Realistic Modeling for Facial Animation

Yuencheng Lee[1], Demetri Terzopoulos[1], and Keith Waters[2]
University of Toronto[1] and Digital Equipment Corporation[2]

Abstract

A major unsolved problem in computer graphics is the construction and animation of realistic human facial models. Traditionally, facial models have been built painstakingly by manual digitization and animated by ad hoc parametrically controlled facial mesh deformations or kinematic approximation of muscle actions. Fortunately, animators are now able to digitize facial geometries through the use of scanning range sensors and animate them through the dynamic simulation of facial tissues and muscles. However, these techniques require considerable user input to construct facial models of individuals suitable for animation. In this paper, we present a methodology for automating this challenging task. Starting with a structured facial mesh, we develop algorithms that automatically construct functional models of the heads of human subjects from laser-scanned range and reflectance data. These algorithms automatically insert contractile muscles at anatomically correct positions within a dynamic skin model and root them in an estimated skull structure with a hinged jaw. They also synthesize functional eyes, eyelids, teeth, and a neck and fit them to the final model. The constructed face may be animated via muscle actuations. In this way, we create the most authentic and functional facial models of individuals available to date and demonstrate their use in facial animation.

CR Categories: I.3.5 **[Computer Graphics]:** Physically based modeling; I.3.7 **[Computer Graphics]:** Animation.

Additional Keywords: Physics-based Facial Modeling, Facial Animation, RGB/Range Scanners, Feature-Based Facial Adaptation, Texture Mapping, Discrete Deformable Models.

1 Introduction

Two decades have passed since Parke's pioneering work in animating faces [13]. In the span of time, significant effort has been devoted to the development of computational models of the human face for applications in such diverse areas as entertainment, low bandwidth teleconferencing, surgical facial planning, and virtual reality. However, the task of accurately modeling the expressive human face by computer remains a major challenge.

Traditionally, computer facial animation follows three basic procedures: (1) design a 3D facial mesh, (2) digitize the 3D mesh, and (3) animate the 3D mesh in a controlled fashion to simulate facial actions.

In procedure (1), it is desirable to have a refined topological mesh that captures the facial geometry. Often this entails digitizing as many nodes as possible. Care must be taken not to oversample the surface because there is a trade-off between the number of nodes and the computational cost of the model. Consequently, meshes developed to date capture the salient features of the face with as few nodes as possible (see [17, 14, 21, 9, 23] for several different mesh designs).

In procedure (2), a general 3D digitization technique uses photogrammetry of several images of the face taken from different angles. A common technique is to place markers on the face that can be seen from two or more cameras. An alternative technique is to manually digitize a plaster cast of the face using manual 3D digitization devices such as orthogonal magnetic fields sound captors [9], or one to two photographs [9, 7, 1]. More recently, automated laser range finders can digitize on the order of 10^5 3D points from a solid object such as a person's head and shoulders in just a few seconds [23].

In procedure (3), an animator must decide which mesh nodes to articulate and how much they should be displaced in order to produce a specific facial expression. Various approaches have been proposed for deforming a facial mesh to produce facial expressions; for example, parameterized models [14, 15], control-point models [12, 7], kinematic muscle models [21, 9], a texture-map-assembly model [25], a spline model [11], feature-tracking models [24, 16], a finite element model [6], and dynamic muscle models [17, 20, 8, 3].

1.1 Our Approach

The goal of our work is to automate the challenging task of creating realistic facial models of individuals suitable for animation. We develop an algorithm that begins with cylindrical range and reflectance data acquired by a Cyberware scanner and automatically constructs an efficient and fully functional model of the subject's head, as shown in Plate 1. The algorithm is applicable to various individuals (Plate 2 shows the raw scans of several individuals). It proceeds in two steps:

In step 1, the algorithm adapts a well-structured face mesh from [21] to the range and reflectance data acquired by scanning the subject, thereby capturing the shape of the subject's face. This approach has significant advantages because it avoids repeated manual modification of control parameters to compensate for geometric variations in the facial features from person to person. More specifically, it allows the automatic placement of facial muscles and enables the use of a single control process across different facial models.

The generic face mesh is adapted automatically through an image analysis technique that searches for salient local minima and maxima in the range image of the subject. The search is directed according to the known relative positions of the nose, eyes, chin, ears, and other facial features with respect to the generic mesh. Facial muscle emergence and attachment points are also known relative to the generic mesh and are adapted automatically as the mesh is conformed to the scanned data.

In step 2, the algorithm elaborates the geometric model constructed in step 1 into a functional, physics-based model of the subject's face which is capable of facial expression, as shown in the lower portion of Plate 1.

We follow the physics-based facial modeling approach proposed

[1]Department of Computer Science, 10 King's College Road, Toronto, ON, Canada, M5S 1A4. {vlee | dt}@cs.toronto.edu
[2]Cambridge Research Lab., One Kendall Square, Cambridge, MA 02139. waters@crl.dec.com

Published in the Proceedings of SIGGRAPH 95 (Los Angeles, CA, August, 1995). In *Computer Graphics* Proceedings, Annual Conference Series, 1995, ACM SIGGRAPH, pp. 55–62.

by Terzopoulos and Waters [20]. Its basic features are that it animates facial expressions by contracting synthetic muscles embeded in an anatomically motivated model of skin composed of three spring-mass layers. The physical simulation propagates the muscle forces through the physics-based synthetic skin thereby deforming the skin to produce facial expressions. Among the advantages of the physics-based approach are that it greatly enhances the degree of realism over purely geometric facial modeling approaches, while reducing the amount of work that must be done by the animator. It can be computationally efficient. It is also amenable to improvement, with an increase in computational expense, through the use of more sophisticated biomechanical models and more accurate numerical simulation methods.

