Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2009)
E. Grinspun and J. Hodgins (Editors)

Face/Off: Live Facial Puppetry

Thibaut Weise Hao Li Luc Van Gool Mark Pauly

ETH Zurich

Abstract

We present a complete integrated system for live facial puppetry that enables high-resolution real-time facial expression tracking with transfer to another person's face. The system utilizes a real-time structured light scanner that provides dense 3D data and texture. A generic template mesh, fitted to a rigid reconstruction of the actor's face, is tracked offline in a training stage through a set of expression sequences. These sequences are used to build a person-specific linear face model that is subsequently used for online face tracking and expression transfer. Even with just a single rigid pose of the target face, convincing real-time facial animations are achievable. The actor becomes a puppeteer with complete and accurate control over a digital face.

1. Introduction

Convincing facial expressions are essential to captivate the audience in stage performances, live-action movies, and computer-animated films. Producing compelling facial animations for digital characters is a time-consuming and challenging task, requiring costly production setups and highly trained artists. The current industry standard in facial performance capture relies on a large number of markers to enable dense and accurate geometry tracking of facial expressions. The captured data is usually employed to animate a digitized model of the actor's own face or transfer the motion to a different one. While recently released feature films such as *The Curious Case of Benjamin Button* demonstrated that flawless re-targetting of facial expressions can be achieved, film directors are often confronted with long turn-around times as mapping such a performance to a digital model is a complex process that relies heavily on manual assistance.

We propose a system for live puppetry that allows transferring an actor's facial expressions onto a digital 3D character in real-time. A simple, low-cost active stereo scanner is used to capture the actor's performance without requiring markers or specialized tracking hardware. A full 3D model of the actor's face is tracked at high spatial and temporal resolution, facilitating the accurate representation of face geometry and expression dynamics. Real-time performance is achieved through extensive pre-processing and a careful design of the online tracking algorithm to enable efficient GPU implementations. Pre-processing includes robust and accurate facial tracking for offline 3D model building and the construction of a simplified facial expression space from a large set of recorded facial expressions. For online capture, we simplify the tracking algorithm to its essentials and exploit the reduced dimensionality of the facial expression model. Real-time transfer of facial expressions onto a different face is achieved by a linear model based on preprocessed deformation transfer [SP04]. This allows plausible live animations of different characters, even when only a single rigid model of the target face is available. For example, ancient Roman statues can be brought to live (Figure 10).

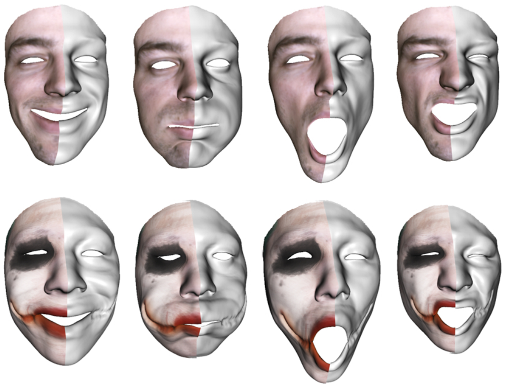

Figure 1: *Accurate 3D facial expressions can be animated and transferred in real-time from a tracked actor (top) to a different face (bottom).*

Markerless live puppetry enables a wide range of new applications. In movie production, our system complements existing off-line systems by providing immediate real-time feedback for studying complex face dynamics. Directors get to see a quick 3D preview of a face performance, including emotional and perceptual aspects such as the effect of the intended makeup (see Figure 1). In interactive settings such as

