Reconstruction of Animatable Personalized 3D Faces by Adaptation-based Modeling

Yu Zhang, Terence Sim and Chew Lim Tan

Department of Computer Science,School of Computing
National University of Singapore, Singapore 117543

Abstract

We present an efficient method for the construction of an animatable 3D facial model of a specific person with minimal user interaction. The method is based on adapting an anatomy-based prototype facial model that is suitable for physically-based facial animation to the geometry of a real person's face recovered from laser-scanned range data. Starting with specification of a set of anthropometric landmarks on the 2D images, we automatically recover the 3D positions of the landmark points on the facial surface. A global shape adaptation is then carried out to align the prototype model to the target geometry using the transformation parameters estimated from measurements between recovered 3D landmark points. A local shape adaptation follows to deform the prototype model for fitting all of its vertices to the scanned surface data. The reconstructed 3D face portrays the geometry and color of the individual face and can be animated immediately with the given muscle parameters.

1. Introduction

Ever since the pioneering work of Parke [16], researchers have investigated techniques for generating facial models and animation. The approaches to facial modeling described in literature range from parameterization of 3D geometric surface models (see e.g. [17]) to models which involve detailed simulation of physical properties of the anatomical facial structures (see e.g. [20]). However, most existing modeling strategies are based on the generic facial models which only depict the shape and features of an average human face. Since a hallmark of the individuality of the people is the range of variation in the shape of their faces, an animation that fails to reproduce this diversity deprives its characters of independent identities. To animate a scene realistically or to play out a virtual interaction believably requires reconstruction of the face of a specific person, i.e. cloning a real person's face.

Current technology allows us to acquire precise 3D geometry of a face easily by using a range scanning device. 3D models reconstructed automatically from range data can bear very good resemblance to the specific persons, especially if they are properly textured. In practice, though, it turns out that there are several obstacles to using the acquired geometry directly for an animatable model:

- the geometry is heavily oversampled: direct conversion of the range data to a triangular mesh regularly yields hundreds of thousands of triangles. We need to reduce the complexity to less than 10k triangles for real-time animation. For optimal models, edges should be properly aligned to facial features, the mesh structure should reflect the basic symmetry of the face, and complexity should vary in different areas, depending on intensity of articulation in that area. Available mesh simplification techniques [2] unfortunately do not give enough control over the mesh connectivity to guarantee satisfying animatable models.
- they provide only the outward shapes of human faces without structured information, therefore are difficult to animate effectively. Besides 3D shape data for the facial surface, sophisticated facial animation techniques often require additional features of the face such as the eyes, eyelids, teeth and lips. These structures are difficult to be created from range data (e.g., there is no opening in the regions of the mouth and eyes). The model should also contain the animation mechanism for controlling facial surfaces deformations, jaw rotation, eyelid opening, etc.

In this paper, we present an efficient method for creating a personalized facial model by adapting a prototype

physically-based model to the geometry of an individual face. The generic prototype facial model has a layered anatomical structure for controlling facial motions and expressions. The face geometry and texture of real individuals are recovered from a set of range and reflectance data acquired from a laser range scanner. For adaptation, we first specify a minimum set of anthropometric landmarks on the 2D images of both the prototype and individual faces to identify facial features. The 3D positions of the landmarks are recovered automatically by using a projection-mapping approach. Based on a series of measurements between the recovered 3D landmark points, a global shape adaptation is then carried out to adapt the size, position and orientation of the prototype model in the 3D space. After global adaptation, a local shape adaptation deforms the skin geometry of the prototype model to fit all of its vertices to the surface data of the real face. The resulting model shares the same muscle structure with the initial prototype model and can be animated immediately using the given muscle parameters.

The novel features of our algorithm are:

- Efficient face reconstruction technique with minimum user intervention.
- A new projection-mapping approach to recover the 3D coordinates of landmark points defined in 2D images.
- Automated global adaptation process with no restriction on the position and orientation of the prototype model and scanned data.
- Framework for representing a static face scanned data-set for efficient animation.

This paper is organized as follows. Section 2 reviews the previous work on modeling of a personalized face. Section 3 presents the laser-scanned face data and our prototype model used for adaptation. The details of our three-step face adaptation algorithm are described in Section 4. Section 5 shows the results of our experiments with the proposed method. Finally in Section 6, conclusion and future work are given.

2. Previous Work

Regarding modeling a specific person's face, a variety of methods have been proposed. In the early works, Parke [17] created various facial shapes by changing the conformation parameters of a generic parametric model. His parametric model is restricted to the range that the conformation parameters can provide, and manually tuning the parameters for a personalized face is tedious. Another early approach to enable the reconstruction of an individualized face is manual construction of facial models, by using a plaster model [14] or interactive deformation [11], which can give nice results, but require considerable artistic skills and are time-consuming.

The more efficient approach to reconstruct a face shape is the image-based technique. Kurihara and Arai [10] used an interactive method to get a few points, and a Delaunay triangulation for the conformation of the face. The drawback of their method is that too few points are used to guarantee appropriate shape from a very different generic face or an accurate texture fitting. Ip and Yin [8] used similar approach to the one of Akimoto et al. [1]. These two approaches tried to automatically detect feature points on two orthogonal facial images captured from frontal and side views using dynamic template matching or Local Maximum Curvature Tracking (LMCT) [22] to check concave and convex points on the side profile of a face. Lee et al. [12] applied a modified Laplacian operator to the range map to make facial features more evident. Mesh adaptation procedures on the Laplacian field map then automatically identify feature points. Lee et al. [13] used structured snakes for extracting the profile boundaries and facial features. The individualized head is then produced by modifying a generic model using Dirichlet Free Form Deformations (DFFD) [15]. Generally, although the image-based technique can provide a facial model easily from 2D information, the methods for automatically finding corresponding feature points are not robust and the reconstructed 3D shape is not completely accurate.