We propose a more accurate biomechanical model for facial animation compared to previous models. We develop a new biomechanical facial skin model which is simpler and better than the one proposed in [20]. Furthermore, we argue that the skull is an important biomechanical structure with regard to facial expression [22]. To date, the skin-skull interface has been underemphasized in facial animation despite its importance in the vicinity of the articulate jaw; therefore we improve upon previous facial models by developing an algorithm to estimate the skull structure from the acquired range data, and prevent the synthesized facial skin from penetrating the skull.

Finally, our algorithm includes an articulated neck and synthesizes subsidiary organs, including eyes, eyelids, and teeth, which cannot be adequately imaged or resolved in the scanned data, but which are nonetheless crucial for realistic facial animation.

2 Generic Face Mesh and Mesh Adaptation

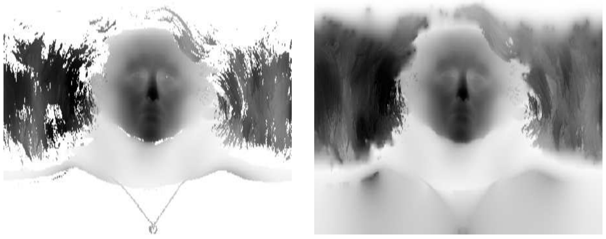
The first step of our approach to constructing functional facial models of individuals is to scan a subject using a Cyberware Color Digitizer™. The scanner rotates 360 degrees around the subject, who sits motionless on a stool as a laser stripe is projected onto the head and shoulders. Once the scan is complete, the device has acquired two registered images of the subject: a range image (Figure 1) — a topographic map that records the distance from the sensor to points on the facial surface, and a reflectance (RGB) image (Figure 2) — which registers the color of the surface at those points. The images are in cylindrical coordinates, with longitude (0–360) degrees along the x axis and vertical height along the y axis. The resolution of the images is typically 512×256 pixels (cf. Plate 1)

The remainder of this section describes an algorithm which reduces the acquired geometric and photometric data to an efficient geometric model of the subject's head. The algorithm is a two-part process which repairs defects in the acquired images and conforms a generic facial mesh to the processed images using a feature-based matching scheme. The resulting mesh captures the facial geometry as a polygonal surface that can be texture mapped with the full resolution reflectance image, thereby maintaining a realistic facsimile of the subject's face.

2.1 Image Processing

One of the problems of range data digitization is illustrated in Figure 1(a). In the hair area, in the chin area, nostril area, and even in the pupils, laser beams tend to disperse and the sensor observes no range value for these corresponding 3D surface points. We must correct for missing range and texture information.

We use a *relaxation method* to interpolate the range data. In particular, we apply a membrane interpolation method described in [18]. The relaxation interpolates values for the missing points so as to bring them into successively closer agreement with surrounding points by repeatedly indexing nearest neighbor values. Intuitively, it stretches an elastic membrane over the gaps in the surface. The images interpolated through relaxation are shown in Figure 1(b) and illustrate improvements in the hair area and chin area. Relaxation works effectively when the range surface is smooth, and particularly in the case of human head range data, the smoothness requirement of the solutions is satisfied quite effectively.



(a) (b)

Figure 1: (a) Range data of "Grace" from a Cyberware scanner. (b) Recovered plain data.

Figure 2(a) shows two 512×256 reflectance (RGB) texture maps as monochrome images. Each reflectance value represents the surface color of the object in cylindrical coordinates with corresponding longitude (0–360 degrees) and latitude. Like range images, the acquired reflectance images are lacking color information at certain points. This situation is especially obvious in the hair area and the shoulder area (see Figure 2(a)). We employ the membrane relaxation approach to interpolate the texture image by repeated averaging of neighboring known colors. The original texture image in Figure 2(a) can be compared with the interpolated texture image in Figure 2(b).

(a) (b)

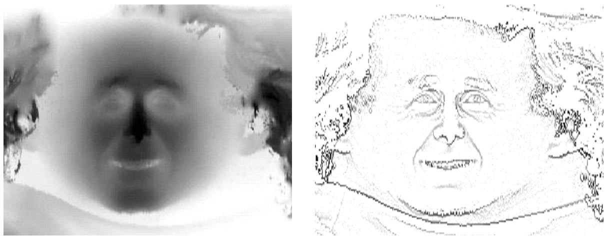
Figure 2: (a) Texture data of "George" with void points displayed in white and (b) texture image interpolated using relaxation method.

The method is somewhat problematic in the hair area where range variations may be large and there is a relatively high percentage of missing surface points. A thin-plate relaxation algorithm [18] may be more effective in these regions because it would fill in the larger gaps with less "flattening" than a membrane [10].

Although the head structure in the cylindrical laser range data is distorted along the longitudinal direction, important features such as the slope changes of the nose, forehead, chin, and the contours of the mouth, eyes, and nose are still discernible. In order to locate the contours of those facial features for use in adaptation (see below), we use a modified Laplacian operator (applied to the discrete image through local pixel differencing) to detect edges from the range map shown in Figure 3(a) and produce the field function in Fig. 3(b). For details about the operator, see [8]. The field function highlights important features of interest. For example, the local maxima of the modified Laplacian reveals the boundaries of the lips, eyes, and chin.

