 3D facial model. Two criteria for evaluating the quality of the facial model are presented as follows

(a) Projection of the facial model from some viewpoint must coincide with the features in the given face image.

(b) The facial model must adhere to the generic knowledge of human faces accepted by the human reception.

Let \( R = \{r_1, r_2, \ldots, r_l\} \) denote the set of model ratios and \( S = \{S_1, S_2, \ldots, S_N\} \) where \( S_i \) is a control point/vertex or a point which may be projected on the silhouette of the transformed FGPFM. \( R \) and \( S \) can be determined from \( V_i \). According to the two criteria, we define

\[
F(V_i) = w_1 f_1(S) + w_2 f_2(V_i) + w_3 f_3(V_i) + w_4 f_4(R),
\]

where \( w_1, w_2, w_3 \) and \( w_4 \) are weighting constants. Four error estimation functions are described as follows.

1. Projection function \( f_1(S) = \langle 1/c \rangle \sum_{i=1}^{N} \text{Dist}(s_i, s_i') \).

Where \( s_i \) and \( s_i' \) are the projections of \( S_i \) and its corresponding feature point in the face image, respectively.

2. Symmetry function \( f_2(V_i) = \langle 1/|V_i| \rangle \sum_{i=1}^{N} S_{tm}(v_i) \).

Let the estimated control point \( v_i = (x_i, y_i, z_i) \) and the estimated symmetric point of \( v_i \) be \( v_i' = (x_i', y_i', z_i') \):

\[
S_{tm}(v_i) = \begin{cases} 
  x_i^2 & \text{if } v_i \text{ should be the symmetry axis,} \\
  \left((x_i + x_i')^2 + (y_i - y_i')^2 + (z_i - z_i')^2 \right) & \text{otherwise.}
\end{cases}
\]

3. Depth value function \( f_3(V_i) = \langle 1/|V_i| \rangle \sum_{i=1}^{N} \text{Dist}(z_i, z_i') \), where \( z_i \) and \( z_i' \) are the depth value of a control point in FGPFM and the corresponding estimated depth value, respectively.

4. Model ratio function \( f_4(R) = \langle 1/|V_i| \rangle \sum_{i=1}^{N} \{1 - \sqrt{2} \sigma_i \exp(1/\sqrt{2} \sigma_i \exp(-1/2 (r_i - \mu)/(\sigma_i)^2))\} \), where \( \sigma_i \) and \( \mu_i \) are the mean and standard deviation of model ratios \( r_i \) of FGPFM, respectively.

3.2. Chromosome representation

Chromosome is encoded as a vector \((m_1, d_1), (m_2, d_2), \ldots, (m_{N_v}, d_{N_v})\) with \( 2 + |V_i| \) parameters, where \( |V_i| \) is the number of control points, \( m_i \) is an integer from 0 to 26, and \( d_i \) is an integer from 1 to \( N_{part} \). \( m_i \) represents the moving direction in 3D space and \( m_i = 0 \) means that the \( i \)th control point is unadjusted, and \( N_{part} \) represents the partition number of search space in each direction. Let \( D_i \) represent the radius of search space for the \( i \)th control point. The moving step size \( S_{step} \) is equal to \( D_i/N_{part} \cdot d_i \). The new position of control point \( C = (x, y, z) \) is \( C + S_{step}v_m \), where \( v_m \) means the moving vector for direction \( m \), listed in Table 1.
example, the control point \( C = (20, 30, -5) \), \((m, d) = (17, 4), N_{\text{part}} = 5 \), and \( D_1 \) is 3. Then, \( S_{\text{step}} = 2.4 \) and \( v_17 = (1, -1, -1) \). Therefore, new control point \( C = (22.4, 27.6, -7.4) \).

3.3. Backprojection method for obtaining initial 3D control points

We have successfully presented an analytic solution for the PDF from an uncalibrated monocular face image using a generic facial model [32]. PDF plays an important role in automatic facial modeling. Only four stable facial feature points, the far corners of the eyes and mouth that can be extracted by deformable templet [36], in an image are needed in the PDF procedure. It can provide an exact and analytic solution for the PDF from an uncalibrated monocular face image.