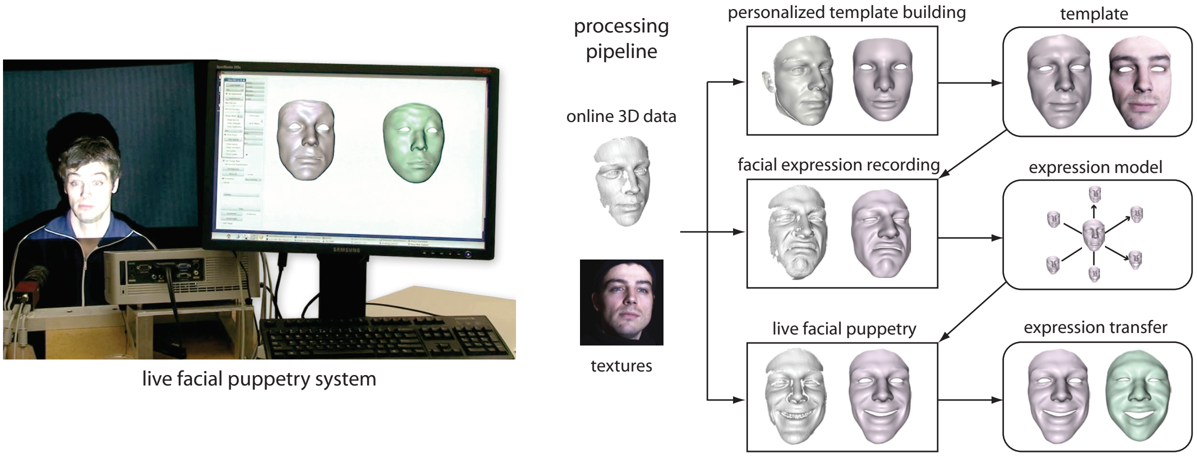

Figure 2: *Our system is composed of three main parts: Personalized template building, facial expression recording, and live facial puppetry. All components rely on input from a real-time structured light scanner. During template building a generic template is fit to the reconstructed 3D model of the actor's face. Dynamic facial expressions of the actor are then recorded and tracked using non-rigid registration. A person-specific facial expression model is constructed from the tracked sequences. The model is used for online tracking and expression transfer, allowing the actor to enact different persons in real-time.*

TV shows or computer games, live performances of digital characters become possible with direct control by the actor.

We put particular emphasis on the robustness of our tracking algorithm to enable processing of extended sequences of facial expressions. This allows building a reduced expression model that accurately captures the dynamics of the actor's face. Our method makes use of a generic template mesh to provide a geometric prior for surface reconstruction and to obtain consistent correspondences across the entire expression sequence. Besides specifying a few feature points for the initial non-rigid alignment of the template to the scanned actor's face, no manual intervention is required anywhere in our live puppetry pipeline. The automatic processing pipeline in combination with a minimal hardware setup for markerless 3D acquisition is essential for the practical relevance of our system that can easily be deployed in different application scenarios.

2. Related Work

Due to the huge amount of research in facial modeling and animation, we only discuss previous work most relevant to our online system and refer to [PW96] and [DN07] for a broader perspective. Facial animation has been driven by different approaches, in general using parametric [Par82, CM93], physics-based [TD90, SSRMF06], and linear models [Par72, BV99, VBPP05].

Linear Face Models. Linear models represent faces by a small set of linear components. Blendshape models store a set of key facial expressions that can be combined to create a new expression [Par72, PSS99, Chu04, JTDP03]. Statistical models represent a face using a mean shape and a set of basis vectors that capture the variability of the training data [Sir87, BV99, KMG04, VBPP05, LCXS07]. This allows modeling of a full population, while blendshape models are only suitable for an individual person. Global dependencies between different face parts arising in linear models are generally handled by segmenting the face into independent subregions [BV99, JTDP03].

Performance Capture. Performance-driven facial animation uses the performance of an actor to animate digital characters and has been developed since the early 80s [PB81]. Marker-based facial motion capture [Wil90, CXH03, DCFN06, LCXS07, BLB*08, MJC*08] is frequently used in commercial movie projects [Hav06] due to the high quality of the tracking. Drawbacks are substantial manual assistance and high calibration and setup overhead. Methods for offline facial expression tracking in 2D video have been proposed by several authors [PSS99, BBPV03, VBPP05]. Hiwada and co-workers [HMN03] developed a real-time face tracking system based on a morphable model, while Chai and colleagues [CXH03] use feature tracking combined with a motion capture database for online tracking. To the best of our knowledge, our system is the first real-time markerless facial expression tracking system using accurate 3D range data. [KMG04] developed a system to record and transfer speech related facial dynamics using a full 3D pipeline. However, their system has no real-time capability and requires some markers for the recording. Similarly, [ZSCS04] present an automatic offline face tracking method on 3D range data. The resulting facial expression sequences are then used in an interactive face modeling and animation application [ZSCS04, FKY08]. We enhance their method for offline face tracking and use the facial expression data for online tracking and expression transfer. While all the above methods use a template model, techniques exist that do not require any prior model and are able to recover non-rigid shape models from single view 2D image sequences [BHB00]. Although only a rough shape can be reconstructed, features such as eyes and mouth can be reliably tracked.

Expression Transfer. Noh and Neumann [NN01] introduced *expression cloning* to transfer the geometric deformation of a source 3D face model onto a target face. Sumner and Popovic [SP04] developed a generalization of this method suitable for any type of 3D triangle mesh. More closely related to our method, [CXH03] perform expression cloning directly on the deformation basis vectors of their linear model. Thus expression transfer is independent of the complexity of the underlying mesh. A different approach is taken by [PSS99, Chu04, ZLG*06] who explicitly apply tracked blendshape weights for expression transfer. The latter one does not require example poses of the source. [CB05] extended the method to reproduce expressive facial animation by extracting information from the expression axis of speech performance. Similarly, [DCFN06] map a set of motion capture frames to a set of manually tuned blendshape models and use radial basis function regression to map new motion capture data to the blendshape weights. In contrast, Vlasic and colleagues [VBPP05] use multi-linear models to both track faces in 2D video and transfer expression parameters between different subjects.