2.2 Generic Face Mesh and Mesh Adaptation

The next step is to reduce the large arrays of data acquired by the scanner into a parsimonious geometric model of the face that can eventually be animated efficiently. Motivated by the adaptive meshing techniques [19] that were employed in [23], we significantly



(a) (b)

Figure 3: (a) Original range map. (b) Modified Laplacian field function of (a).

improved the technique by adapting a generic face mesh to the data. Figure 4 shows the planar generic mesh which we obtain through a cylindrical projection of the 3D face mesh from [21]. One of the advantages of the generic mesh is that it has well-defined features which form the basis for accurate feature based adaptation to the scanned data and automatic scaling and positioning of facial muscles as the mesh is deformed to fit the images. Another advantage is that it automatically produces an efficient triangulation, with finer triangles over the highly curved and/or highly articulate regions of the face, such as the eyes and mouth, and larger triangles elsewhere.

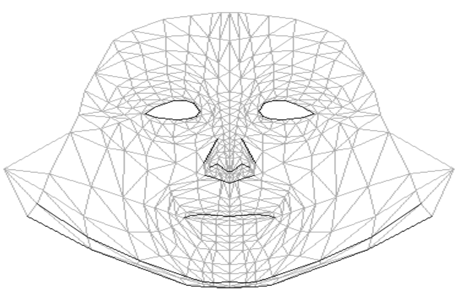


Figure 4: Facial portion of generic mesh in 2D cylindrical coordinates. Dark lines are features for adaptation.

We label all facial feature nodes in the generic face prior to the adaptation step. The feature nodes include eye contours, nose contours, mouth contours, and chin contours.

For any specific range image and its positive Laplacian field function (Figure 3), the generic mesh adaptation procedure performs the following steps to locate feature points in the range data (see [8] for details):

Mesh Adaptation Procedures

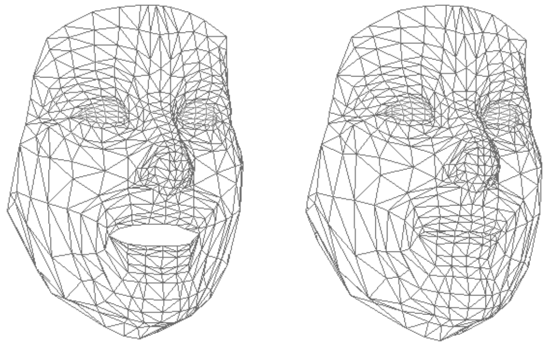
1. Locate nose tip
2. Locate chin tip
3. Locate mouth contour
4. Locate chin contour
5. Locate ears
6. Locate eyes
7. Activate spring forces
8. Adapt hair mesh
9. Adapt body mesh
10. Store texture coordinates

Once the mesh has been fitted by the above feature based matching technique (see Plate 3), the algorithm samples the range image at the location of the nodes of the face mesh to capture the facial geometry, as is illustrated in Figure 5.

The node positions also provide texture map coordinates that are used to map the full resolution color image onto the triangles (see Plate 3).

2.3 Estimation of Relaxed Face Model

Ideally, the subject's face should be in a neutral, relaxed expression when he or she is being scanned. However, the scanned woman in



(a) (b)

Figure 5: (a) Generic geometric model conformed to Cyberware scan of "Heidi". (b) Same as (a). Note that "Heidi's" mouth is now closed, subsequent to estimation of the relaxed face geometry.

the "Heidi" dataset is smiling and her mouth is open (see Plate 2). We have made our algorithm tolerant of these situations. To construct a functional model, it is important to first estimate the relaxed geometry. That is, we must infer what the "Heidi" subject would look like had her face been in a relaxed pose while she was being scanned. We therefore estimate the range values of the closed mouth contour from the range values of the open mouth contour by the following steps:

1. Perform adaptation procedures in Sec. 2.2 without step 3.
2. Store nodal longitude/latitude into adapted face model.
3. Perform lip adaptation in step 3 in sec. 2.2
4. Store nodal range values into adapted face model.

As a result, the final reconstructed face model in Figure 5(b) will have a relaxed mouth because the longitude and latitude recorded is the default shape of our closed mouth model (see Figure 4). Moreover, the shape of the final reconstructed face is still faithful to the head data because the range value at each facial nodal point is obtained correctly after the lip adaptation procedure has been performed. Relaxing the face shown in Figure 5(a) results in the image in Figure 5(b) (with eyelids inserted — see below).

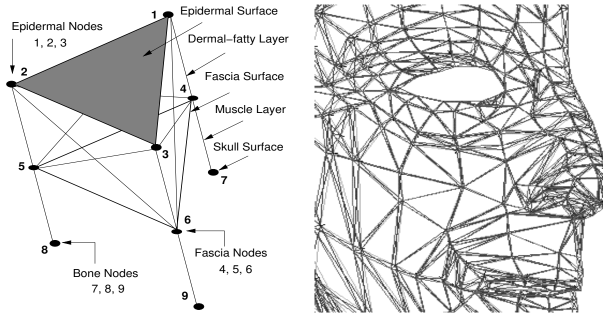
3 The Dynamic Skin and Muscle Model

This section describes how our system proceeds with the construction of a fully functional model of the subject's face from the facial mesh produced by the adaptation algorithm described in the previous section. To this end, we automatically create a dynamic model of facial tissue, estimate a skull surface, and insert the major muscles of facial expression into the model. The following sections describe each of these components. We also describe our high-performance parallel, numerical simulation of the dynamic facial tissue model.