Let the facial plane equation be \( aX + bY + cZ = D \) where the vector \((a, b, c)\) is the facial normal vector derived from PDF and the depth parameter \( D \) is determined from the given value \( L_e \), the distance between the far corners of eyes. Since the facial normal is outward, i.e., \( c < 0 \), \( D \) is smaller than zero. Let \( H_i \) be the depth value of some control point \( p_i \) stored in FGPFM and its corresponding 2D feature point \( f_i = (x_i, y_i, f_i) \) where \( f_i \) is the focal length. Using the backprojection technique, the viewer-centered control point \( p_i \) is equal to \((x_{ti}, y_{ti}, f_{ti})\), where \( t_i = (D + H_i)/(ax + by + cf) \). It is easy to derive the face-centered control point of \( p_i \), \( P_f = (x_f, y_f, z_f) \) using the equation

\[
\begin{bmatrix}
x_f \\
y_f \\
z_f \\
w
\end{bmatrix} =
\begin{bmatrix}
x_{ti} & N_{ey} & N_{ez} & 0 \\
x_{ti} & N_{ey} & N_{ez} & 0 \\
x_{ti} & N_{ey} & N_{ez} & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x \\\ny \\\nf \\\n1
\end{bmatrix}
\]

where \((x_0, y_0, z_0)\) is the origin of the face-centered coordinate system with respect to the view-centered coordinate system and the three facial axes vector \( N_e = (N_{ex}, N_{ey}, N_{ez}) \), \( N_s = (N_{sx}, N_{sy}, N_{sz}) \) and \( N_f = (N_{fx}, N_{fy}, N_{fz}) \) are derived from the results of PDF. It is noted that \( z_f = H_i \).

Generally, the iterative process without predetermined the pose is adopted to modify the 3D facial model until the superimposed face contours resemble those on the face image [27,29]. However, using the iterative process to solve the combination problems of the pose and depth is not only difficult but also inefficient. Herein, we solve this difficult problem based on the ability to separate the pose and depth problems. The estimated initial control points are helpful for IGA to accelerate the search process.

3.4. Coarse-to-fine IGA

Due to the huge search space for adjusting 3D control points, the search process of optimal control points consists of many steps using various partition resolutions of search space. Theoretical analysis, experimental studies and the application of IGA can be found in our recent work [31,37]. IGA has demonstrated the capabilities of solving the large parameter optimization problems with fast convergence and high accuracy.

We present a coarse-to-fine approach using IGA to efficiently find the optimal control points. The coarse and fine searches can be regarded as global and local searches, respectively. The parameter \( D_i \), radius of search space, serves as the tuning parameter. Call one adjustment of all control points using one run of IGA one step. Let the radius of the search space for the \( j \)th step be \( D_j = D_{j-1}/\rho \) where \( D_1 \) contains all the global search space. The stopping condition for one step can be determined by an improving factor \( E_{\text{step}} \):

\[
E_{\text{step}} = \frac{|X_i - X_{i-1}|}{X_{i-1}}, \quad (5)
\]

where \( X_i \) represents the fitness function evaluation value of the \( i \)th generation in IGA. If \( E_{\text{step}} \) is smaller than a threshold value \( \rho \), it means that the optimal control point set has been obtained in the given search window.
Once the control points are adjusted by moving one step according to the derived moving direction and step size, the radius of search window for next move must be adjusted in order to refine the solution. Notably, an elitist strategy is adapted in the coarse-to-fine IGA by adding a specific chromosome ((0, 1), (0, 1), ..., (0, 1)) into the initial population for the next step. The stopping condition of the entire coarse-to-fine IGA algorithm can be determined by

$$E_{\text{stop}} = \frac{|Y_i - Y_{i-1}|}{Y_{i-1}}$$

where $Y_i$ represents the fitness function evaluation value of $i$th step in IGA. If $E_{\text{stop}}$ is smaller than a threshold value $\lambda$, the stop condition is met.