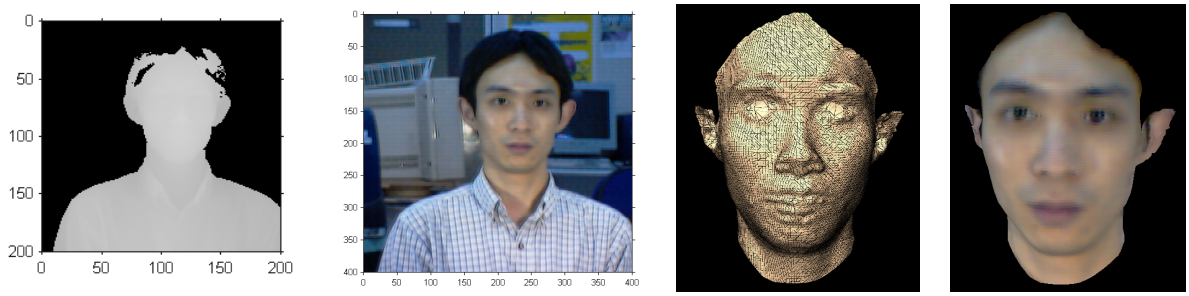
A variational approach is presented by Decarlo et al. [3] to construct a range of static facial models with realistic proportions without assistance from images. They used anthropometric measurements from [6], which constrain the deformation of a B-spline generic head model.

To recover an accurate shape of facial surface, a few methods have been utilized to obtain detailed range data of faces, such as laser scanning [12], stereoscopy [5, 7] and stripe generator [19]. However, as described in Section 1, the models obtained by these methods are poorly suited for animation. Information about the facial structure is missing; measurement noise produces artifacts; and model vertices are poorly distributed.

Some approaches morph a generic facial mesh into specific shapes with scattered data interpolation technique. Ulgen [21] used 3D-volume morphing to obtain a smooth transition from a generic facial model to a target model. Pighin et al. [18] interactively mark corresponding facial features in several photos of an individual to deform a generic head model using radial basis functions. Enciso et al. [5] and Kähler et al. [9] also used landmark information to specify a deformation of the object space so that they can warp the complete generic head to a prescribed target configuration. However, the scattered data interpolation approach only guarantees that the morphing of the generic model is accurate near the landmark points, and since these points are sparse and scattered, a large number of landmarks are needed to ensure the overall quality of the morphing. The manual specification of a dense set of landmarks can be laborious and thus restricts this method to be implemented efficiently in practice.

3. Scanned Geometry and Animatable Prototype Model

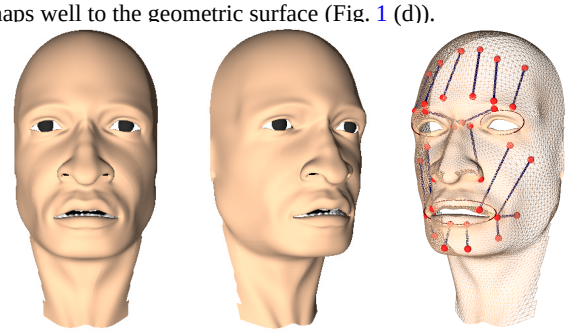
Facial geometry and texture information is acquired using a laser range scanner Minolta VIVID 700 Digitizer^{TM}. When



(a) (b) (c) (d)

Figure 1: *Range data of a face with details but without functional structure: (a) range map; (b) reflectance image; (c) target mesh recovered from range scans (12,194 vertices and 23,642 triangles); (d) target model rendered with texture mapping.*

each scan is completed, the device has acquired two registered images of the subject: a 200 × 200 range image (Fig. 1 (a)) and a 400 × 400 reflectance (RGB) image (Fig. 1 (b)). The range map can be transformed into a set of 3D points. After Delaunay triangulation of the point set and an interactive post-processing to fill holes and to remove noise, a triangular mesh representing the facial surface is generated (Fig. 1 (c)). We use the acquired reflectance image for texture mapping. As the reflectance image is automatically registered against the range data, texture coordinates are attached to the data structure of recovered vertices and facial texture maps well to the geometric surface (Fig. 1 (d)).



(a) (b) (c)

Figure 2: *The generic prototype facial model with animation structure: (a) and (b) two views of face geometry; (c) layered structure of the skin and muscles.*

Although the sacnned data captures the geometry and coloring of a person, it is not suitable for direct animation. We have developed a prototype facial model that resembles an average human face for use in our physically-based animation system [23]. The facial model has a hierarchical structure of the skin, muscle, and subsidy components (Fig. 2).

- The skin layer is represented by a triangular mesh. It consists of 2746 vertices and 5253 triangles. The edges are aligned to facial features to reduce animation artifacts. Basically, this skin mesh is converted to a mass-spring system: each vertex corresponds to a point mass and along each edge of a triangular element there is a nonlinear spring connecting two adjacent mass points.
- A layer of facial muscle actuators are attached to facial skin to control facial movement. Different kinds of physical facial muscles have been modeled to simulate the contraction of the muscle in a linear or circular fashion. In the prototype model, 23 facial muscles which are primarily related to facial expressions are simulated. Each muscle has the contraction parameter to animate facial expressions.
- Additional geometric models of eyes and teeth are included.

