Multi-Scale Capture of Facial Geometry and Motion

Multi-Scale Capture of Facial Geometry and Motion

Bernd Bickel* Mario Botsch* Roland Angst† Wojciech Matusik† Miguel Otaduy* Hanspeter Pfister†
Markus Gross*

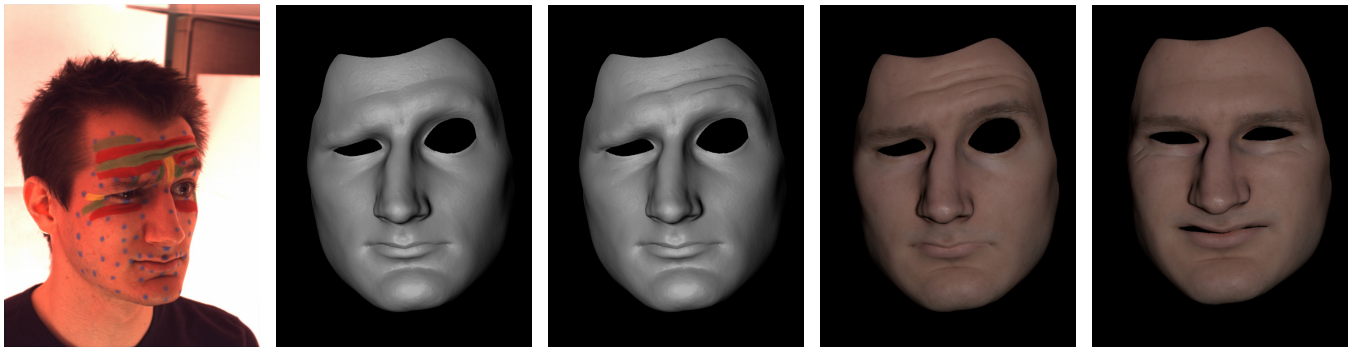

Figure 1: *Animation of a high-resolution face scan using marker-based motion capture and a video-driven wrinkle model. From left to right: video frame, large-scale animation without wrinkles, synthesis of medium-scale wrinkles, realistic skin-rendering, different expression.*

Abstract

We present a novel multi-scale representation and acquisition method for the animation of high-resolution facial geometry and wrinkles. We first acquire a static scan of the face including reflectance data at the highest possible quality. We then augment a traditional marker-based facial motion-capture system by two synchronized video cameras to track expression wrinkles. The resulting model consists of high-resolution geometry, motion-capture data, and expression wrinkles in 2D parametric form. This combination represents the facial shape and its salient features at multiple scales. During motion synthesis the motion-capture data deforms the high-resolution geometry using a linear shell-based mesh-deformation method. The wrinkle geometry is added to the facial base mesh using nonlinear energy optimization. We present the results of our approach for performance replay as well as for wrinkle editing.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Hierarchy and Geometric Transformations; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: animation, motion capture, face modeling

*ETH Zürich, E-mail: [bickel, botsch, otaduy, grossm]@inf.ethz.ch
†MERL, E-mail: [angst, matusik, pfister]@merl.com

1 Introduction

Capturing the likeness and dynamic performance of a human face with all its subtleties is one of the most challenging problems in computer graphics. Humans are especially good at detecting and recognizing subtle facial expressions. A twitch of an eye or a glimpse of a smile are subtle but important aspects of human communication and might occur in a fraction of a second. Both the *dynamics* of the expression and the *detailed spatial deformations* convey personality and intensity [Essa and Pentland 1997].

Although the movie industry continues to make steady progress in digital face modeling, current facial capture, modeling, and animation techniques are not able to generate an adequate level of spatio-temporal detail without substantial manual intervention by skilled artists. Our goal is to easily acquire and represent 3D face models that can accurately animate the spatial and temporal behavior of a real person's facial wrinkles.

Facial skin can be represented by a hierarchy of skin components based on their geometric scale and optical properties [Igarashi et al. 2005]. In the visible domain, they range from the fine scale (e.g., pores, moles, freckles, spots) to the coarse scale (e.g., nose, cheeks, lips, eyelids). Somewhere between those scales are expression wrinkles that occur as a result of facial muscle contraction [Wu et al. 1996]. We call this hierarchy the *spatial scales* of the face.

Facial motion can also be characterized at multiple time scales. At the short-time, high-frequency end of the scale are subtle localized motions that can occur in a fraction of a second. More global motions, such as the movement of the cheeks when we speak, are somewhat slower. And at the smallest spatial scale, features such as pores or moles hardly show any local deformations and can be considered static in time. Expression wrinkles are somewhere between those extremes. They can occur quickly, but they do not move fast during facial expressions (e.g., try moving the wrinkles on your forehead quickly). We call this hierarchy the *motion scales* of the face.

In this paper we present a three-dimensional dynamic face model that can accurately represent the different types of spatial and motion scales that are relevant for wrinkle modeling and animation. A central design element of our model is *a decomposition of the facial features into fine, medium, and coarse spatial scales, each*

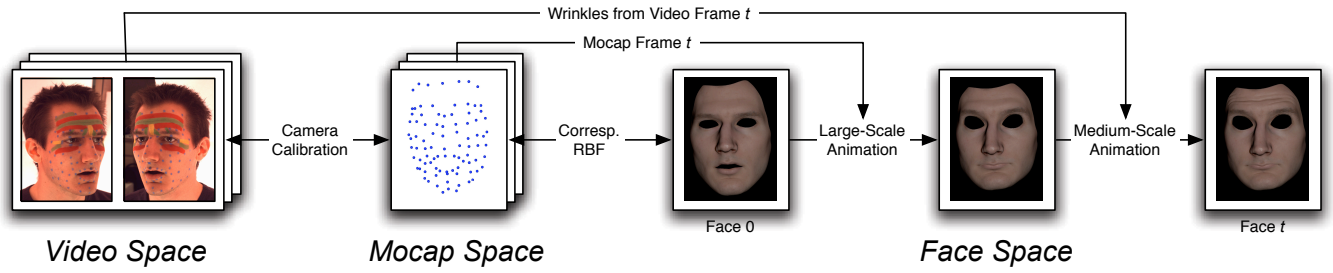

Figure 2: *In our framework we capture a video sequence and motion capture markers of an actor's performance, together with a static, high-resolution face scan. The camera calibration and correspondence function enable the transfer of information between those spaces. Our multi-scale face model first computes a large-scale linear deformation, on top of which medium-scale wrinkles are synthesized.*

representing a different level of motion detail. Medium-scale wrinkle geometry is added to the coarse-scale facial base mesh. Surface microstructure, such as pores, is represented in the fine scale of the model. This decomposition allows us to uniquely tailor the acquisition process to the spatial and temporal scale of expression wrinkle motions.

The conceptual components of our facial-capture approach and representation are illustrated in Figure 2. First we acquire a static high-resolution model of the face, including reflectance data. Then we place approximately 80–90 markers on the face and mark expression wrinkles with a diffuse color. We add two synchronized cameras to a marker-based optical motion-capture system and capture the facial performance. We adapt a linearized thin shell model to deform the high-resolution face mesh according to the captured motion markers. From the video data we estimate the expression wrinkles using a 2D parametric wrinkle model and add them to the deformed 3D face mesh by solving a nonlinear energy minimization problem.