### 3.5. Overview of the facial modeling algorithm

Input data of the reconstruction procedure is a 2D face image with the visibility of the eyes and mouth corners, and an FGPFM. Output is the 3D face-centered facial model. Fig. 3 depicts the block diagram of the facial modeling. The coarse-to-fine IGA plays the key point to obtain the facial model of a input face image. The entire facial modeling procedure is designed as follows.

**Step 1**: Extract facial feature points using deformable template method, such as mouth contour, eye contours, head contour and nose feature (if possible).

**Step 2**: Obtain the face pose using PDF procedure.

---

![Fig. 3. The block diagram of the facial modelling Notation: □: coarse-to-fine IGA, and -- one step.](image-url)
4.1. Experiment 1

Step 3: Derive the initial 3D control points by back-projection technique.

Step 4: Set starting set of control points using elitist strategy.

Step 5: Perform one generation of IGA to obtain a better solution for one step.

Step 6: If stopping condition of one step is not met, i.e., \( E_{\text{step}} = \rho \), go to Step 5.

Step 7: Adjust the control points by moving one step.

Step 8: If stopping condition of the coarse-to-fine IGA procedure is not met, then go to Step 4.

Step 9: Use the optimal solution of the coarse-to-fine IGA to obtain an optimal solution of \( F(V_1) \) and the control point set \( V_1^+ \) of FGPFM.

Step 10: Reconstruct the optimal facial model using \( V_1^+ \) and FGPFM.

4. Experimental results

In this section, five experiments using FGPFM, synthetic face images and actual face images are analyzed to demonstrate the feasibility of the proposed method. In the first experiment, a known facial model is used to verify the superiority of IGA and the effectiveness of the coarse-to-fine approach. The second experiment demonstrates the effectiveness of the proposed fitting function and the high performance of the coarse-to-fine IGA. In the third and fourth experiments, an application applies our algorithm to obtain optimal control points in the reconstruction of a 3D facial model from a monocular face image and two uncalibrated face images, respectively. In the least experiment, the impact of facial expressions on the accuracy of the constructed face models is demonstrated and discussed. In all experiments, the parameters of IGA and the simple genetic algorithm with elitist strategy (ESGA) are: the population size is 20, mutation rate is 0.1, and the crossover rate is 0.5. The number of control points \( |v_1| \) is 24 and \( L_v = 100 \).

4.1. Experiment 1

In this experiment, the graphics-generated facial models with perturbation are used to investigate the capability of the proposed algorithm for searching optimal control points. The test models are generated using the randomly perturbed control points in a well-defined FGPFM. Let each of the control points randomly drift in 3D space inside the area in which the radius is equal to \( w \% \) of \( L_v \). The aim of the GAs is to adjust the perturbed 3D control points to superimpose with original 3D ones in FGPFM.

Owing to the elastic structure of FGPFM, the control points must be globally adjusted to achieve a good facial model. We apply IGA and ESGA to obtain a good control point set of FGPFM. The fitness function is

\[
\text{Min. Fit} = \sum_{i=1}^{w} \text{Dist}(p_i - q_i),
\]

where \( p_i \) and \( q_i \) are the unperturbed and adjusted control points, respectively.

In order to demonstrate the superiority of IGA and the effectiveness of the coarse-to-fine approach, two simulation results, shown in Figs. 4 and 5, are given for comparison and analysis. In Fig. 4, no coarse-to-fine procedure is used. In other words, only one step is adopted. However, the partition number \( N_{\text{part}} \) is decoupled and the stopping condition is 5000 generations. The average results using 10 independent runs of IGA and ESGA are illustrated in Fig. 4 for various \( w \) values. The average results using 10 independent runs of coarse-to-fine IGA and coarse-to-fine ESGA are illustrated in Fig. 5 for various \( w \) values. From these figures we can infer:

1. In Figs. 4 and 5, IGA is obviously superior to ESGA.
2. In Fig. 4, the average final Fit values of IGA and ESGA after 5000 generations are 28.2, 51.5 and 89.4 with respect to \( w = 4, 10 \), and 20. The average final Fit values after 10 steps in Fig. 5 are 2.4, 2.6 and 2.8. It reveals that IGA and ESGA with the coarse-to-fine procedure are superior to those without coarse-to-fine procedure for robustly finding the optimal solution.