Animation of the face is achieved by physics-based simulation of the mass-spring skin mesh under the influence of a field of muscle forces [23]. In the prototype facial model, motion parameters such as muscle contraction rate and jaw action parameters are used to simulate facial action. At the higher level, animation is controlled by expression commands based on the Action Units of the Facial Action Coding System [4].

4. Facial Model Adaptation

The facial model reconstructed from the range data is called the *target model* $\mathcal{F}^*$, while the prototype face is referred to as the *source model* $\mathcal{F}$. By adapting $\mathcal{F}$ to $\mathcal{F}^*$, $\mathcal{F}$ takes on the shape and texture of specific individual and we can animate the personalized model with predictability. The adaptation of $\mathcal{F}$ takes place in three successive steps:

1. Landmark location. This procedure tags $\mathcal{F}^*$ with a set of landmarks corresponding to the set specified on $\mathcal{F}$ and automatically recovers 3D positions of all landmarks.
2. Global shape adaptation. This automated procedure accounts for resizing of $\mathcal{F}$ as well as repositioning of both $\mathcal{F}$ and $\mathcal{F}^*$ to make them aligned in the 3D space.
3. Local shape adaptation. A deformation is computed based on the aligned shapes, and position of all vertices on $\mathcal{F}$ is fitted to $\mathcal{F}^*$ accordingly.

4.1. Landmark Location

The landmark location procedure begins with the specification of a set of landmarks on the 2D texture images. For $\mathcal{F}$, we first save a bitmap from the color buffer using OpenGL.

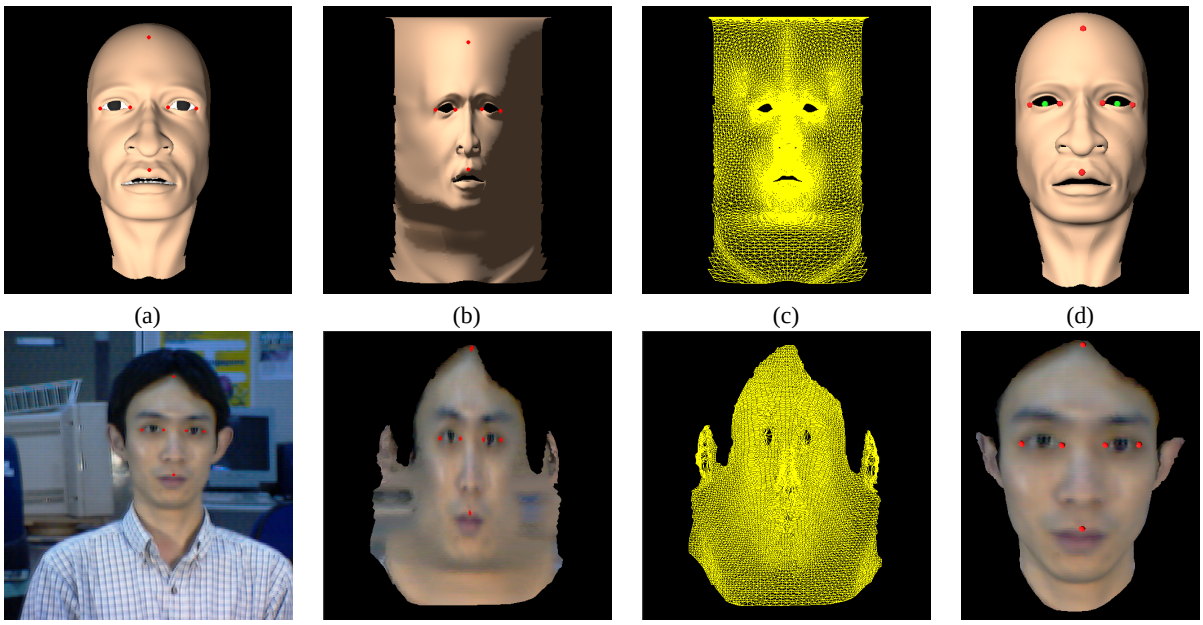


Figure 3: *Specification of landmarks and recovery of their 3D positions based on a projection-mapping approach: (a) landmark image of the source model; (b) projected textured-mapped source model; (c) projected facial mesh of the source model; (d) Recovered 3D positions of specified landmarks (red points) and computed reference points (green points) on the source model; (e) landmark image of the target model; (f) projected textured-mapped target model; (g) projected facial mesh of the target model; (h) Recovered 3D positions of specified landmarks (red points) on the target model (reference points are occluded).*

We then interactively specify a set of landmark points on this bitmap and the resulting image is called *landmark image* (Fig. 3 (a)). The landmarks used in our method follow the conventions laid out in [6], where we have chosen a minimum subset of landmarks according to their prominence in the face. We define the landmarks as follows: $\mathbf{p}_l^{le}$ and $\mathbf{p}_r^{le}$ are the left and right corners of the left eye, $\mathbf{p}_l^{re}$ and $\mathbf{p}_r^{re}$ are the left and right corners of the right eye, $\mathbf{p}^m$ is the center of the upper mouth contour, and $\mathbf{p}^h$ is the hairline point of the head.

Once the landmarks are all made, the texture coordinates of each vertex on $\mathcal{F}$ in the landmark image are calculated based on an orthographic projection and the landmark image is mapped to the 3D surface of $\mathcal{F}$ automatically. In order to locate all defined landmarks on the 2D image, we use a cylindrical projection to map rendered 3D face to a 2D image plane. The projection results in a 512×512 *cylindrical landmark image* (Fig. 3 (b)). As we have marked landmark points in distinct colors, we can easily detect them and calculate their image positions in the cylindrical landmark image.