Decomposing the face model into these separate components has several advantages. The motion-capture process needs only the addition of synchronized video cameras to capture expression wrinkles. Throughout the animation, the face geometry maintains the high-resolution of the static scan and preserves a consistent parameterization for the texture and reflectance data. In addition, the face mesh maintains dense correspondence throughout the animation, so that edits on the geometry, textures, and reflectance parameters are automatically propagated to each frame. The model is compact and provides data in a form that is easy to edit.

The primary contribution of our work is the multi-scale facial representation for the animation of expression wrinkles. This model, which is practical and easy to use, allows for the decomposition of the capture process for dynamic faces into fine, medium, and coarse components. The model includes a variety of computational steps for the mapping of motion-capture data, facial deformation, and wrinkle animation.

We have implemented a prototype that demonstrates our approach, and we show results for performance replay and wrinkle processing. Our method creates high-quality facial animations without the intervention of a skilled artist.

2 Related Work

Face modeling, acquisition, and animation are rich areas of research in computer graphics [Noh and Neumann 1999] and computer vision. Here we focus on the related work in capturing 3D models of facial performance.

Marker-Based Motion Capture The basic idea of combining 3D face geometry with marker-based motion-capture data dates back to [Williams 1990]. Today, Vicon dominates the commercial market for marker-based facial-capture systems, although many smaller companies and custom environments exist. These systems acquire data with excellent temporal resolution (up to 450 Hz), but due to their low spatial resolution (100-200 markers) they are not capable of capturing expression wrinkles.

Structured Light Systems Structured light techniques are capable of capturing models of dynamic faces in real time. [Zhang et al. 2004] use spacetime stereo to capture face geometry, color, and motion. They fit a deformable face template to the acquired depth maps using optical flow. [Wang et al. 2004] use a sinusoidal phase-shifting acquisition method and fit a multi-resolution face mesh to the data using free-form deformations (FFD). [Zhang and Huang 2006] improve this acquisition setup and achieve real-time (40 Hz) depth-map acquisition, reconstruction, and display. Structured light systems cannot match the spatial resolution of high-quality static face scans [Borshukov and Lewis 2003; Sifakis et al. 2005] or the acquisition speed of marker-based systems. They also have difficulties in dealing with the concavities and self-shadowing that are typical for expression wrinkles.

Model-Based Animation from Video There has been a lot of work in fitting a deformable 3D face model to video (e.g., [Li et al. 1993; Essa et al. 1996; DeCarlo and Metaxas 1996; Pighin et al. 1999]). Of special interest are linear [Blanz et al. 2003] or multi-linear [Vlasic et al. 2005] morphable models that parameterize variations of human face geometry along different attributes (age, gender, expressions). Because these methods make use of some generic, higher level model, the reconstructed geometry and motion do not approach the quality of person-specific captured data. [Hyneman et al. 2005] compensated the lack of details by adding a dynamic displacement map that included hand-painted wrinkles and furrows.

Image-Based Methods with 3D Geometry [Guenter et al. 1998] and [Borshukov et al. 2003] compute a time-varying texture map from multiple videos and apply it to a deformable face model fitted to the video. [Jones et al. 2006] use the USC Light Stage [Wenger et al. 2005] augmented with a high-speed camera and projector to capture the reflectance field and 3D geometry of a face. They relight the face using the time-varying reflectance data and simulate spatially-varying indirect illumination. Image-based methods are able to produce the most photo-realistic examples of facial performance. However, they typically lack in versatility with respect to editing and changes in head pose and illumination. In principle it should be possible to combine our approach with an image-based method.

Anatomical Face Models Anatomical models provide an animator with model parameters that have bio-mechanical meaning [Koch

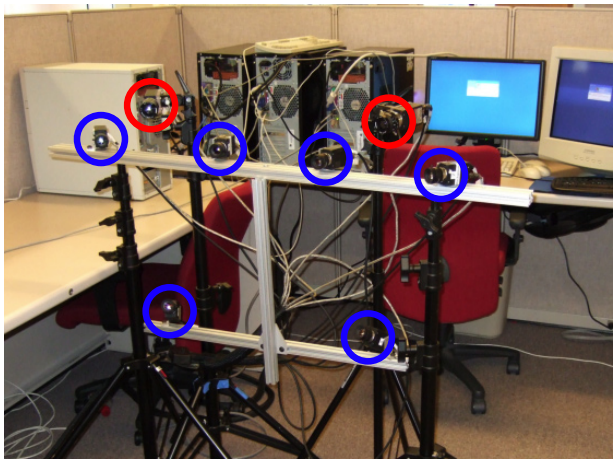

Figure 3: *Our face-capturing setup consists of six cameras (indicated by blue circles) for tracking large-scale facial motions based on 80–90 marker points, complemented by two cameras (red circles) for detecting and fitting medium-scale expression wrinkles.*

et al. 1996; Magnenat-Thalmann et al. 2002]. Some models were specifically developed for wrinkles [Wu et al. 1996; Zhang and Sim 2005; Venkataraman et al. 2005], but have not been applied to facial motion capture. To simulate wrinkle bulges due to facial expressions, we have found that it suffices to model the skin as a nonlinear shell resistant to stretching and bending [Grinspun et al. 2003; Bridson et al. 2003]. [Terzopoulus and Waters 1993] tracked marks on a performer's face using snakes, and used these curves to drive a muscle-based facial model. [Sifakis et al. 2005] developed a highly detailed anatomical face model and morph it to fit laser and MRI scans of a new subject. They use sparse marker-based motion-capture data to automatically determine muscle activations. The face mesh is deformed using a 3D nonlinear finite element method. However, generic anatomical face models are currently not able to produce expression wrinkles for an individual.

3 Data Acquisition

The **static, high-resolution face mesh** is acquired using a commercial face-scanning system from 3QTech [1]. We use the method of [Nehab et al. 2005] to improve the acquired geometry using photometric stereo, thereby successfully capturing even fine-scale geometric details. The acquisition process takes 30 seconds and produces a face mesh F with approximately 500k–700k vertices, depending on the face size. We also acquire reflectance data and compute the albedo texture, spatially-varying coefficients of the Torrance-Sparrow BRDF model, and subsurface scattering parameters [Weyrich et al. 2006].

The **faster, large-scale face motion** is captured with a setup consisting of six Basler cameras running at 50 fps with a resolution of 656×490 pixels (Figure 3). The cameras are placed slightly staggered so that each point of the face is clearly visible in at least two cameras. All cameras are synchronized using an external trigger signal from a USB I/O device [2]. We track about 80–90 marker points on the face, which are painted blue to maximize the color difference with skin.

The tracking requires a correspondence between points in the video footage, which we establish by consistently labeling them by hand in the first frame of each camera. The labeled 2D points are then tracked throughout the whole sequence independently for each camera. After establishing the intrinsic camera parameters [Svoboda et al. 2005] we use a standard triangulation method to compute the 3D location of every marker in every frame. To suppress noise in the reconstructed 3D positions, we apply a spatio-temporal bilateral filter to the marker positions, which reduces smoothing for time frames with large movement. This controlled smoothing is important for preserving convincing facial expressions.

To capture the **slower, medium-scale expression wrinkles** we add two high-resolution Basler cameras with 12.5 fps and 1384×1038 pixels. These cameras run exactly four times slower than the motion-capture cameras, making the synchronization easier. All cameras are extrinsically calibrated so that the reconstructed motion-capture performance can be easily projected into the views of the high-resolution cameras.