3.1 Layered Synthetic Tissue Model

The skull is covered by deformable tissue which has five distinct layers [4]. Four layers—epidermis, dermis, sub-cutaneous connective tissue, and fascia—comprise the skin, and the fifth consists of the muscles of facial expression. Following [20], and in accordance with the structure of real skin [5], we have designed a new, synthetic tissue model (Figure 6(a)).

The tissue model is composed of triangular prism elements (see Figure 6(a)) which match the triangles in the adapted facial mesh. The epidermal surface is defined by nodes 1, 2, and 3, which are connected by epidermal springs. The epidermis nodes are also connected by dermal-fatty layer springs to nodes 4, 5, and 6, which define the fascia surface. Fascia nodes are interconnected by fascia



(a) (b)

Figure 6: (a) Triangular skin tissue prism element. (b) Close-up view of right side of an individual with conformed elements.

springs. They are also connected by muscle layer springs to skull surface nodes 7, 8, 9.

Figure 9(b) shows 684 such skin elements assembled into an extended skin patch. Several synthetic muscles are embedded into the muscle layer of the skin patch and the figure shows the skin deformation due to muscle contraction. Muscles are fixed in an estimated bony subsurface at their point of emergence and are attached to fascia nodes as they run through several tissue elements. Figure 6(b) shows a close-up view of the right half of the facial tissue model adapted to an individual's face which consists of 432 elements.

3.2 Discrete Deformable Models (DDMs)

A discrete deformable model has a node-spring-node structure, which is a uniaxial finite element. The data structure for the node consists of the nodal mass m_i, position $\mathbf{x}_i(t) = [x_i(t), y_i(t), z_i(t)]'$, velocity $\mathbf{v}_i = d\mathbf{x}_i/dt$, acceleration $\mathbf{a}_i = d^2\mathbf{x}_i/dt^2$, and net nodal forces $\mathbf{f}_i^{\mathbf{n}}(t)$. The data structure for the spring in this DDM consists of pointers to the head node i and the tail node j which the spring interconnects, the natural or rest length l_k of the spring, and the spring stiffness c_k.

3.3 Tissue Model Spring Forces

By assembling the discrete deformable model according to histological knowledge of skin (see Figure 6(a)), we are able to construct an anatomically consistent, albeit simplified, tissue model. Figure 6(b) shows a close-up view of the tissue model around its eye and nose parts of a face which is automatically assembled by following the above approach.

- The force spring j exerts on node i is

  $\mathbf{g}_j = c_j(l_j - l_j^r)\mathbf{s}_j$

  - each layer has its own stress-strain relationship c_j and the dermal-fatty layer uses biphasic springs (non-constant c_j) [20]
  - l_j^r and $l_j = ||\mathbf{x}_j - \mathbf{x}_i||$ are the rest and current lengths for spring j
  - $\mathbf{s}_j = (\mathbf{x}_j - \mathbf{x}_i)/l_j$ is the spring direction vector for spring j

3.4 Linear Muscle Forces

The muscles of facial expression, or the muscular plate, spreads out below the facial tissue. The facial musculature is attached to the skin tissue by short elastic tendons at many places in the fascia, but is fixed to the facial skeleton only at a few points. Contractions of the facial muscles cause movement of the facial tissue. We model 28 of the primary facial muscles, including the zygomatic major and minor, frontalis, nasii, corrugator, mentalis, buccinator, and angulii depressor groups. Plate 4 illustrates the effects of automatic scaling and positioning of facial muscle vectors as the generic mesh adapts to different faces.

To better emulate the facial muscle attachments to the fascia layer in our model, a group of fascia nodes situated along the muscle path—i.e., within a predetermined distance from a central muscle vector, in accordance with the muscle width—experience forces from the contraction of the muscle. The face construction algorithm determines the nodes affected by each muscle in a precomputation step.

To apply muscle forces to the fascia nodes, we calculate a force for each node by multiplying the muscle vector with a force length scaling factor and a force width scaling factor (see Figure 7(a)). Function Θ_1 (Figure 8(a)) scales the muscle force according to the length ratio $\varepsilon_{j,i}$, while Θ_2 (Figure 8(b)) scales it according to the width $\omega_{j,i}$ at node i of muscle j:

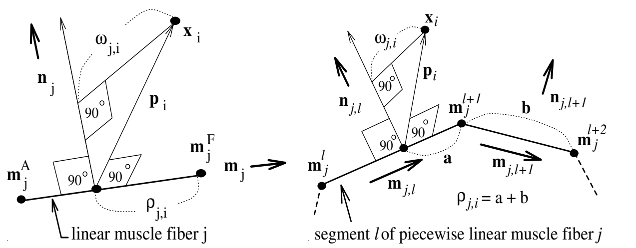
$$\begin{aligned} \varepsilon_{j,i} &= ((\mathbf{m}_j^F - \mathbf{x}_i) \cdot \mathbf{m}_j)/(\|\mathbf{m}_j^A - \mathbf{m}_j^F\|) \\ \omega_{j,i} &= \|\mathbf{p}_i - (\mathbf{p}_i \cdot \mathbf{n}_j)\mathbf{n}_j\| \end{aligned}$$

- The force muscle j exerts on node i is

  $\mathbf{f}_i^j = \Theta_1(\varepsilon_{j,i})\Theta_2(\omega_{j,i})\mathbf{m}_j$

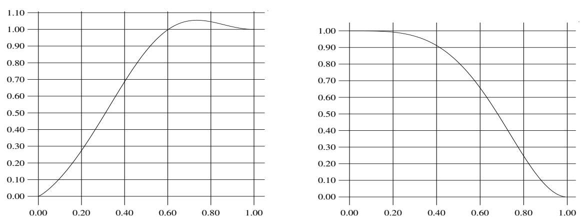
  - Θ_1 scales the force according to the distance ratio $\varepsilon_{j,i}$, where $\varepsilon_{j,i} = \rho_{j,i}/d_j$, with d_j the muscle j length.
  - Θ_2 scales the force according to the width ratio $\omega_{j,i}/w_j$, with w_j the muscle j width.
  - $\mathbf{m}_j$ is the normalized muscle vector for muscle j

Note that the muscle force is scaled to zero at the root of the muscle fiber in the bone and reaches its full strength near the end of the muscle fiber. Figure 9(b) shows an example of the effect of muscle forces applied to a synthetic skin patch.