To calculate 3D positions of landmark points we create a mapping from the 3D facial skin mesh of $\mathcal{F}$ to 2D image space by using the same cylindrical projection (Fig. 3 (c)). Each detected landmark point p can be located inside one of the triangles of the projected triangular mesh. The image position of p can be represented by its barycentric coordinates $\alpha = (\alpha_1, \alpha_2, \alpha_3)$ inside this triangle $\triangle(w_1, w_2, w_3)$:

$$p = \sum_{i=1}^{3} \alpha_i w_i \tag{1}$$

with $\sum_{i=1}^{3} \alpha_i = 1$ and $\alpha_i \geq 0$ for all i. Each triangle in the 2D image space corresponds to a triangle on the 3D facial surface. The 3D position of p is calculated by linear interpolation of the positions of three vertices, weighted using its barycentric coordinates in the 2D triangle:

$$P = \sum_{i=1}^{3} \alpha_i W_i \tag{2}$$

where P and W_i are the 3D positions of the landmark point p and vertex w_i that are in 2D image space, respectively.

For the target model $\mathcal{F}^*$, the corresponding set of landmark points (including $\mathbf{p}_l^{*le}, \mathbf{p}_r^{*le}, \mathbf{p}_l^{*re}, \mathbf{p}_r^{*re}$, $\mathbf{p}^{*m}$ and $\mathbf{p}^{*h}$) are defined on the acquired reflectance image (Fig. 3 (e)). By mapping this marked image to $\mathcal{F}^*$, applying cylindrical projection to texture-mapped $\mathcal{F}^*$ (Fig. 3 (f)), and then using the same position interpolation approach, we calculate the 3D positions of all landmark points that should be located on the surface of $\mathcal{F}^*$.

Given these recovered landmarks, more reference points that will be used for adaptation can be generated. Let $\mathbf{p}^{le}$, $\mathbf{p}^{re}$, and $\mathbf{p}^c$ be the 3D positions of the center of the left eye, the center of the right eye, and the center between both eyes of $\mathcal{F}$, and let $\mathbf{p}^{*le}$, $\mathbf{p}^{*re}$, and $\mathbf{p}^{*c}$ be those corresponding 3D positions on $\mathcal{F}^*$. Then, $\mathbf{p}^{le}$, $\mathbf{p}^{re}$, and $\mathbf{p}^c$ can be calculated as:

$$\mathbf{p}^{le} = \frac{1}{2}(\mathbf{p}_l^{le} + \mathbf{p}_r^{le}), \quad \mathbf{p}^{re} = \frac{1}{2}(\mathbf{p}_l^{re} + \mathbf{p}_r^{re}), \quad \mathbf{p}^{c} = \frac{1}{2}(\mathbf{p}^{le} + \mathbf{p}^{re}) \tag{3}$$

The 3D positions of $\mathbf{p}^{*le}$, $\mathbf{p}^{*re}$, and $\mathbf{p}^{*c}$ of $\mathcal{F}^*$ are calculated in the same way. Fig. 3 (d) and (h) show the recovered landmark points (red points) and reference points (green points) of $\mathcal{F}$ and $\mathcal{F}^*$ respectively.

4.2. Global Shape Adaptation

The global shape adaptation is carried out in two steps. In the first step, $\mathcal{F}^*$ is transformed such that the line through the estimated eye center positions is parallel to the x-axis of the Cartesian coordinate system and that the sagittal plane coincides with the y-z plane. Let $\mathbf{x}_i^*$ and $\mathbf{x}_i^{*'}$ be 3D positions of an vertex on $\mathcal{F}^*$ before and after the transformation, $\mathbf{R}^*$ be the rotation matrix, $\mathbf{T}^*$ be the translation vector, and $\mathbf{C}^{*0}$ be the face model center. Eq. 4 computes the transformation.

$$\mathbf{x}_i^{*'} = \mathbf{R}^*(\mathbf{x}_i^* - \mathbf{C}^{*0}) + \mathbf{T}^* \tag{4}$$

with

$$\mathbf{C}^{*0} = \mathbf{p}^{*c} \tag{5}$$

The translation vector $\mathbf{T}^*$ is estimated as follows to guarantee the y-z plane cuts through the center of the face:

$$\mathbf{T}^* = \mathbf{p}^{*c} + \mathbf{t}^* \tag{6}$$

where $\mathbf{t}^* = (t_x^*, t_y^*, t_z^*)^T$ and

$$t_x^* = -\mathbf{p}_x^{*c}, \qquad t_y^* = t_z^* = 0 \tag{7}$$

Other parameters that must be estimated in Eq. 4 are two rotation angles correspond to the face rotation around y-axis (r_y^*) and z-axis (r_z^*). These two parameters are determined by 3D positions of the left and right eye centers ($\mathbf{p}^{*le}$ and $\mathbf{p}^{*re}$). As illustrated in Fig. 4, we project $\mathbf{p}^{*le}$ and $\mathbf{p}^{*re}$ onto the 2D x-z plane using an orthographic projection. By evaluating the projected positions $\mathbf{p}^{*le|xz}$ and $\mathbf{p}^{*re|xz}$, the amount of face rotation around y-axis is calculated as

$$r_y^* = \begin{cases} \arccos\left(\dfrac{\overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}} \cdot \vec{\mathbf{n}}_x}{\|\overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}}\|}\right) & \text{if}(\overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}} \times \vec{\mathbf{n}}_x) \cdot \vec{\mathbf{n}}_y > 0 \\ -\arccos\left(\dfrac{\overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}} \cdot \vec{\mathbf{n}}_x}{\|\overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}}\|}\right) & \text{otherwise} \end{cases} \tag{8}$$

where $\overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}} = \mathbf{p}^{*le|xz} - \mathbf{p}^{*re|xz}$ using vector notation, $\vec{\mathbf{n}}_x$ and $\vec{\mathbf{n}}_y$ are unit vectors of the x- and y-axes. The magnitude of r_z^* can be obtained from Eq. 9.