The scene is captured under approximate ambient uniform illumination, without any light source intensity calibration. We assume that the subject faces approximately the same direction throughout the acquisition process.

4 Large-Scale Animation

The motion-tracking process results in a set of time-dependent marker positions $\mathbf{m}_{i,t} \in \mathbb{R}^3$, $i = \{1,\dots,n\}$, $t = \{0,1,\dots\}$ in the reference space of the motion-capture system (*mocap space*). At a certain time t, the difference vectors $\left(\mathbf{m}_{i,t} - \mathbf{m}_{i,0}\right)$ represent point-samples of the continuous deformation field that deforms the initial face model into the expression at frame t. Our goal is to deform the initial face mesh F based solely on these displacement constraints.

Since the 3D scan F and the mocap points are defined with respect to different coordinate systems, the points $\mathbf{m}_{i,0}$ and their respective displacements $\left(\mathbf{m}_{i,t} - \mathbf{m}_{i,0}\right)$ first have to be mapped to the coordinate space of the face mesh F (*face space*), resulting in points $\mathbf{f}_{i,0}$ and displacements $\mathbf{d}_{i,t} = \left(\mathbf{f}_{i,t} - \mathbf{f}_{i,0}\right)$. We achieve this by establishing a correspondence function as described in Section 4.1.

The resulting displacements $\mathbf{d}_{i,t}$ in face space are then used as constraints for our physically inspired face deformation model. Notice that a physically accurate face deformation — including the interaction of bones, muscles, and tissue — is too complex for our purposes. From our experiments it turned out that the mocap points capture the *large-scale* face behavior sufficiently well, so that we can use a simplified deformation model that interpolates the mocap points (see Section 4.2).

4.1 Mocap / Face Correspondence

In order to transfer the mocap displacements $\left(\mathbf{m}_{i,t} - \mathbf{m}_{i,0}\right)$ to displacements $\mathbf{d}_{i,t} = \left(\mathbf{f}_{i,t} - \mathbf{f}_{i,0}\right)$ in face space we have to establish a correspondence map between the mocap points and the 3D face mesh. For that we pick the mocap frame most similar in facial expression to the face scan. Let us assume without loss of generality that this is the first frame, consisting of the points $\mathbf{m}_{i,0}$.

The user first manually selects the corresponding vertex positions $\mathbf{f}_{i,0} \in F$ by clicking on the face mesh. Given this coarse set of corresponding points, position, orientation, and scaling of the face mesh could in principle be adjusted using Horn's shape matching method [1987]. Since the mocap points and the face mesh were captured from the same person, the resulting rigid registration would be quite accurate. However, subtle variations in facial expression, e.g., in the opening angle of the mouth, would not be accounted for.

[1] www.3qmd.com

[2] http://www.datx.com/econ

Therefore, we use a non-rigid registration technique that interpolates the discrete point correspondences over space in order to achieve a smooth correspondence space warp $\mathbf{c} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$. Similar to [Noh and Neumann 2001], we use radial basis functions (RBFs) for this scattered data interpolation problem, which represents the function $\mathbf{c}$ as

$$\mathbf{c}(\mathbf{x}) \;=\; \sum_{i=1}^{n} \mathbf{w}_i \cdot \phi(\|\mathbf{x}-\mathbf{c}_i\|) + \mathbf{q}(\mathbf{x}) \quad , \tag{1}$$

where $\phi : \mathbb{R} \rightarrow \mathbb{R}$ is a scalar basis function, $\mathbf{w}_i, \mathbf{c}_i \in \mathbb{R}^3$ are the weights and centers of the RBF, and $\mathbf{q} : \mathbb{R}^3 \rightarrow \mathbb{R}$ is a quadratic trivariate polynomial. In order to find the RBF that interpolates the constraints, i.e.,

$$\mathbf{c}\big(\mathbf{m}_{i,0}\big) \;=\; \mathbf{f}_{i,0}\,, \qquad i = 1,\ldots,n \quad ,$$

the centers are chosen to coincide with the constraints, i.e., $\mathbf{c}_i = \mathbf{m}_{i,0}$. This results in a symmetric linear system to be solved for the weights $\mathbf{w}_i$ and the coefficients of the quadratic polynomial $\mathbf{q}$ [Carr et al. 2001]. In contrast to [Noh and Neumann 2001] we use the triharmonic RBF basis function $\phi(r) = r^3$, which yields a smooth C^2 function of provable global fairness [Duchon 1977; Botsch and Kobbelt 2005]. Although the resulting linear system is dense, it can be solved efficiently since the number of constraints n is < 100. Notice that because of the polynomial term $\mathbf{q}(\mathbf{x})$, the function $\mathbf{c}$ can exactly reproduce affine motions, which makes a rigid pre-registration unnecessary.

Given the space warp $\mathbf{c}$, we now have to transform the mocap displacements $\big(\mathbf{m}_{i,t} - \mathbf{m}_{i,0}\big)$ into face space. For a similar setting, [Noh and Neumann 2001] proposed a heuristic to transfer displacement vectors from one *mesh* onto another by adjusting the displacements' scaling and orientation based on local frames and local bounding boxes associated with mesh vertices. In contrast, we want to transform displacements from only a coarse point cloud $\mathbf{m}_{i,0}$ to a face mesh, and hence cannot use their surface-to-surface heuristic.

However, the space warp $\mathbf{c}$ already contains all the required information to transfer the mocap displacements: We simply use $\mathbf{c}$ to transfer the displaced mocap points $\mathbf{m}_{i,t}$, which yields $\mathbf{f}_{i,t}$. From those points we compute the face-space displacements as

$$\mathbf{d}_{i,t} \;=\; \mathbf{c}(\mathbf{m}_{i,t}) - \mathbf{f}_{i,0} \quad .$$

4.2 Linear Deformation Model

After transferring the mocap displacements into face space, we deform the initial face mesh based on these displacement constraints. This requires a deformation function $\mathbf{d}_t : F \rightarrow \mathbb{R}^3$ that is smooth and physically plausible while interpolating the constraints of frame t:

$$\mathbf{d}_t\big(\mathbf{f}_{i,0}\big) \;=\; \mathbf{d}_{i,t}\,, \qquad \forall\, i = 1,\ldots,n \quad , \tag{2}$$

such that $\mathbf{f}_{i,0} + \mathbf{d}_t\big(\mathbf{f}_{i,0}\big) = \mathbf{f}_{i,t}$. Note that another RBF-like space deformation is not suitable, since the desired deformation might be discontinuous around the mouth and eyes, whereas an RBF would always yield a C^2 continuous deformation.

For the global *large-scale* face deformation we propose using a *linear* shell model, since this allows for efficient as well as robust animations, even for our complex meshes of about 700k vertices. The missing medium-scale nonlinear effects, i.e., wrinkles and bulges, are added later on as described in Section 5.3.