(a) (b)

Figure 7: (a) Linear muscle fiber. (b) Piecewise linear muscle fiber.



(a) (b)

Figure 8: (a) Muscle force scaling function Θ_1 wrt $\varepsilon_{j,i}$, (b) Muscle force scaling function Θ_2 wrt $\omega_{j,i}/w_j$

3.5 Piecewise Linear Muscle Forces

In addition to using linear muscle fibers in section 3.4 to simulate sheet facial muscles like the frontalis and the zygomatics, we also model sphincter muscles, such as the orbicularis oris circling the mouth, by generalizing the linear muscle fibers to be piecewise linear and allowing them to attach to fascia at each end of the segments. Figure 7(b) illustrates two segments of an N-segment piecewise linear muscle j showing three nodes $\mathbf{m}_j^l$, $\mathbf{m}_j^{l+1}$, and $\mathbf{m}_j^{l+2}$. The unit vectors $\mathbf{m}_{j,l}$, $\mathbf{m}_{j,l+1}$ and $\mathbf{n}_{j,l}$, $\mathbf{n}_{j,l+1}$ are parallel and normal to the segments, respectively. The figure indicates fascia node i at $\mathbf{x}_i$, as well as the distance $\rho_{j,i} = a + b$, the width $\omega_{j,i}$, and the perpendicular vector $\mathbf{p}_i$ from fascia node i to the nearest segment of the muscle. The length ratio $\varepsilon_{j,i}$ for fascia node i in muscle fiber j is

$$\varepsilon_{j,i} = \frac{(\mathbf{m}_j^{l+1} - \mathbf{x}_i) \cdot \mathbf{m}_{j,l} + \sum_{k=l+1}^{N} \parallel \mathbf{m}_j^{k+1} - \mathbf{m}_j^k \parallel}{\sum_{k=1}^{N} \parallel \mathbf{m}_j^{k+1} - \mathbf{m}_j^k \parallel}$$

The width $\omega_{j,i}$ calculation is the same as for linear muscles. The remaining muscle force computations are the same as in section 3.4. Plate 4 shows all the linear muscles and the piecewise linear sphincter muscles around the mouth.

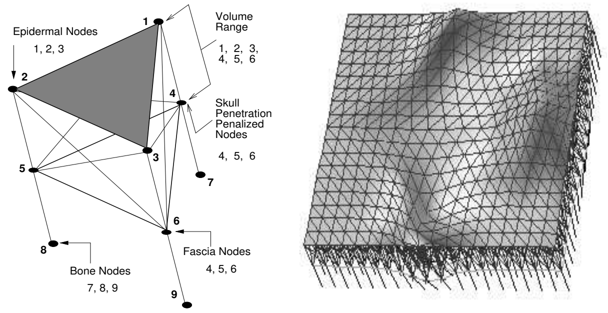
3.6 Volume Preservation Forces

In order to faithfully exhibit the incompressibility [2] of real human skin in our model, a volume constraint force based on the change of volume (see Figure 9(a)) and displacements of nodes is calculated and applied to nodes. In Figure 9(b) the expected effect of volume preservation is demonstrated. For example, near the origin of the muscle fiber, the epidermal skin is bulging out, and near the end of the muscle fiber, the epidermal skin is depressed.

- The volume preservation force element e exerts on nodes i in element e is

  $$\mathbf{q}_i^e = k_1(V^e - \tilde{V}^e)\mathbf{n}_i^e + k_2(\mathbf{p}_i^e - \tilde{\mathbf{p}}_i^e)$$

  - $\tilde{V}^e$ and V^e are the rest and current volumes for e
  - $\mathbf{n}_i^e$ is the epidermal normal for epidermal node i
  - $\tilde{\mathbf{p}}_i^e$ and $\mathbf{p}_i^e$ are the rest and current nodal coordinates for node i with respect to the center of mass of e
  - k_1, k_2 are force scaling constants



(a) (b)

Figure 9: (a) Volume preservation and skull nonpenetration element. (b) Assembled layered tissue elements under multiple muscle forces.

3.7 Skull Penetration Constraint Forces

Because of the underlying impenetrable skull of a human head, the facial tissue during a facial expression will slide over the underlying bony structure. With this in mind, for each individual's face model reconstructed from the laser range data, we estimate the skull surface normals to be the surface normals in the range data image. The skull is then computed as an offset surface. To prevent nodes from penetrating the estimated skull (see Figure 9(a)), we apply a skull non-penetration constraint to cancel out the force component on the fascia node which points into the skull; therefore, the resulting force will make the nodes slide over the skull.