$$|r_z^*| = \arccos\left(\frac{\overrightarrow{\mathbf{p}^{*re}\mathbf{p}^{*le}} \cdot \overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}}}{\|\overrightarrow{\mathbf{p}^{*re}\mathbf{p}^{*le}}\| \, \|\overrightarrow{\mathbf{p}^{*re|xz}\mathbf{p}^{*le|xz}}\|}\right) \tag{9}$$

To determine the direction of rotation around the z-axis, $\overrightarrow{\mathbf{p}^{*re}\mathbf{p}^{*le}}$ is rotated around the y-axis with computed r_y^*. Let $\mathbf{p}^{*le'}$ and $\mathbf{p}^{*re'}$ be the new positions of $\mathbf{p}^{*le}$ and $\mathbf{p}^{*re}$ after the rotation. $\overrightarrow{\mathbf{p}^{*re'}\mathbf{p}^{*le'}}$ is then in the x-y plane and

$$r_z^* = \begin{cases} |r_z^*| & \text{if}(\overrightarrow{\mathbf{p}^{*re'}\mathbf{p}^{*le'}} \times \vec{\mathbf{n}}_x) \cdot \vec{\mathbf{n}}_z > 0 \\ -|r_z| & \text{otherwise} \end{cases} \tag{10}$$

After all parameters have been calculated, $\mathcal{F}^*$ is transformed automatically according to Eq. 4.

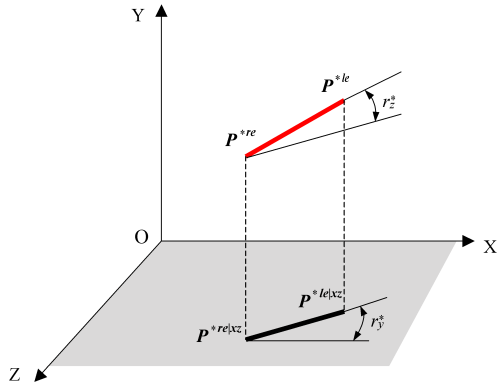


Figure 4: *Projection of the eye centers of the target model onto a 2D plane for estimation of the rotation angles. $\boldsymbol{p}^{*le}$ and $\boldsymbol{p}^{*re}$ are the left and right eye centers. $\boldsymbol{p}^{*le|xz}$ and $\boldsymbol{p}^{*re|xz}$ are the projected positions of the eye centers in the x-z plane.*

In the second step, $\mathcal{F}$ is to be scaled for matching the size of $\mathcal{F}^*$ and to be rotated and translated for matching the position of $\mathcal{F}^*$ in the 3D space. Let $\mathbf{x}_i$ and $\mathbf{x}_i'$ be the positions of an arbitrary vertex on $\mathcal{F}$ before and after the adaptation. Then, the global adaptation of $\mathcal{F}$ can be formulated as

$$\mathbf{x}_i' = \mathbf{SR}(\mathbf{x}_i - \mathbf{C}^0) + \mathbf{T} \tag{11}$$

with the translation vector $\mathbf{T}$, the face model center $\mathbf{C}^0$, the rotation matrix $\mathbf{R}$, and the scaling matrix

$$\mathbf{S} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{pmatrix} \tag{12}$$

where s_x, s_y and s_z are the scaling factors along the x-, y-, and z-axes. The face model center $\mathbf{C}^0$ is defined here as the center between both eyes of $\mathcal{F}$, i.e., $\mathbf{C}^0 = \mathbf{p}^c$.

In the rotation matrix $\mathbf{R}$, three rotation angles (r_x, r_y and r_z) must be estimated. By using the 3D eye center positions of $\mathcal{F}$, $\mathbf{p}^{le}$ and $\mathbf{p}^{re}$, the rotation angles r_y and r_z can be calculated in the same way as to calculate r_y^* and r_z^* of $\mathcal{F}^*$. The face tilt, r_x, is calculated after $\mathcal{F}$ is rotated around y- and z-axes in turn using obtained r_y and r_z. Suppose $\mathbf{p}^c$ and $\mathbf{p}^m$ move to their new positions $\mathbf{p}^{c'}$ and $\mathbf{p}^{m'}$ after the rotation, and $\mathbf{p}^{*c'}$ and $\mathbf{p}^{*m'}$ are the 3D positions of the corresponding points in the transformed $\mathcal{F}^*$. r_x is then calculated as the angle between $\overrightarrow{\mathbf{p}^{m'}\mathbf{p}^{c'}}$ and $\overrightarrow{\mathbf{p}^{*m'}\mathbf{p}^{*c'}}$.