Our linearized shell model incorporates the prescribed displacements $\mathbf{d}_{i,t}$ as boundary constraints, and otherwise minimizes surface stretching and bending. After linearization, the required stretching and bending energies can be modeled as integrals over first- and second-order partial derivatives of the displacement function $\mathbf{d}_t$ [Celniker and Gossard 1991]:

$$\int_F \underbrace{k_s \left(\left\| \frac{\partial \mathbf{d}_t}{\partial u} \right\|^2 + \left\| \frac{\partial \mathbf{d}_t}{\partial v} \right\|^2 \right)}_{\text{stretching}} + \underbrace{k_b \left(\left\| \frac{\partial^2 \mathbf{d}_t}{\partial^2 u} \right\|^2 + 2 \left\| \frac{\partial^2 \mathbf{d}_t}{\partial u \partial v} \right\|^2 + \left\| \frac{\partial^2 \mathbf{d}_t}{\partial^2 v} \right\|^2 \right)}_{\text{bending}} \mathrm{d}u\,\mathrm{d}v. \tag{3}$$

The deformation $\mathbf{d}_t$ that minimizes this energy functional can be found by solving its corresponding Euler-Lagrange equations

$$-k_s\,\Delta \mathbf{d}_t \;+\; k_b\,\Delta^2 \mathbf{d}_t \;=\; 0 \tag{4}$$

under the constraints (2). Since our displacement function $\mathbf{d}_t$ is defined on the initial mesh F, i.e., on a triangulated two-manifold, Δ represents the discrete Laplace-Beltrami operator as defined in [Meyer et al. 2003]. With this discretization, the above PDE leads to a sparse linear system to be solved for the displacements at all mesh vertices, similar to [Botsch and Kobbelt 2004]. Notice, however, that in contrast to the latter paper, we compute a smooth deformation field instead of a smooth surface. As a consequence, all small-scale details of F, such as pores and fine aging wrinkles, are retained by the deformation.

This linear system has to be solved for every frame of the mocap sequence, since each set of transferred mocap displacements $\mathbf{d}_{i,t}$ yields new boundary constraints, i.e., a new right-hand side. Although the linear system can become rather complex — its dimension is the number of free vertices — it can be solved efficiently using either a sparse Cholesky factorization or iterative multigrid solvers [Botsch et al. 2005; Shi et al. 2006]. All animations in this paper were computed with the parameters $k_s = 1$ and $k_b = 100$.

Since the global face motion does not contain significant local rotations, there is no need to explicitly rotate small-scale details, e.g., by multi-resolution decomposition or differential coordinates [Botsch and Sorkine 2007]. Although the deformation of the human face is the result of complex interactions between skull, muscles, and skin tissue, the linear deformation model yields visually plausible results because the motion-capture markers provide sufficient geometric constraints. While the resulting animations are of high visual quality, nonlinear effects such as expression wrinkle formation obviously cannot be produced by the linearized deformation model. The next section describes how we enhance the large-scale facial animation with medium-scale expression features extracted from video data.

5 Medium-Scale Animation

In this section, we first describe an image-based algorithm for tracking wrinkles in video data, fitting 2D B-splines to them, and estimating their cross-section shapes from self-shadowing effects. Then, we describe a physically-inspired *nonlinear* shell deformation model that, with the 2D data as input, allows us to synthesize *medium-scale* 3D expression wrinkles and bulging onto the large-scale animation.

Skin is a multilayer, anisotropic, viscoelastic tissue, whose mechanical behavior is dominated by collagen fibers present in the dermis [Lanir 1987]. Hence, accurate simulation of skin folding would require a complex volumetric representation with carefully chosen model parameters [Magnenat-Thalmann et al. 2002; Sifakis et al. 2005]. For the purpose of simulating wrinkle bulge formation due to facial expressions, however, we found our nonlinear shell model to be sufficient.

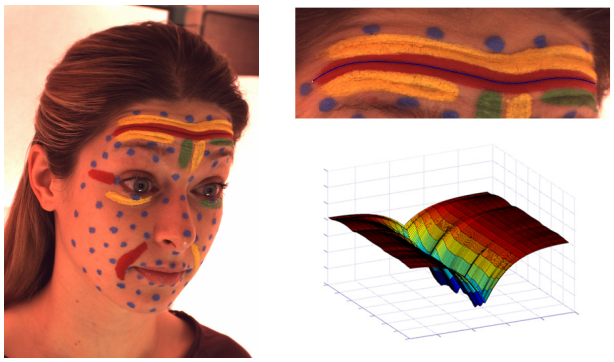

Figure 4: *For each wrinkle marked in the video (left) a B-spline curve $\mathbf{v}(x)$ is fitted (top right) and corresponding cross-section shapes $S(w,d,p)$ are extracted (bottom right).*

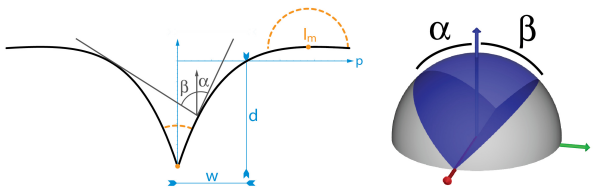

Figure 5: *Left: Wrinkle cross-section function $S(w,d,p)$. At point m the observed intensity $I_{\text{obs}}(m)$ is maximal because no incoming light is blocked, in contrast to points $p \in [-m,m]$. Right: The observed intensity $I_{\text{obs}}(p)$ at an arbitrary point p of the wrinkle is computed by integrating the incoming light over the hemisphere. The two angles α and β determine the area of the spherical wedge of incoming (blue) and blocked (gray) light.*

5.1 Wrinkle Tracking

In the spirit of shape-from-shading, we exploit self-shadowing effects to track wrinkles and estimate their properties. In the acquisition process, wrinkles are marked with a diffuse color, as shown in Figure 4. It masks the underlying skin, making the depth estimation more robust and independent of skin type and pigmentation, e.g., freckles. Furthermore, to simplify the tracking we choose colors that are clearly silhouetted against skin albedo. Neighboring wrinkles that are close to each other are marked with different colors. Our lighting setup produces approximately uniform ambient illumination.

The first step in wrinkle tracking is to find image pixels associated with each predefined wrinkle. We use a binary support vector machine (SVM) with L2 soft margin and RBF kernel [Cortes and Vapnik 1995] to classify the video images into wrinkle and non-wrinkle patches. It turned out that training the machine was easy. In most cases it was sufficient to create a binary mask for the first image in the video, and use this as training data for estimating the support vectors that were then used for classifying the remaining video images. In case of multiple wrinkles and thus multiple marker colors, the binary support vector machine is trained and applied for each color independently. We apply morphological operations (e.g., erosion and dilation) to remove possible pixel classification errors caused by noise.

For wrinkle patches, we represent each wrinkle valley in a compact and smooth way using a uniform B-Spline curve $\mathbf{v} : \mathbb{R} \to \mathbb{R}^2$. For each patch of wrinkle pixels $\{\mathbf{p}_1, \ldots, \mathbf{p}_k\}$, we perform a PCA, resulting in a mean pixel position $\bar{\mathbf{p}}$ and the patch's principal axis $\mathbf{a}$. We parameterize the pixels $\mathbf{p}_i$ by their position x_i along the axis $\mathbf{a}$, i.e.,

$$x_i := x(\mathbf{p}_i) = (\mathbf{p}_i - \bar{\mathbf{p}})^T \mathbf{a}.$$

Since wrinkles do not deviate too much from straight lines, this kind of parameterization does not cause any problems. The number of control points is chosen between 5–12, depending on the length of the wrinkle.