A typical result of the coarse-to-fine IGA in convergence speed and accuracy is illustrated in Fig. 6. Fig. 6(a) illustrates the relationship between the generations in IGA and the steps in the coarse-to-fine procedure with \( w = 20 \). Fig. 6(b) illustrates the comparisons of convergence and accuracy under various perturbation conditions. The number of generations for each step can be adaptively adjusted. Since IGA can efficiently find the better solution with a small number of generations in one step (see Fig. 4), coarse-to-fine IGA can economically find the optimal solution by sufficiently making use of the IGA’s ability for every step, as shown in Fig. 6(a).

Define the relative error \( E_{\text{rr}} \) as follows:

\[
E_{\text{rr}} = \frac{\text{Fit}}{|v_1|L_v}.
\]

After 400 steps of the proposed algorithm, the \( E_{\text{rr}} \) values are equal to 0.0021, 0.0022, and 0.0040\% for \( w = 4, 10 \), and 20, respectively, as shown in Fig. 6(b). It reveals that the proposed algorithm can robustly and efficiently adjust the generic facial model to an optimal facial model.

4.2. Experiment 2

In this experiment, the used FGPFM is the same as the one used in Experiment 1. The perturbed 3D control
points in FGPFM are the same as those in Experiment 1. An additional synthetic 2D test image is generated by projecting the unperturbed FGPFM with yawed angle 30° and the known focal length. The used fitness function is described in Section 3.1. The aim of the GAs is to adjust the perturbed 3D control points to fit the 2D image of the transformed FGPFM and simultaneously take the human perception of human face into consideration. The simulation results of ESGA, IGA, coarse-to-fine ESGA and coarse-to-fine IGA under various perturbation conditions are illustrated in Fig. 7. The \( N_{part} \) values used in GAs without coarse-to-fine procedure and GAs with coarse-to-fine procedure are 250 and 5, respectively. The performance in Fig. 7 is the average measurement using 10 independent runs. From these figures, it reveals that coarse-to-fine IGA is still superior to other GA approaches using only a single 2D face image.

After executing 400 steps of the coarse-to-fine IGA, the final \( E_{tr} \) values are 0.44, 1.4304, and 4.5529% corresponding to \( w = 4, 10, \) and 20. From these accurate results, it demonstrates the effectiveness of the proposed fitness function for the coarse-to-fine IGA.

4.3. Experiment 3

Two uncalibrated face images, one front-viewed and one yawed face images, are used to examine the applicability of the proposed algorithm to actual facial images with unknown focal length of the camera and optical center of the image, as shown in Figs. 8(a) and (b). Using PDF to obtain the initial control points of FGPFM, the coarse-to-fine IGA is applied to obtain an optimal set of control points. The fitness function used in the coarse-to-fine IGA is described in Section 3.1. Figs. 9(a) and (b) display the convergence from the input images Figs. 8(a) and (b), respectively. Since the exact facial model information cannot be obtained, only the difference between two reconstructed facial models from various poses is of concern. Let the optimal control point sets
Fig. 5. Convergence of GAs with coarse-to-fine procedure using $N_{part} = 4$. (a) $w = 4$, (b) $w = 10$, (c) $w = 20$. Notations: +: ESGA, and o: IGA.