- The force to penalize fascia node i during motion is:

  $$\mathbf{s}_i = \begin{cases} -(\mathbf{f}_i^n \cdot \mathbf{n}_i)\mathbf{n}_i & \text{when } \mathbf{f}_i^n \cdot \mathbf{n}_i < 0 \\ \mathbf{0} & \text{otherwise} \end{cases}$$

  - $\mathbf{f}_i^n$ is the net force on fascia node i
  - $\mathbf{n}_i$ is the nodal normal of node i

3.8 Equations of Motion for Tissue Model

Newton's law of motion governs the response of the tissue model to forces. This leads to a system of coupled second order ODEs that relate the node positions, velocities, and accelerations to the nodal forces. The equation for node i is

$$m_i \frac{d^2\mathbf{x}_i}{dt^2} + \gamma_i \frac{d\mathbf{x}_i}{dt} + \tilde{\mathbf{g}}_i + \tilde{\mathbf{q}}_i + \tilde{\mathbf{s}}_i + \tilde{\mathbf{h}}_i = \tilde{\mathbf{f}}_i$$

- m_i is the nodal mass,
- γ_i is the damping coefficient,
- $\tilde{\mathbf{g}}_i$ is the total spring force at node i,
- $\tilde{\mathbf{q}}_i$ is the total volume preservation force at node i,
- $\tilde{\mathbf{s}}_i$ is the total skull penetration force at node i,
- $\tilde{\mathbf{h}}_i$ is the total nodal restoration force at node i,
- $\tilde{\mathbf{f}}_i$ is the total applied muscle force at node i,

3.9 Numerical Simulation

The solution to the above system of ODEs is approximated by using the well-known, explicit Euler method. At each iteration, the nodal acceleration at time t is computed by dividing the net force by nodal mass. The nodal velocity is then calculated by integrating once, and another integration is done to compute the nodal positions at the next time step $t + \Delta t$, as follows:

$$\begin{aligned} \mathbf{a}_i^t &= \frac{1}{m_i}(\tilde{\mathbf{f}}_i^t - \gamma_i \mathbf{v}_i^t - \tilde{\mathbf{g}}_i^t - \tilde{\mathbf{q}}_i^t - \tilde{\mathbf{s}}_i^t - \tilde{\mathbf{h}}_i^t) \\ \mathbf{v}_i^{t+\Delta t} &= \mathbf{v}_i^t + \Delta t \mathbf{a}_i^t \\ \mathbf{x}_i^{t+\Delta t} &= \mathbf{x}_i^t + \Delta t \mathbf{v}_i^{t+\Delta t} \end{aligned}$$

3.10 Default Parameters

The default parameters for the physical/numerical simulation and the spring stiffness values of different layers are as follows:

Mass (m)	Time step (Δt)	Damping (γ)
0.5	0.01	30

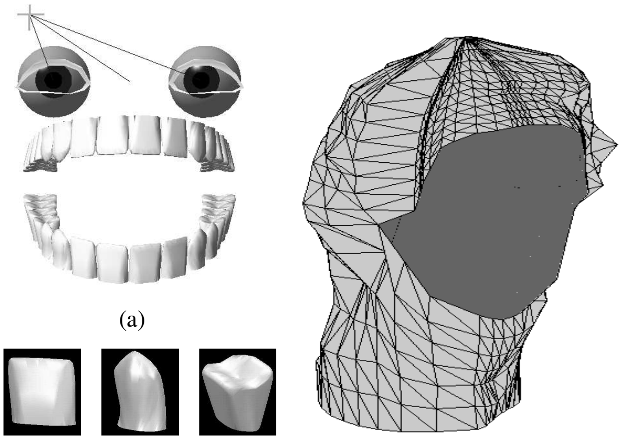
	Epid	Derm-fat 1	Derm-fat 2	Fascia	Muscle
c	60	30	70	80	10

3.11 Parallel Processing for Facial Animation

The explicit Euler method allows us to easily carry out the numerical simulation of the dynamic skin/muscle model in parallel. This is because at each time step all the calculations are based on the results from the previous time step. Therefore, parallelization is achieved by evenly distributing calculations at each time step to all available processors. This parallel approach increases the animation speed to allow us to simulate facial expressions at interactive rates on our Silicon Graphics multiprocessor workstation.

4 Geometry Models for Other Head Components

To complete our physics-based face model, additional geometric models are combined along with the skin/muscle/skull models developed in the previous section. These include the eyes, eyelids, teeth, neck, hair, and bust (Figure 10). See Plate 5 for an example of a complete model.



(a)

(b) (c)

Figure 10: (a) Geometric models of eyes, eyelids, and teeth (b) Incisor, canine, and molar teeth. (c) hair and neck.

4.1 Eyes

Eyes are constructed from spheres with adjustable irises and adjustable pupils (Figure 10(a)). The eyes are automatically scaled to fit the facial model and are positioned into it. The eyes rotate kinematically in a coordinated fashion so that they will always converge on a specified fixation point in three-dimensional space that defines the field of view. Through a simple illumination computation, the eyes can automatically dilate and contract the pupil size in accordance with the amount of light entering the eye.

4.2 Eyelids

The eyelids are polygonal models which can blink kinematically during animation (see Figure 10(a)). Note that the eyelids are open in Figure 10(a).

If the subject is scanned with open eyes, the sensor will not observe the eyelid texture. An eyelid texture is synthesized by a relaxation based interpolation algorithm similar to the one described in section 2.1. The relaxation algorithm interpolates a suitable eyelid texture from the immediately surrounding texture map. Figure 11 shows the results of the eyelid texture interpolation.

(a) (b)

Figure 11: (a) Face texture image with adapted mesh before eyelid texture synthesis (b) after eyelid texture synthesis.