The spline $\mathbf{v}(x)$ is fitted in a weighted least-squares sense, minimizing an energy

$$E_{\text{spline}} = \sum_{i=1}^{k} w_i \, \|\mathbf{p}_i - \mathbf{v}(x_i)\|^2 \qquad (5)$$

that measures the Euclidean distance from the pixels $\mathbf{p}_i$ to the valley curve $\mathbf{v}(x)$. We weight each pixel $\mathbf{p}_i$ with a value w_i inversely proportional to its gray-scale intensity g_i, $w_i = (g_{max} - g_{min})/(g_i - g_{min})$, where g_{max} and g_{min} are the maximum and minimum of the observed gray-scale intensities in the wrinkle segment. Due to self-shadowing effects, wrinkle valleys are darker than their surroundings, and our weighting strategy favors the B-spline curve that follows the wrinkle valley.

5.2 Cross-Section Shape Estimation

A wrinkle cross-section can be classified into two characteristic parts: the wrinkle valley and the bulges, one on each side. We have designed a method for locally estimating the gradient of wrinkle valleys from the ratio of observed image intensity on the bulges and the valley. Our method exploits the self-shadowing effect of wrinkles, assuming a Lambertian surface (thanks to the use of a diffuse marker color) and uniform but unknown ambient illumination. We define wrinkle gradient through width $w(x)$ and depth $d(x)$ in image space, varying along the parameterization x of each valley spline $\mathbf{v}(x)$. Later, in 5.3, we describe the projection of image space wrinkles onto the large-scale facial animation. Following Bando et al. [2002], we use an analytic function to model the cross-section of a wrinkle valley. The wrinkle bulge is a more complex phenomenon involving neighboring wrinkles and, unlike Bando et al., we model it separately as described in the next subsection. With p the distance orthogonal to the spline $\mathbf{v}(x)$, the wrinkle cross-section at x is modeled by the function

$$S(p) = S(w,d,p) = d \cdot \left(\frac{p}{w} - 1\right) \cdot \mathrm{e}^{-p/w}.$$

Then, the intensity at a point $(p, S(p))$ on this cross-section (under ambient illumination I_{ambient}) can be locally estimated by employing a 2.5D model (Figure 5) to integrate the incoming light over a hemisphere Ω:

$$I(p) = \frac{1}{\pi} \int_{\Omega} V(p,\omega) \cdot \left(\mathbf{n}(p)^T \omega\right) \cdot I_{\text{ambient}} \, \mathrm{d}\omega, \qquad (6)$$

where $\mathbf{n}(p)$ is the normal vector to the curve $S(p)$ and $V(p,\omega)$ is the visibility function, which is 1 if $(p, S(p))$ is visible from direction ω and 0 otherwise. Notice that if no incoming light is blocked, i.e., $V(m,\omega) = 1, \forall \omega$, the intensity is maximum, $I_{\max}(p) = I_{\text{ambient}}$.

For a given wrinkle shape $S(p) = S(w,d,p)$, the visibility function $V(p,\omega)$ can be computed from the apex angles α and β of the spherical wedge (Figure 5), representing all directions of incoming light. These angles are given by the tangent at point $(p, S(p))$ of the wrinkle shape and the tangent at the opposite valley shape going through $(p, S(p))$. Once α and β are computed, the hemisphere integral (6) turns into a 2.5D integral over the visible spherical wedge

$$I(p) = \frac{1}{\pi} \int_{-\alpha}^{\beta} \int_{0}^{\pi} \left(\mathbf{n}(p)^T \omega\right) \cdot I_{\text{ambient}} \, \mathrm{d}\omega.$$

Our goal is to find the cross-section parameters d and w such that the computed intensities $I(p)$ match the intensities $I_{\text{obs}}(p)$ observed in the image.

Assuming that there is a point m on the wrinkle bulge without self-shadowing, we can estimate the ambient illumination, $I_{\text{ambient}} \approx I_{\text{obs}}(m)$. Then, we can work with the ratio $I(p)/I(m)$, which is independent of I_{ambient}, and compute the wrinkle-shape parameters without measuring or calibrating the light-source intensity. We obtain the intensity values of a cross-section $I_{\text{obs}}(p)$ by extracting the pixel values perpendicular to the valley spline $\mathbf{v}(x)$ in 2D image space. Then, we compute d and w by minimizing the nonlinear least-squares problem (using Matlab's Gauss-Newton optimization)

$$\min_{d,w} \sum_p \left\| \frac{I_{\text{obs}}(p)}{I_{\text{obs}}(m)} - \frac{\int_\Omega V(p,\omega) \left(\mathbf{n}(p)^T \omega\right) \mathrm{d}\omega}{\pi} \right\|^2 . \quad (7)$$

If no wrinkle is present, the fitted depth d is 0.

5.3 3D Wrinkle Synthesis

In Section 4.2 we employed a *linear* shell model for the *large-scale* face animation. We now refine this result by synthesizing *medium-scale* wrinkles onto the large-scale facial animation based on a *nonlinear* shell energy minimization.

We employ the nonlinear discrete shell energy of [Grinspun et al. 2003] to measure the difference between the initial mesh F and its deformed version. Their energy is defined in terms of geometric quantities of the triangle mesh, and measures the change of edge lengths $\|e_i\|$, dihedral angles θ_i, and triangle areas $\|t_i\|$, over all edges e_i and triangles t_i.

$$\begin{aligned} E_{\text{shell}} \quad = \quad & \sum_{e_i} k_e \cdot \frac{(\|e_i\| - \|\bar{e}_i\|)^2}{\|\bar{e}_i\|} + k_b \cdot \frac{\|\bar{e}_i\| \left(\theta_i - \bar{\theta}_i\right)^2}{\bar{h}_i} \quad (8) \\ & + \sum_{t_i} k_a \cdot \frac{(\|t_i\| - \|\bar{t}_i\|)^2}{\|\bar{t}_i\|} \ , \end{aligned}$$

where the "barred" terms $\|\bar{e}_i\|$, $\|\bar{t}_i\|$, and $\bar{\theta}_i$ denote the edge length, triangle area, and dihedral angle in the undeformed rest state F. The angle weighting by edge length $\|\bar{e}_i\|$ and triangle height $\bar{h}_i$ accounts for irregular triangulations [Grinspun et al. 2003].

The geometric constraints for the nonlinear energy minimization are constituted by the locations and cross-section profiles of the wrinkle valleys extracted from video data. We map the linearly deformed face F into mocap space using the inverse correspondance map $\mathbf{c}^{-1}$, project all 2D wrinkle splines $\mathbf{v}(x)$ onto it based on the camera parameters, and map the result back to face space using $\mathbf{c}$.

Then, we evaluate the wrinkle shape function $S(p)$ in the valley $-w < p < w$ for all cross-sections x along the spline $\mathbf{v}(x)$, and offset the affected mesh vertices along their (smoothed) normals (see Section 5.4). This procedure provides absolute positions for vertices corresponding to the wrinkle valley. Recall that we work with two cameras in order to cover the whole facial area. If a wrinkle is tracked by both cameras, we merge the detected segments by simple snapping and linear blending.

With wrinkle valleys constituting the geometric constraints, we perform a minimization of the shell energy (8). This updates the vertices of the face mesh, such that surface area and curvature of the initial scan F are approximately preserved, which then leads to the required bulging between neighboring wrinkles.