Fig. 6. (a) Relationship between generations in IGA and steps in coarse-to-fine procedure. (b) The convergence and accuracy for various perturbation conditions. Notations: o: $w = 4$, +: $w = 10$, and *: $w = 20$. 
Fig. 7. Simulation results of various GA approaches. (a) $w = 4$, (b) $w = 10$, (c) $w = 20$. A unit GS represents a generation/step for the non-coarse-to-fine/coarse-to-fine approach. Notation: $\therefore$: ESGA, $\ast$: IGA, $+$: coarse-to-fine ESGA, and $\circ$: coarse-to-fine IGA.

Fig. 8. Two input actual face images with different poses.
Fig. 9. The convergence speed and accuracy of Figs. 8(a) and (b), respectively.

of coarse-to-fine IGA independently derived from the initial control point set using Figs. 8(a) and (b) be \( \{ p_{f_1}, p_{f_2}, \ldots, p_{f_{\text{v}}_1} \} \) and \( \{ p_{y_1}, p_{y_2}, \ldots, p_{y_{\text{v}}_1} \} \), respectively. Define the relative difference \( E_{\text{ref}} \) as follows:

\[
E_{\text{ref}} = \frac{\sum_{i=1}^{\text{v}_1} \text{Dist} (p_{f_i} - p_{y_i})}{|\text{v}_1| L_e}
\]  

(9)

After executing 400 steps of the coarse-to-fine IGA, \( E_{\text{ref}} \) is 1.7002%. The small difference means the reconstructed facial models are acceptable. Notably, the model ratios of FGPFM are average measurements from sample face images and may be different from those in the given actual face.

Fig. 10(a) shows the initial facial model constructed from the initial control points of FGPFM using the image in Fig. 8(b). Figs. 10(b) and (c) show the optimal facial model obtained using the coarse-to-fine IGA in various poses. The texture-mapped face images using the reconstructed facial model Fig. 10(b) in various poses are illustrated in Fig. 11.

4.4. Experiment 4

In this experiment, we examine the effectiveness of the proposed algorithm for facial modeling from two uncalibrated face images in Figs. 8(a) and (b). To make use of the evidence from the two face images, the projection function \( f_p(S) \) of the fitness function \( F(V_1) \) is modified. The optimal facial model should be optimally superimposed with all the given face images. Therefore, the projection function takes all the feature points in the given face images into consideration. Two reconstruct facial

Fig. 10. (a) The initial unadjusted facial model in front view. (b) The adjusted facial model using the coarse-to-fine IGA. (c) The yawed facial model of Fig. 10(b).
models, using the sets of initial control points derived from a single face image in Figs. 8(a) and (b), are generated. The $E_{ref}$ is 0.95% after 400 steps of the coarse-to-fine IGA. It reveals that the proposed algorithm can find the optimal solution using various sets of initial control points. In other words, the proposed algorithm is robust for various poses of human faces. Compare the $E_{ref}$ values of Experiments 3 and 4, 1.7002 and 0.95%, it demonstrates that two face images can obtain the better facial model than a single face image. Fig. 12 illustrates the optimal facial model which is the best fitting of Figs. 8(a) and (b) simultaneously.

4.5. Experiment 5

An uncalibrated actual face image with facial expressions is used to discuss and demonstrate the applicability of the proposed algorithm, as shown in Fig. 13. In our system, the statistical model ratios and the weights of the set $U$ of interrelationship functions of FGPFM are learned from actual face images with no facial expression. Dealing with the images with facial expressions, we adopt a larger weight $w_l$ for projection function $f_p(S)$ and a smaller weight $w_s$ for model ration function $f_r(R)$. The facial modeling results for convergence speed and accuracy, reconstructed facial model, and the texture-mapped face images are shown in Figs. 14–16, respectively. From the experimental results, the proposed method can cope with the facial expression problem. However, the quality of the reconstructed facial model can be improved in the following aspects.

1) Due to the large variance among various facial models and facial expressions of individuals, the standard
deviations of model ratios and weights of the set $U$ must be reduced to efficiently modify the FGPFM. The clustering process is helpful to construct the accurate generic facial model for representing a particular set of individuals with facial expressions. Restated, a codebook like database with various typical FGPFMs must be prepared.