$$r_x = \begin{cases} \arccos\left(\dfrac{\overrightarrow{\mathbf{p}^{m'}\mathbf{p}^{c'}} \cdot \overrightarrow{\mathbf{p}^{*m'}\mathbf{p}^{*c'}}}{\|\overrightarrow{\mathbf{p}^{m'}\mathbf{p}^{c'}}\| \|\overrightarrow{\mathbf{p}^{*m'}\mathbf{p}^{*c'}}\|}\right) & \text{if}(\overrightarrow{\mathbf{p}^{m'}\mathbf{p}^{c'}} \times \overrightarrow{\mathbf{p}^{*m'}\mathbf{p}^{*c'}}) \cdot \vec{\mathbf{n}}_x > 0 \\ -\arccos\left(\dfrac{\overrightarrow{\mathbf{p}^{m'}\mathbf{p}^{c'}} \cdot \overrightarrow{\mathbf{p}^{*m'}\mathbf{p}^{*c'}}}{\|\overrightarrow{\mathbf{p}^{m'}\mathbf{p}^{c'}}\| \|\overrightarrow{\mathbf{p}^{*m'}\mathbf{p}^{*c'}}\|}\right) & \text{otherwise} \end{cases} \tag{13}$$

The translation vector $\mathbf{T}$ can be expressed as

$$\mathbf{T} = \mathbf{p}^c + \mathbf{t} \tag{14}$$

where $\mathbf{t} = (t_x, t_y, t_z)^T$. It is essential that after the global adaptation, the projection of the model center of $\mathcal{F}$ in the *x*-*y* plane coincides with that of the target model $\mathcal{F}^*$. Thus, three translation parameters can be estimated as follows:

$$t_x = -\mathbf{p}_x^c, \quad t_y = \mathbf{p}_y^{*c} - \mathbf{p}_y^c, \quad t_z = 0 \tag{15}$$

The scaling factors can be estimated from the ratio of distances between a pair of landmark points as measured both in $\mathcal{F}$ and $\mathcal{F}^*$. Since we fit the position of all vertices of $\mathcal{F}$ to the scanned data $\mathcal{F}^*$ (local shape adaptation) based on a cylindrical projection, as will be demonstrated in Section 4.3, it is essential that the particular facial features such as the eyes of $\mathcal{F}$ are brought coincide with those of $\mathcal{F}^*$ after the cylindrical projection (see Fig. 5). In Fig. 5, the two half circles represent the top view of the source and target models. The large circle represents the cylindrical mapping plane onto which $\mathcal{F}$ and $\mathcal{F}^*$ are projected, and its vertical rotation axis coincides with the *y*-axis. The scaling factor s_x is estimated from the measured length between two eye centers by taking the cylindrical projection into account.

$$s_x = \frac{\| \mathbf{p}_z^c \| \| \mathbf{p}^{*le} - \mathbf{p}^{*re} \|}{\| \mathbf{p}_z^{*c} \| \| \mathbf{p}^{le} - \mathbf{p}^{re} \|} \tag{16}$$

The factors s_y and s_z are calculated as follows:

$$s_y = \frac{\| \mathbf{p}^{*c} - \mathbf{p}^{*m} \|}{\| \mathbf{p}^c - \mathbf{p}^m \|}, \quad s_z = \frac{\| \mathbf{p}_z^{*m} - \mathbf{p}_z^{*h} \|}{\| \mathbf{p}_z^m - \mathbf{p}_z^h \|} \tag{17}$$

After the required parameters have been determined, the source model is rotated, scaled and shifted to the global 3D position according to Eq. 11.

4.3. Local Shape Adaptation

The proposed local shape adaptation procedure can be regarded as an inflation process, where each vertex of $\mathcal{F}$ is shifted along a certain direction outwards until it fits to the measured data. Therefore, the shape fitting becomes a problem of how to determine the corresponding 3D position of each vertex of $\mathcal{F}$ on the surface of $\mathcal{F}^*$. In order to find this correspondence, we use a cylindrical projection to map the 3D coordinates on both $\mathcal{F}$ and $\mathcal{F}^*$ to a 2D image plane, as shown in Fig. 5.

After the projection, the image coordinates (u_i, v_i) of an arbitrary vertex $\mathbf{x}_i = (x_i, y_i, z_i)$ on $\mathcal{F}$ are calculated. The 2D projection position of each vertex on $\mathcal{F}^*$, (u_i^*, v_i^*), are also computed. For each projected vertex of $\mathcal{F}$, it can be located inside one of the triangles of the projected triangular mesh of $\mathcal{F}^*$. By using its barycentric coordinates $\beta = (\beta_1, \beta_2, \beta_3)$ inside this triangle, the 2D position of the projected vertex (u_i, v_i) can be expressed as

$$u_i = \sum_{j=1}^{3} \beta_j u_j^*, \qquad v_i = \sum_{j=1}^{3} \beta_j v_j^* \tag{18}$$

where u_j^* and v_j^* are the 2D coordinates of vertices of the projected triangle that contains (u_i, v_i).

To compute the 3D point $\mathbf{x}_i^{adapted}$ on the surface of $\mathcal{F}^*$ whose cylindrical projection is (u_i, v_i), we cast a ray from

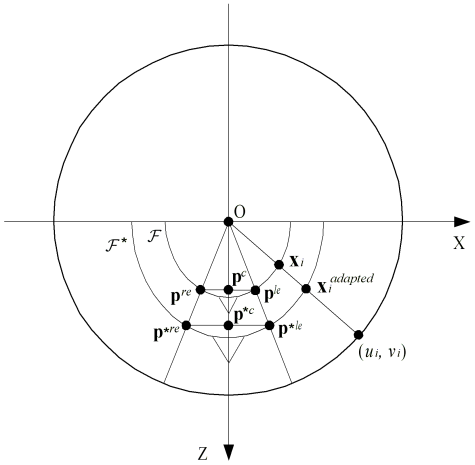


Figure 5: *Local adaptation of the source mesh to the target model using cylindrical projection.*