We solve the minimization problem, and thus compute the final face animation, using Gauss-Newton optimization. We initialize vertex positions at the configuration obtained by the large-scale linear deformation. All animations in this paper were rendered with the parameters $k_b = 2$, $k_e = 300$, and $k_a = 30$k. As we are dealing with high-resolution meshes, a global Gauss-Newton optimization would be computationally too expensive. Therefore, as a heuristic, we determine the influence region of the wrinkles by using a predefined maximum influence distance. By merging overlapping regions, we obtain an automatic segmentation of the face into wrinkling and non-wrinkling areas. For each wrinkling area, we optimize (8) independently, while keeping the remaining vertices fixed.

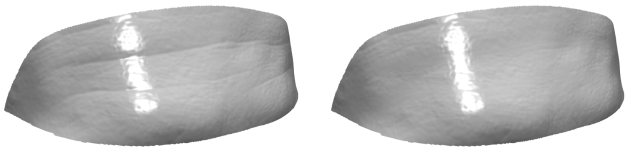

Figure 7: *In a preprocess we remove wrinkles that already exist in the initial, relaxed-pose face scan (left). Our multi-scale smoothing eliminates medium-scale wrinkles, but at the same time preserves large-scale geometry as well as small-scale details (right).*

5.4 Wrinkle Removal

The framework presented so far detects wrinkles from video data, projects them onto the linearly deformed face mesh, and recovers bulges between wrinkles by a nonlinear energy minimization. If the initial face scan (relaxed pose) already contains noticeable wrinkles, however, they would be detected and erroneously amplified by our technique. We therefore remove existing wrinkles from the static face scan in a preprocess.

We employ a three-step multi-scale smoothing to wrinkle regions in order to remove only the medium-scale wrinkles, but preserve both the large-scale face geometry and the small-scale details.

1. First we subtract the fine-scale details by a small amount of Laplacian smoothing [Desbrun et al. 1999], and store them as local-frame displacements [Kobbelt et al. 1999].

2. We eliminate the medium-scale wrinkles by minimizing curvature energies. This is equivalent to computing the steady state of bi-Laplacian smoothing [Desbrun et al. 1999], but only requires solving a bi-Laplacian linear system.

3. The resulting surface patch is smooth and blends with the surrounding non-wrinkle mesh in a tangent-continuous manner. Consequently, it preserves the global, large-scale geometry. On top of this smooth patch we finally add back the fine-scale details as normal displacements to get the desired result.

The effect of this multi-scale smoothing is depicted in Figure 7.

6 Results

This section presents still images from various animation sequences computed with our model. To see the full model performance please see the accompanying video. All images and animations in this paper were rendered using an extended version of PBRT[3] that supports skin subsurface scattering. The facial reflectance data as well as the high-resolution facial geometry were acquired using the hardware described in [Weyrich et al. 2006].

[3] http://www.pbrt.org/

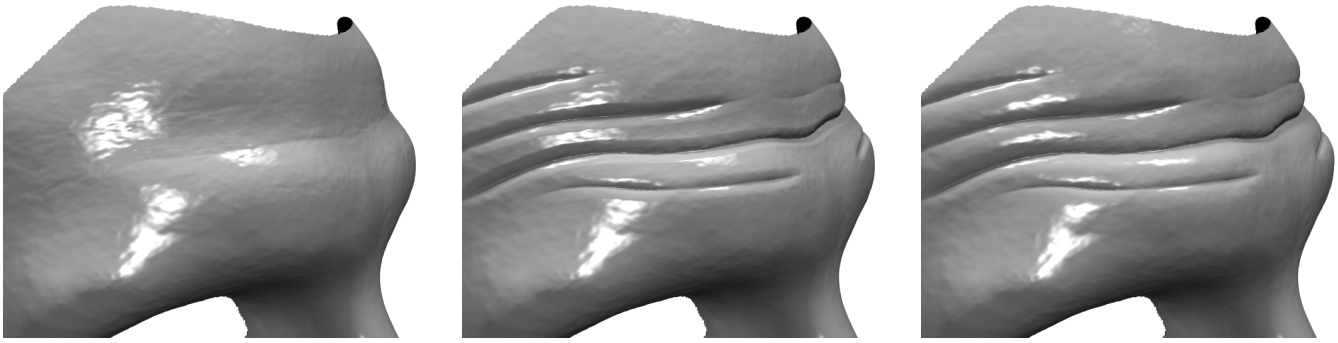

Figure 6: *The synthesis of medium-scale wrinkles start from the large-scale linearly deformed mesh (left), on top which wrinkle valleys are added as normal displacements, based on the projected wrinkle functions extracted from the video (center). Our nonlinear minimization of surface stretching and bending finally gives the missing bulging between neighboring wrinkles (right).*

6.1 Performance

In this section we give timings for the different stages of processing the video data and animating the face mesh. Since the processing times are almost equal for the different subjects we list only average timings. All computations were carried out on a standard PC with an Intel Pentium 2.8 GHz and 1 GByte of main memory.

The large-scale linear animation involves the computation of the correspondence RBF (Equation 1) and the solution of the bi-Laplacian linear system (Equation 4) for the actual surface deformation. The RBF interpolation can be solved within milliseconds due to its small size. After a pre-factorization of about 120s, the surface deformation can be performed at a rate of about 3s per frame.

For each video frame, the wrinkle-capture process takes about 5s for image segmentation and spline fitting, and about 8min for the nonlinear cross-section estimation, which currently is implemented in Matlab. The medium-scale wrinkle synthesis projects the extracted 2D wrinkles onto the large-scale animation and solves a nonlinear minimization of stretching and bending, which is the dominant cost of about 20min. The final rendering takes about 10min per frame in high quality mode.

6.2 Expression Replay and Wrinkle Editing

Figure 8 depicts a sequence of still images with varying facial expressions for two different subjects. The images were taken from the video animation and show replays of facial expressions animated with our model. For all facial animations, we cut out the subjects' eyes, and the meshes were clipped along the hair and ear lines of the persons. In the second column from left, we show the deformed facial geometry as computed by our large-scale linear deformation model. Note that at this stage, the faces do not contain any expression wrinkles. The third (geometry only) and fourth (skin rendering) columns show the results after adding wrinkles to the deformed model. The facial expression of the female subject in the upper row has large forehead wrinkles that are modeled and animated very realistically by our model. The performance of the male subject primarily leads to wrinkle formations around the eyes, and our model captures the resulting deformations very convincingly.

Figure 9 presents two standard facial expressions, "astonished" and "angry," which lead to different wrinkle formation. Despite the lack of self-collision detection, our model replays these deformations very well. An illustration of our editing capabilities is given in Figure 10. In this sequence, we gradually scaled the wrinkle depth to weaken or enhance the effect of the forehead wrinkles. The skin bulges created by our wrinkle model provide a realistic deformation of the facial skin in all images of this sequence. The figure also shows the flexibility of our multi-scale model on (per-frame) manual edits. The rightmost image shows a single frame edit, where the nasolabial wrinkles were added manually simply by drawing their valley splines into the video frame and specifying depth and width parameters.

7 Discussion and Future Work

By design, our model model is suited only for performance capture and replay. In its current form it does not provide intuitive parameters for animation control. A further limitation of our model is its lack of facial anatomy and physics. This includes eyes and teeth, but also skin and muscle layers, or self-collision. If needed, such features could be imported by combining our model with other existing ones, such as [Sifakis et al. 2005]. The acquisition and hence the ultimate quality is currently limited by the frame rate of the cameras and by the homemade motion tracker we utilized to produce our results. However, this is not an inherent limitation of the model, because it could easily be alleviated by taking commercially available high-speed cameras and motion-tracking systems.