(2) In face, if the above-mentioned database of FGPFMs can be used, the reconstruction speed can be improved using the non-GA approach. For instance, in our recent work, a hybrid approach using the Taguchi method and a best-first search algorithm is used to accelerate the search for an optimal solution.

5. Conclusions

This study has presented a novel genetic algorithm-based optimization approach for facial modeling from an uncalibrated monocular face image using flexible generic parameterized facial model. The proposed method, has the following features. (1) FGPFM is presented so that the good parameters, the control points, of FGPFM can yield the good facial model for a specific person. (2) An analytic solution for the pose determination of human faces (PDF) from a monocular image is applied to obtain the initial 3D control points and make the coarse-to-fine IGA more efficient. (3) The reconstruction problem is
formulated as a parameter optimization problem based on the ability of FGPFM and PDF. Furthermore, the coarse-to-fine IGA is also used to speed up the search for an optimal solution which is a set of optimal control points. Finally, sensitivity analysis and experimental results with texture mapping have demonstrated the effectiveness of the proposed method.

Appendix A

An intelligent genetic algorithm (IGA) with a new intelligent crossover (IC) is introduced. The principle of the IC approach is to consider the contribution for individual gene on the fitness based on the ability of orthogonal arrays (OAs). The chromosomes of children are formed from the best combinations of the better genes representing variables of a function coming from the parents rather than the random combinations of parents’ genes to achieve crossover.

A.1. Orthogonal arrays and factor analysis

Orthogonal arrays (OAs) and factor analysis, which are representative methods of quality control [33], also work to improve the crossover operator more efficiently. We provide a definition of the OA as follows. Let there be \( N \) factors of two levels. The number of total combinations is \( 2^N \). Columns of two factors are orthogonal when four pairs, \((1, 1), (1, 2), (2, 1), \) and \((2, 2), \) occur equally in all experiments. When any two factors in an experimental set are orthogonal, the set is called an OA. To establish an OA of \( N \) factors of two levels, we obtain an integer \( n = 2^{\lceil \log_2(N + 1) \rceil} \), build an orthogonal array \( L_n(2^{n-1}) \) with \( n \) rows and \((n-1)\) columns, and select \( N \) columns.

Factor analysis can evaluate the effects of factors on the evaluation function, rank the most effective factors, and determine the best level for each factor such that the evaluation function is optimized. Orthogonal experiment design can reduce the number of experiments for the factor analysis. The number of OAs for single factor analysis is only \( n \). For instance, Table 2 shows an orthogonal array \( L_n(2^7) \). Except that the variable with the lowest rank adopts the other level.

Let \( y_t \) be the positive function evaluation value of experiment no. \( t \) Define the main effect of factor \( j \) with level \( k \) as

\[
s_{jk} = \sum_{t=1}^{n} Y_t^j \times \text{[the level of Exp no. } t \text{ of factor } j \text{ is } k],
\]

(A.1)

where

\[
\text{[condition]} = \begin{cases} 
1 & \text{if the condition is true,} \\
0 & \text{otherwise}
\end{cases}
\]

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Orthogonal array ( L_n(2^7) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exp. no.</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
</tr>
</tbody>
</table>

and

\[
Y_i = \begin{cases} 
 y_i & \text{if the function is to be maximized,} \\
1/y_i & \text{if the function is to be minimized.}
\end{cases}
\]

Note that the main effect reveals the individual effect of a factor. The most effective factor \( j \) has the largest main effect difference \( \text{MED} = |S_{j1} - S_{j2}| \). If main effect \( S_{j1} > S_{j2} \), the level 1 of factor \( j \) is better than the level 2 on the contribution for the optimization function. Otherwise, level 2 is better.

A.2. Intelligent crossover using orthogonal arrays

Genetic algorithm (GA) uses binary variables of a function to represent a chromosome. For instance, a function with \( N \) variables of \( l \) bits is encoded in binary codes of \( Nl \) length in a chromosome, or \( Nl \) binary variables. The representation of chromosomes for IC approach is the same as that of the traditional GA. Two parents breed two children using IC at a time. How to use the OA to achieve the IC is described in the following steps.