(u_i, v_i) on the cylinder radially back towards the cylinder's axis. The intersection between this ray and the surface of $\mathcal{F}^*$ is the point $\mathbf{x}_i^{adapted}$. Obviously, $\mathbf{x}_i^{adapted}$ is located in a triangle on $\mathcal{F}^*$ which corresponds to the triangle containing (u_i, v_i) in the 2D projection plane. The 3D position of $\mathbf{x}_i^{adapted}$ thus can be calculated as a linear interpolation of the positions of three vertices of this triangle, weighted using its barycentric coordinates in the 2D triangle:

$$\mathbf{x}_i^{adapted} = \sum_{j=1}^{3} \beta_j \mathbf{x}_j^* \tag{19}$$

where $\mathbf{x}_j^*$ are 3D positions of vertices of the triangle on $\mathcal{F}^*$.

After $\mathcal{F}$ has been locally adapted, texture image is applied to the adapted model to increase the realism. As the same image is used for texture-mapping of the target model $\mathcal{F}^*$, the texture coordinates of $\mathbf{x}_i^{adapted}$ on the adapted $\mathcal{F}$, $(s_i^{adapted}, t_i^{adapted})$, can be calculated as follows:

$$s_i^{adapted} = \sum_{j=1}^{3} \beta_j s_j^*, \qquad t_i^{adapted} = \sum_{j=1}^{3} \beta_j t_j^* \tag{20}$$

where (s_j^*, t_j^*) are the texture coordinates associated with three vertices of the triangle on $\mathcal{F}^*$ that contains $\mathbf{x}_i^{adapted}$.

5. Results

For the scanned data in Fig. 1 the presented algorithm has been performed. Fig. 6 (a) shows the adapted facial model with smooth shading. A final texture-mapped face is shown in Fig. 6 (b), where texture image is shown in Fig. 1 (b).

For the generic source model, animation can be controlled on different levels. At the lowest level, the vertices of the region influenced by an individual muscle are driven by a field of muscle force vectors to be displaced to their new positions. At the highest level, a synthesis of facial expressions is carried out by a deformation of the skin mesh resulting from the combined contraction of a certain set of facial

muscles [23]. Through our process using three-step adaptation with the generic model, the geometric shape of the model is transformed without changing the underlying animation structure, the resulting new face thus can be animated immediately. Fig. 7 shows some examples of facial animation on the adapted model with given expression parameters.

We have used our modeling approach to construct facial models for various people. Fig. 8 shows side-by-side comparisons of several reconstructed models with the real images. Females and males in different ethnic groups with various ages are reconstructed using one generic model shown in Fig. 2 being adapted to the scanned data. In our method, only the landmark location step requires manual intervention. Since the landmark points can be defined on the generic model once for its adaptation to different target faces, reconstruction of different individualized 3D models only requires the user to specify a small set of landmarks on the facial texture image. In practice, this process takes within 1 minute for reconstructing each face. After marking, the adaptation is automatically executed to generate personalized face.

6. Conclusions

In this paper, a face adaptation method for efficient reconstruction of a personalized 3D facial model for animation has been presented. The algorithm consists of three steps, namely: 1) the landmark location; 2) the global shape adaptation; and 3) the local shape adaptation. The landmark location consists of only defining a few corresponding landmark points on the 2D images and recovering their 3D positions. In the second step, the global shape adaptation transforms the prototype source model to align it with the target model in the 3D space using the transformation parameters estimated from measurements between recovered 3D landmark points. In the final local shape adaptation, the positions of all vertices of the source model are fitted to the target model by means of the cylindrical projection. Once the skin geometry is adapted, the facial texture and the underlying muscle structure are automatically transferred, such that the resulting model portrays the geometry and color of the individual face and remains completely animatable.

While the method is efficient enough for personalized facial model reconstruction, it still involves manual interaction in the landmark location process. We will further automate this process by using the vision face recognition approach to automatically estimate facial features from an image. The reconstructed facial model currently uses the texture extracted from a reflectance image with relatively low resolution acquired from the range scanner. Using a separate RGB camera of high resolution for texture acquisition will definitely enhance the photorealism of the textured model. To make our adaptation-based modeling technique also effective to the incomplete data, we would like to develop more sophisticated method to fit the prototype model to the poor scanned data without fixing such as hole filling.

Acknowledgements

The authors wish to thank the support of the Institute of Engineering Science, NUS, as well as acknowledge the grant provided by research project R252-000-146-112.