The lack of a strict vertex correspondence between wrinkles and mesh over time could potentially lead to minor drifts of the wrinkles, but we have not observed this issue in practice. Another important extension of our approach is an explicit representation of small wrinkles. The finite camera resolution and the explicit coloring of wrinkles artificially limits their actual minimum size. We plan to extend our multi-scale model to explicitly represent and animate small-scale wrinkles. Finally, we are interested in wrinkle animation transfer between individuals, an issue of high practical relevance for applications in the special-effects industry.

Acknowledgments

We would like to thank the anonymous reviewers for their insightful comments, Janet McAndless for scanning our subjects, Basil Weber for extending PBRT to support the skin-reflectance model, Jennifer Roderick for proofreading the paper, and our patient actors for their willingness to be painted. This research has been supported by the NCCR Co-Me of the Swiss National Science Foundation.

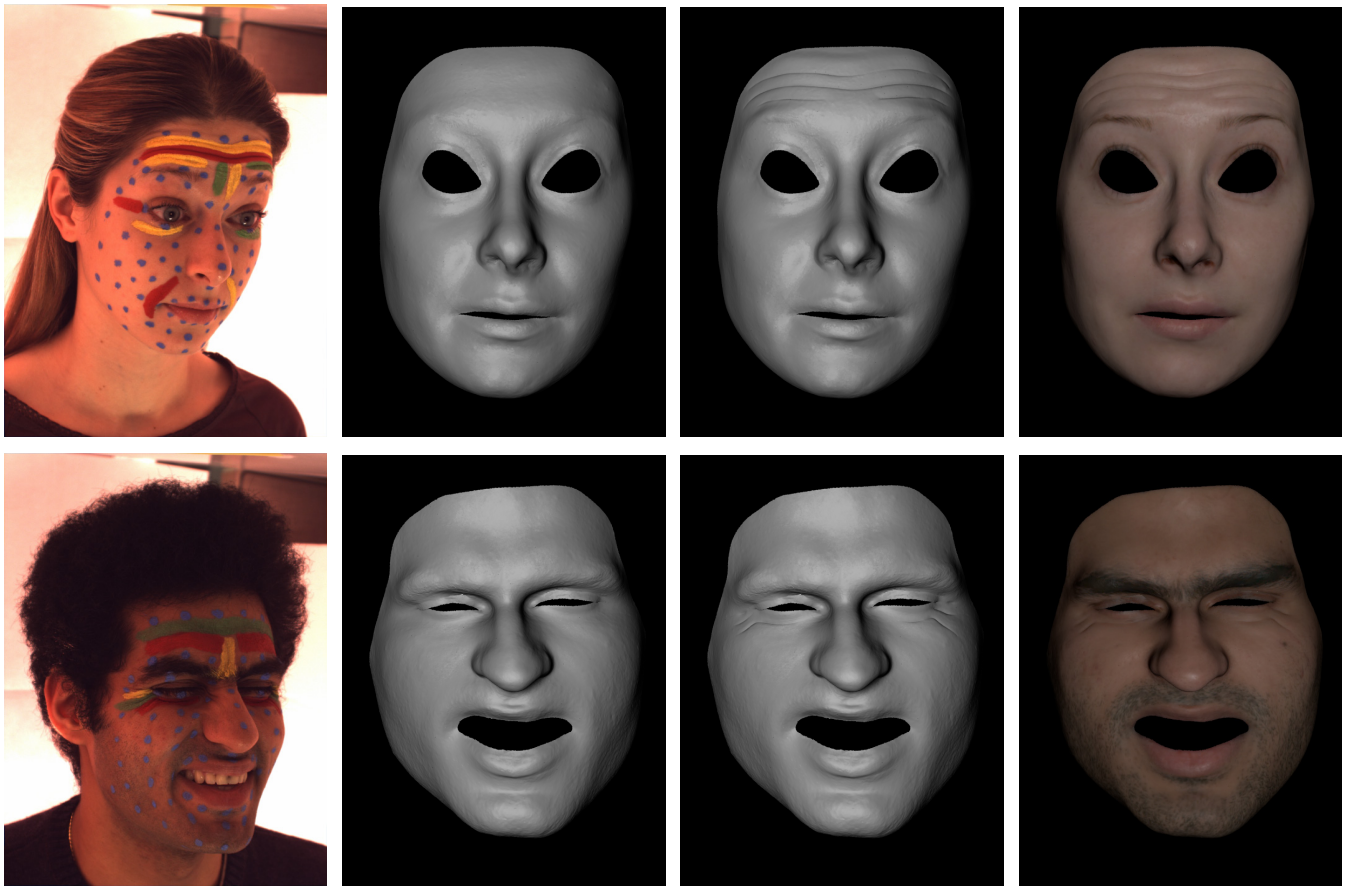

Figure 8: *Performance replay of captured video sequences (left) of two different subjects. The large-scale linear animation first deforms the high-resolution face mesh based on tracked mocap markers (center left). The missing medium-scale expression wrinkles are synthesized by a nonlinear energy minimization (center right). The rightmost column shows high-quality skin rendering including subsurface scattering.*

Figure 9: *Two more examples showing facial expressions for the standard emotions "astonished" (left) and "angry" (right).*

Figure 10: *Our multi-scale face model enables wrinkle processing by scaling the depth parameters extracted from video. This allows us to either weaken (50%) or enhance (200%) the original wrinkles (100%). The rightmost image shows a single frame edit, where the nasolabial wrinkles were added manually simply by drawing their valley splines into the video frame and specifying depth and width parameters.*

References

BANDO, Y., KURATATE, T., AND NISHITA, T. 2002. A simple method for modeling wrinkles on human skin. In *Proc. of Pacific Conference on Computer Graphics and Applications*.

BLANZ, V., BASSO, C., POGGIO, T., AND VETTER, T. 2003. Reanimating faces in images and video. *Computer Graphics Forum 22*, 3, 641–650.

BORSHUKOV, G., AND LEWIS, J. 2003. Realistic human face rendering for "The Matrix Reloaded". In *ACM SIGGRAPH 03 Sketches & Applications*.

BORSHUKOV, G., PIPONI, D., LARSEN, O., LEWIS, J., AND TEMPELAAR-LIETZ, C. 2003. Universal capture – Image-based facial animation for "The Matrix Reloaded". In *ACM SIGGRAPH 03 Sketches & Applications*.

BOTSCH, M., AND KOBBELT, L. 2004. An intuitive framework for real-time freeform modeling. *ACM Transactions on Graphics 23*, 3, 630–634.

BOTSCH, M., AND KOBBELT, L. 2005. Real-time shape editing using radial basis functions. *Computer Graphics Forum 24*, 3, 611–621.

BOTSCH, M., AND SORKINE, O. 2007. On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics (TVCG), to appear*.

BOTSCH, M., BOMMES, D., AND KOBBELT, L. 2005. Efficient linear system solvers for geometry processing. In *11th IMA conference on the Mathematics of Surfaces*, 62–83.

BRIDSON, R., MARINO, S., AND FEDKIW, R. 2003. Simulation of clothing with folds and wrinkles. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, 28–36.

CARR, J. C., BEATSON, R. K., CHERRIE, J. B., MITCHELL, T. J., FRIGHT, W. R., MCCALLUM, B. C., AND EVANS, T. R. 2001. Reconstruction and representation of 3D objects with radial basis functions. In *Proc. of ACM SIGGRAPH 01*, 67–76.