Step 1: Select the first \( N \) columns of OA \( L_n(2^{n-1}) \) where \( n = 2^{\lceil \log_2(N + 1) \rceil} \). Note that one variable of a function is regarded as a factor in OA.

Step 2: Let level 1 and level 2 of factor \( j \) represent the \( j \)th variable of a function coming from the parent 1 and parent 2, respectively.

Step 3: Evaluate the function values \( y_t \) for experiment no. \( t \) where \( t = 1, 2, \ldots, n \).

Step 4: Compute the main effect \( S_{jk} \) where \( j = 1, 2, \ldots, N \) and \( k = 1, 2 \). Determine the best level for each variable. Select level 1 for the \( j \)th variable if \( S_{j1} > S_{j2} \). Otherwise, select level 2.
Step 6: The chromosome of the first child is formed from the best combinations of the better variables from the derived corresponding parents.

Step 7: Rank the most effective factors from rank 1 to rank N. The factor with large MED has a higher rank.

Step 8: The chromosome of the second child is formed similarly as the first child.

A.3. Intelligent genetic algorithm IGA

Traditional genetic algorithm which is called simple genetic algorithm (SGA) consists of five primary operations: initialization, evaluation, selection, crossover, and mutation operations. IGA may use the same initialization as SGA. The IGA uses a novel intelligent crossover (IC) based on the ability of orthogonal arrays (OAs). However, an improved initialization for producing the population of the first generation to obtain fast convergence speed is presented in the following steps.

Step 1: Give a population randomly which consists of \( N_{pop} \) individuals, \( I_i \), \( i = 1, 2, \ldots, N_{pop} \).

Step 2: Elitist strategy: Repeat the following steps for \( i = 1 \) to \( N_{pop} - 1 \):

Step 2a: Select \( I_i \) and \( I_{i+1} \) as the parents and produce the two children \( I_1 \) and \( I_2 \), using IC.

Step 2b: Replace \( I_i \) and \( I_{i+1} \) with the second and the best individuals using fitness performance among \( I_i \), \( I_1 \), and \( I_2 \), respectively.

Step 3: Evaluation: Evaluate the function values for all individuals.

Step 4: Selection: Use the rank selection that replace the worst \( N_{pop} \times P_s \) individuals by the best \( N_{pop} \times P_s \) individuals to form the new population. According to the selection probability \( P_s \), select \( N_{pop} \times P_s \) parents for intelligent crossover operations.

Step 5: Crossover: Apply IC to the selected pairs of parents. The two children are replaced by two individuals with the better fitness function values among the parents and children for the elitist strategy.

Step 6: Mutation: Apply the mutation operator to the generated new population using mutation probability \( P_m \). To prevent the fitness value from deteriorating, mutation is not applied to the best individual.

Step 7: Termination test: If a prespecified stopping condition is satisfied, end the algorithm. Otherwise, return to Step 3.

References


About the Author—SHINN-YING HO was born in Taiwan, ROC, on March 25, 1962. He received the B.S., M.S., and Ph.D. degrees in computer science and information engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1984, 1986, and 1992, respectively. He is currently an associate professor in the Department of Information Engineering at Feng Chia University, Taichung, Taiwan. His research interests include image processing, pattern recognition, virtual reality applications of computer vision, genetic algorithms, large parameter optimization problems, and system optimization.

About the Author—HUI-LING HUANG was born in Taiwan, ROC, on April 4, 1968. She received the B.S. degree in Mathematics from Fu Jen C.U., Csin Chuang, Taiwan, in 1992, and the M.S. degree in information engineering from Feng Chia University, Taichung, Taiwan, in 1998. She is presently working toward the Ph.D. degree in the Institute of Information Engineering at Feng Chia University, Taichung, Taiwan. Her research interests include image processing, pattern recognition, virtual reality applications of computer vision, and large parameter optimization problems.