References

1. T. Akimoto, Y. Suenaga, and R. S. Wallace. "Automatic creation of 3D facial models." *IEEE Computer Graphics and Application*, 13(5):16-22, September 1993. 2
2. P. Cignoni, C. Montani, and R. Scopigno. "A comparison of mesh simplification algorithms." *Computer & Graphics*, 22(1): 37-54, 1998. 1
3. D. DeCarlo, D. Metaxas, and M. Stone. "An anthropometric face model using variational techniques." *Proc. SIGGRAPH'98*, pp. 67-74, July 1998. 2
4. P. Ekman and W. V. Friesen. *Facial Action Coding System*. Consulting Psychologists Press Inc., California, 1978. 3
5. R. Enciso, J. Li, D. Fidaleo, T-Y. Kim, J-Y. Noh, and U. Neumann. "Synthesis of 3D faces." *International Workshop on Digital and Computational Video (DCV'99)*, 1999. 2
6. L. G. Farkas. *Anthropometry of the Head and Face*, 2nd ed. Raven Press, 1994. 2, 4
7. P. Fua. "From multiple stereo views to multiple 3D surfaces." *International Journal of Computer Vision*, 24(1): 19-35, 1997. 2
8. Horace H. S. Ip, Lijun Yin. "Constructing a 3D individualized head model from two orthogonal views." *The Visual Computer*, 12:254-266, 1996. 2
9. K. Kähler, J. Haber, H. Yamauchi, and H. P. Seidel. "Head shop: Generating animated head models with anatomical structure." *Proc. ACM SIGGRAPH Symposium on Computer Animation*, pp. 55-64, 2002. 2
10. T. Kurihara and K. Arai. "A transformation method for modeling and animation of the human face from photographs." *Proc. Computer Animation'91*, pp. 45-58, 1991. 2
11. A. LeBlanc, P. Kalar, N. Magnenat-Thalmann, and D. Thalmann. "Sculpting with the 'ball and mouse' metaphor." *Proc. Graphics Interface'91*, pp. 152-159, 1991. 2
12. Y. Lee, D. Terzopoulis, and K. Waters. "Realistic modeling for facial animation." *Proc. SIGGRAPH'95*, pp. 55-62, August 1995. 2
13. W. S. Lee and N. Magnenat-Thalmann. "Fast head modeling for animation." *Journal Image and Vision Computing*, 18(4): 355-364, March 2000. 2
14. N. Magnenat-Thalmann and D. Thalmann. "The direction of synthetic actors in the film Rendez-vous *à* Montrèal." *IEEE Computer Graphics and Application*, 7(12): 9-19, 1987. 2
15. L. Moccozet and N. Magnenat-Thalmann. "Dirichlet free-from deformation and their application to hand simulation." *Proc. Computer Animation'97*, pp. 93-102, 1997. 2
16. F. I. Parke. *Computer generated animation of faces*. Master's thesis, University of Utah, Salt Lake City, 1972. 1
17. F. I. Parke. "Parameterized models for facial animation." *IEEE Computer Graphics and Application*, 2(9): 61-68, November 1982. 1, 2

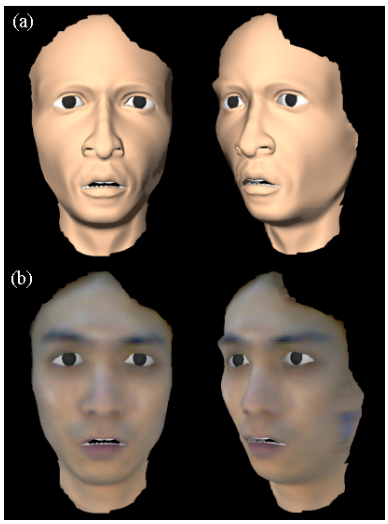


Figure 6: *Adapted personalized face model with (a) Gouraud shading and (b) texture mapping (two views of each).*

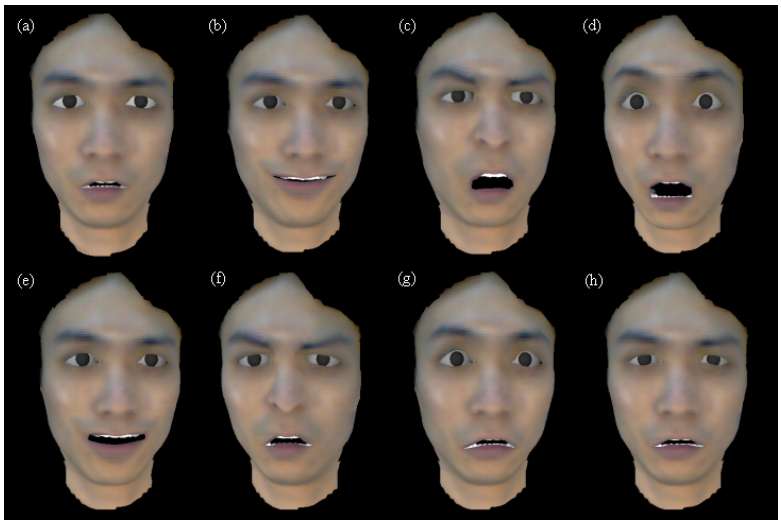


Figure 7: *Several expressions synthesized on the adapted facial model: (a) neutral expression; (b) happiness; (c) shouting; (d) surprise; (e) laughing; (f) anger; (g) fear; (h) sadness.*

Figure 8: *Reconstructed 3D facial models of various people. In each example, two original facial images (left) are compared with the reconstructed face (right) which can be animated immediately.*

18. F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. H. Salesin. “Synthesizing realistic facial expressions from photographs.” *Proc. SIGGRAPH’98*, pp. 75-84, July 1998. 2

19. M. Proesmans and L. V. Gool. “Reading between the lines - a method for extracting dynamic 3D with texture.” *Proc. VRST’97*, pp. 95-102, 1997. 2

20. D. Terzopoulus and K. Waters. “Physically-based facial modeling, analysis and animation.” *Journal of Visualization and Computer Animation* , vol.1, pp. 73-80, 1990. 1

21. F. Ulgen. “A step toward universal facial animation via volume morphing.” *Proc. 6th IEEE Interantional Workshop on Robot and Human Communication*, pp. 358-363, 1997. 2

22. L. Yin. “A fast feature detection algorithm for human face contour based on local maximum curvature tracking.” *Technique Report*, ICG, Department of Computing Science, City University of HK, 1995. 2

23. Y. Zhang, E. C. Prakash and E. Sung. “A physically-based model with adaptive refinement for facial animation." *Proc. IEEE Computer Animation 2001*, pp. 28-39, 2001. 3, 7