CELNIKER, G., AND GOSSARD, D. 1991. Deformable curve and surface finite-elements for free-form shape design. In *Proc. of ACM SIGGRAPH 91*, 257–266.

CORTES, C., AND VAPNIK, V. 1995. Support-vector networks. *Machine Learning 20*, 3, 273–297.

DECARLO, D., AND METAXAS, D. 1996. The integration of optical flow and deformable models with applications to human face shape and motion estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 231–238.

DESBRUN, M., MEYER, M., SCHRÖDER, P., AND BARR, A. H. 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proc. of ACM SIGGRAPH 99*, 317–324.

DUCHON, J. 1977. Spline minimizing rotation-invariant semi-norms in Sobolev spaces. In *Constructive Theory of Functions of Several Variables*, W. Schempp and K. Zeller, Eds., no. 571 in Lecture Notes in Mathematics. Springer, 85–100.

ESSA, I. A., AND PENTLAND, A. 1997. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 19*, 7, 757–763.

ESSA, I., BASU, S., DARRELL, T., AND PENTLAND, A. 1996. Modeling, tracking and interactive animation of faces and heads: Using input from video. In *Proc. of Computer Animation 96*, 68–79.

GRINSPUN, E., HIRANI, A. N., DESBRUN, M., AND SCHRÖDER, P. 2003. Discrete shells. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, 62–67.

GUENTER, B., GRIMM, C., WOOD, D., MALVAR, H., AND PIGHIN, F. 1998. Making faces. In *Proc. of ACM SIGGRAPH 98*, 55–66.

HORN, B. K. 1987. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America 4*, 4, 629–642.

HYNEMAN, W., ITOKAZU, H., WILLIAMS, L., AND ZHAO, X. 2005. Human face project. In *ACM SIGGRAPH 05 Course Notes*.

IGARASHI, T., NISHINO, K., AND NAYAR, S. 2005. The appearance of human skin. Tech. Rep. CUCS-024-05, Department of Computer Science, Columbia University.

JONES, A., GARDNER, A., BOLAS, M., MCDOWALL, I., AND DEBEVEC, P. 2006. Performance geometry capture for spatially varying relighting. In *3rd European Conference on Visual Media Production (CVMP 2006)*.

KOBBELT, L., VORSATZ, J., AND SEIDEL, H.-P. 1999. Multiresolution hierarchies on unstructured triangle meshes. *Comput. Geom. Theory Appl. 14*, 1-3, 5–24.

KOCH, R. M., GROSS, M. H., CARLS, F. R., VON BÜREN, D. F., FANKHAUSER, G., AND PARISH, Y. 1996. Simulating facial surgery using finite element methods. In *Proc. of ACM SIGGRAPH 96*, 421–428.

LANIR, Y. 1987. Skin mechanics. In *Handbook of Bioengineering*, R. Skalak and S. Chien, Eds. McGraw-Hill, 11.1–11.25.

LI, H., ROIVAINEN, P., AND FORCHHEIMER, R. 1993. 3-D motion estimation in model-based facial image coding. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 15*, 6, 545–555.

MAGNENAT-THALMANN, N., KALRA, P., LÉVÊQUE, J. L., BAZIN, R., BATISSE, D., AND QUELEUX, B. 2002. A computational skin model: fold and wrinkle formation. *IEEE Trans. on Information Technology in Biomedicine 6*, 4, 317–323.

MEYER, M., DESBRUN, M., SCHRÖDER, P., AND BARR, A. H. 2003. Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and Mathematics III*, H.-C. Hege and K. Polthier, Eds. Springer-Verlag, Heidelberg, 35–57.

NEHAB, D., RUSINKIEWICZ, S., DAVIS, J., AND RAMAMOORTHI, R. 2005. Efficiently combining positions and normals for precise 3d geometry. *ACM Transactions on Graphics 24*, 3, 536–543.

NOH, J.-Y., AND NEUMANN, U. 1999. A survey of facial modeling and animation techniques. Tech. Rep. USC-TR-99-705, Univeristy of Southern Californina.

NOH, J.-Y., AND NEUMANN, U. 2001. Expression cloning. In *Proc. of SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, 277–288.

PIGHIN, F. H., SZELISKI, R., AND SALESIN, D. 1999. Resynthesizing facial animation through 3D model-based tracking. In *International Conference on Computer Vision (ICCV)*, 143–150.

SHI, L., YU, Y., BELL, N., AND FENG, W.-W. 2006. A fast multigrid algorithm for mesh deformation. *ACM Transactions on Graphics 25*, 3, 1108–1117.

SIFAKIS, E., NEVEROV, I., AND FEDKIW, R. 2005. Automatic determination of facial muscle activations from sparse motion capture marker data. *ACM Transactions on Graphics 24*, 3, 417–425.

SVOBODA, T., MARTINEC, D., AND PAJDLA, T. 2005. A convenient multicamera self-calibration for virtual environments. *Presence: Teleoper. Virtual Environ. 14*, 4, 407–422.

TERZOPOULUS, D., AND WATERS, K. 1993. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 14*, 569–579.

VENKATARAMAN, K., LODHA, S., AND RAGHAVAN, R. 2005. A kinematic-variational model for animating skin with wrinkles. *Computers & Graphics 29*, 5, 756–770.

VLASIC, D., BRAND, M., PFISTER, H., AND POPOVIĆ, J. 2005. Face transfer with multilinear models. *ACM Transactions on Graphics 24*, 3, 426–433.

WANG, Y., HUANG, X., LEE, C.-S., ZHANG, S., LI, Z., SAMARAS, D., METAXAS, D., ELGAMMAL, A., AND HUANG, P. 2004. High resolution acquisition, learning and transfer of dynamic 3-D facial expressions. *Computer Graphics Forum 23*, 3, 677–686.

WENGER, A., GARDNER, A., TCHOU, C., UNGER, J., HAWKINS, T., AND DEBEVEC, P. 2005. Performance relighting and reflectance transformation with time-multiplexed illumination. *ACM Transactions on Graphics 24*, 3, 756–764.

WEYRICH, T., MATUSIK, W., PFISTER, H., BICKEL, B., DONNER, C., TU, C., MCANDLESS, J., LEE, J., NGAN, A., JENSEN, H. W., AND GROSS, M. 2006. Analysis of human faces using a measurement-based skin reflectance model. *ACM Transactions on Graphics 25*, 3, 1013–1024.

WILLIAMS, L. 1990. Performance-driven facial animation. In *Proc. of ACM SIGGRAPH 90*, vol. 24, 235–242.

WU, Y., KALRA, P., AND MAGNENAT-THALMANN, N. 1996. Simulation of static and dynamic wrinkles of skin. In *Proc. of Computer Animation*, 90–97.

ZHANG, S., AND HUANG, P. 2006. High-resolution, real-time three-dimensional shape measurement. *Optical Engineering 45*, 12.

ZHANG, Y., AND SIM, T. 2005. Realistic and efficient wrinkle simulation using an anatomy-based face model with adaptive refinement. In *Computer Graphics International 2005*, 3–10.

ZHANG, L., SNAVELY, N., CURLESS, B., AND SEITZ, S. M. 2004. Spacetime faces: High resolution capture for modeling and animation. *ACM Transactions on Graphics 23*, 3, 548–558.