4.3 Teeth

We have constructed a full set of generic teeth based on dental images. Each tooth is a NURBS surfaces of degree 2. Three different teeth shapes, the incisor, canine, and molar, are modeled (Figure 10(b)). We use different orientations and scalings of these basic shapes to model the full set of upper and lower teeth shown in Figure 10(a). The dentures are automatically scaled to fit in length, curvature, etc., and are positioned behind the mouth of the facial model.

4.4 Hair, Neck, and Bust Geometry

The hair and bust are both rigid polygonal models (see Figure 10(c)). They are modeled from the range data directly, by extending the facial mesh in a predetermined fashion to the boundaries of the range and reflectance data, and sampling the images as before.

The neck can be twisted, bent and rotated with three degrees of freedom. See Figure 12 for illustrations of the possible neck articulations.

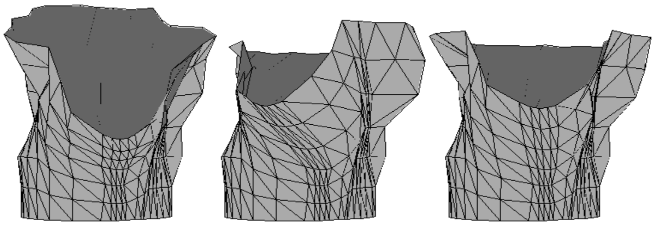


Figure 12: articulation of neck.

5 Animation Examples

Plate 1 illustrates several examples of animating the physics-based face model after conformation to the "Heidi" scanned data (see Plate 2).

- The *surprise* expression results from contraction of the outer frontalis, major frontalis, inner frontalis, zygomatics major, zygomatics minor, depressor labii, and mentalis, and rotation of the jaw.
- The *anger* expression results from contraction of the corrugator, lateral corrugator, levator labii, levator labii nasi, anguli depressor, depressor labii, and mentalis.
- The *quizzical look* results from an asymmetric contraction of the major frontalis, outer frontalis, corrugator, lateral corrugator, levator labii, and buccinator.
- The *sadness* expression results from a contraction of the inner frontalis, corrugator, lateral corrugator, anguli depressor, and depressor labii.

Plate 6 demonstrates the performance of our face model construction algorithm on two male individuals ("Giovanni" and "Mick"). Note that the algorithm is tolerant of some amount of facial hair.

Plate 7 shows a third individual "George." Note the image at the lower left, which shows two additional expression effects—cheek puffing, and lip puckering—that combine to simulate the vigorous blowing of air through the lips. The cheek puffing was created by applying outwardly directed radial forces to "inflate" the deformable cheeks. The puckered lips were created by applying radial pursing forces and forward protruding forces to simulate the action of the orbicularis oris sphincter muscle which circles the mouth.

Finally, Plate 8 shows several frames from a two-minute animation "*Bureaucrat Too*" (a second-generation version of the 1990 "*Bureaucrat*" which was animated using the generic facial model in [20]). Here "George" tries to read landmark papers on facial modeling and deformable models in the SIGGRAPH '87 proceedings, only to realize that he doesn't yet have a brain!

6 Conclusion and Future Work

The human face consists of a biological tissue layer with nonlinear deformation properties, a muscle layer knit together under the skin, and an impenetrable skull structure beneath the muscle layer. We have presented a physics-based model of the face which takes all of these structures into account. Furthermore, we have demonstrated a new technique for automatically constructing face models of this sort and conforming them to individuals by exploiting high-resolution laser scanner data. The conformation process is carried out by a feature matching algorithm based on a reusable generic mesh. The conformation process, efficiently captures facial geometry and photometry, positions and scales facial muscles, and also estimates the skull structure over which the new synthetic facial tissue model can slide. Our facial modeling approach achieves an unprecedented level of realism and fidelity to any specific individual. It also achieves a good compromise between the complete emulation of the complex biomechanical structures and functionality of the human face and real-time simulation performance on state-of-the-art computer graphics and animation hardware.

Although we formulate the synthetic facial skin as a layered tissue model, our work does not yet exploit knowledge of the variable thickness of the layers in different areas of the face. This issue will in all likelihood be addressed in the future by incorporating additional input data about the subject acquired using noninvasive medical scanners such as CT or MR.

Acknowledgments

The authors thank Lisa White and Jim Randall for developing the piecewise linear muscle model used to model the mouth. Range/RGB facial data were provided courtesy of Cyberware, Inc., Monterey, CA. The first two authors thank the Natural Science and Engineering Research Council of Canada for financial support. DT is a fellow of the Canadian Institute for Advanced Research.

References

[1] T. Akimoto, Y. Suenaga, and R. Wallace. Automatic creation of 3D facial models. *IEEE Computer Graphics and Applications*, 13(5):16–22, September 1993.

[2] James Doyle and James Philips. *Manual on Experimental Stress Analysis*. Society for Experimental Mechanics, fifth edition, 1989.

[3] Irfan A. Essa. *Visual Interpretation of Facial Expressions using Dynamic Modeling*. PhD thesis, MIT, 1994.

[4] Frick and Hans. *Human Anatomy*, volume 1. Thieme Medical Publishers, Stuttgart, 1991.

[5] H. Gray. *Anatomy of the Human Body*. Lea & Febiber, Philadelphia, PA, 29th edition, 1985.

[6] Brian Guenter. A system for simulating human facial expression. In *State of the Art in Computer Animation*, pages 191–202. Springer-Verlag, 1992.

[7] T. Kurihara and K. Arai. A transformation method for modeling and animation of the human face from photographs. In *State of the Art in Computer Animation*, pages 45–57. Springer-Verlag, 1991.

[8] Y.C. Lee, D. Terzopoulos, and K. Waters. Constructing physics-based facial models of individuals. In *Proceedings of Graphics Interface '93*, pages 1–8, Toronto, May 1993.

[9] N. Magneneat-Thalmann, H. Minh, M. Angelis, and D. Thalmann. Design, transformation and animation of human faces. *Visual Computer*, 5:32–39, 1989.

[10] D. Metaxas and E. Milios. Reconstruction of a color image from nonuniformly distributed sparse and noisy data. *Computer Vision, Graphics, and Image Processing*, 54(2):103–111, March 1992.

[11] M. Nahas, H. Hutric, M. Rioux, and J. Domey. Facial image synthesis using skin texture recording. *Visual Computer*, 6(6):337–343, 1990.

[12] M. Oka, K. Tsutsui, A. Ohba, Y. Kurauchi, and T. Tago. Real-time manipulation of texture-mapped surfaces. In *SIGGRAPH 21*, pages 181–188. ACM Computer Graphics, 1987.

[13] F. Parke. Computer generated animation of faces. In *ACM National Conference*, pages 451–457. ACM, 1972.

[14] F. Parke. Parameterized models for facial animation. *IEEE Computer Graphics and Applications*, 2(9):61–68, November 1982.

[15] F. Parke. Parameterized models for facial animation revisited. In *SIGGRAPH Facial Animation Tutorial Notes*, pages 43–56. ACM SIGGRAPH, 1989.

[16] Elizabeth C. Patterson, Peter C. Litwinowicz, and N. Greene. Facial animation by spatial mapping. In *State of the Art in Computer Animation*, pages 31–44. Springer-Verlag, 1991.

[17] S. Platt and N. Badler. Animating facial expression. *Computer Graphics*, 15(3):245–252, August 1981.

[18] D. Terzopoulos. The computation of visible-surface representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-10(4):417–438, 1988.

[19] D. Terzopoulos and M. Vasilescu. Sampling and reconstruction with adaptive meshes. In *Proceedings of Computer Vision and Pattern Recognition Conference*, pages 70–75. IEEE, June 1991.

[20] D. Terzopoulos and K. Waters. Physically-based facial modeling, analysis, and animation. *Visualization and Computer Animation*, 1:73–80, 1990.

[21] K. Waters. A muscle model for animating three-dimensional facial expression. *Computer Graphics*, 22(4):17–24, 1987.

[22] K. Waters. A physcial model of facial tissue and muscle articulation derived from computer tomography data. In *Visualization in Biomedical Computing*, pages 574–583. SPIE, Vol. 1808, 1992.

[23] K. Waters and D. Terzopoulos. Modeling and animating faces using scanned data. *Visualization and Computer Animation*, 2:123–128, 1991.

[24] L. Williams. Performance-driven facial animation. In *SIGGRAPH 24*, pages 235–242. ACM Computer Graphics, 1990.

[25] J. Yau and N. Duffy. 3-D facial animation using image samples. In *New Trends in Computer Graphics*, pages 64–73. Springer-Verlag, 1988.

Plate 1: Objective. *Input*: Range map in 3D and texture map (top). *Output*: Functional face model for animation.

Plate 2: Raw 512 $\times$ 256 digitized data for Heidi (top left), George (top right), Giovanni (bottom left), Mick (bottom right).

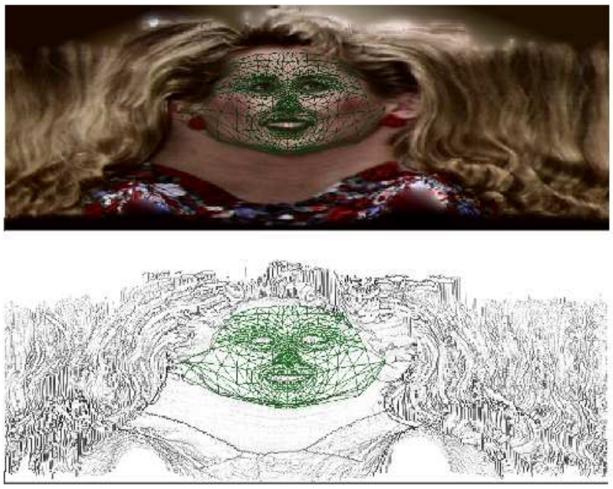


Plate 3: Adapted face mesh overlaying texture map and Laplacian filtered range map of Heidi.

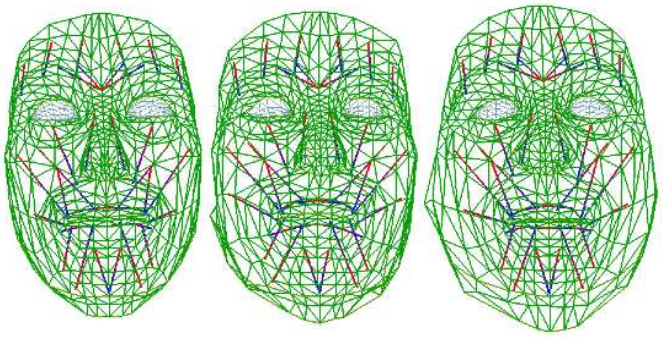


Plate 4: Muscle fiber vector embedded in generic face model and two adapted faces of Heidi and George.

Plate 5: Complete, functional head model of Heidi with physics-based face and geometric eyes, teeth, hair, neck, and shoulders (in Monument Valley).

Plate 6: Animation examples of Giovanni and Mick.

Plate 7: Animation example of George.

Plate 8: George in four scenes from "*Bureaucrat Too*".