3. System Overview

Our facial puppetry system allows live control of an arbitrary *target* face by simply acting in front of a real-time structured light scanner projecting phase shift patterns. Geometry and texture are both captured at 25 fps. All necessary details of the scanning system can be found in [WLG07]. The actor's face is tracked online and facial expressions are transferred to the puppet in real-time. As illustrated in Figure 2, our system consists of three main components: Personalized template building, facial expression recording, and live facial puppetry. These components are described in more detail in Sections 5. 6, and 7, respectively. The key to online performance is to first record a set of facial expressions of the actor that are processed offline, and then build a simplified facial expression model specific to the actor for efficient online tracking and expression transfer. For template building, non-rigid registration is used to deform a generic template mesh to the 3D reconstruction of the actor's face. This personalized template is then tracked offline through a set of expression sequences. We take advantage of face specific constraints to make the tracking accurate and robust. The recorded expression sequences are used to build a simplified representation of the facial expression space using principal component analysis (PCA). The reduced set of parameters of the model enables efficient online tracking of the facial expressions. We propose a simple yet effective method for real-time expression transfer onto an arbitrary target face: We build a linear face model of the target face that uses the same parameters as the actor's facial expression model, reducing expression transfer to parameter transfer. To build the linear model we use deformation transfer [SP04] on the facial expression sequences of the actor and then find the optimal linear facial expression space for the target.

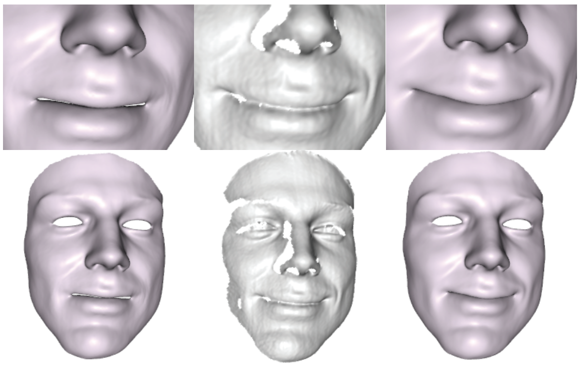

Figure 3: *The membrane model for vertex displacements (right) allows for more natural deformations than directly minimizing vertex displacement differences (left). The difference is particularly visible in the corners of the mouth.*

4. Deformable Face Model

Building the personalized template and recording facial expressions both require a non-rigid registration method to deform a face mesh to the given input geometry. Non-rigid registration methods typically formulate deformable registration as an optimization problem consisting of a mesh smoothness term and several data fitting terms [ACP03, ARV07, LSP08]. We represent deformations with displacement vectors $\mathbf{d}_i = \tilde{\mathbf{v}}_i - \mathbf{v}_i$ for each mesh vertex $\mathbf{v}_i \in \mathcal{V}$ and deformed mesh vertex $\tilde{\mathbf{v}}_i \in \tilde{\mathcal{V}}$. Deformation smoothness is achieved by minimizing a membrane energy $E_{\text{memb}} = \sum_{i\in\mathcal{V}} \|\Delta\mathbf{d}_i\|^2$ on the displacement vectors, using the standard cotangent discretization of the Laplace-Beltrami operator Δ (see [BS08]). The resulting linear deformation model is suitable for handling a wide range of facial deformations, while still enabling efficient processing of extended scan sequences. We prefer the membrane model over minimizing vertex displacement differences as in [ZSCS04], since the former results in more natural deformations as illustrated in Figure 3. Our experiments showed that these differences are particularly noticeable when incorporating dense and sparse constraints simultaneously in the optimization.

When personalizing the template (Section 5) we employ dense closest-point, and point-to-plane constraints [MGPG04], as well as manually selected sparse geometric constraints each formulated as energy terms for data fitting. For the automated expression recording, a combination of sparse and dense optical flow texture constraints [HS81] replaces the manually selected correspondences (Section 6). In both cases, face deformations are computed by minimizing a weighted sum of the different linearized energy terms described below. The resulting overdetermined linear system is sparse and can be solved efficiently via Cholesky decomposition [SG04].

5. Personalized Template Building

We generate an actor-specific template $\mathcal{M}$ by deforming a generic template mesh $\mathcal{M}_{\text{neutral}}$ to the rigid reconstruction of the actor's face (see Figures 2 and 4). Besides enabling a hole-free reconstruction and a consistent parameterization, using the same generic template has the additional benefit that we obtain full correspondence between the faces of the different characters.

Rigid Reconstruction. The face model is built by having the actor turn his head in front of the scanner with a neutral expression and as rigidly as possible. The sequence of scans is combined using on-line rigid registration similar to [RHHL02] to obtain a dense point cloud $\mathcal{P}$ of the complete face model. Approximately 200 scans are registered and merged for each face.

Template Fitting. We use manually labeled reference points $\mathbf{r}_j \in \mathcal{P}$ for initial rigid ICP registration of the generic template and the reconstructed face model (Figure 4). The reference points also provide sparse correspondence constraints in a subsequent non-rigid registration that deforms the template $\mathcal{M}_{\text{neutral}}$ towards $\mathcal{P}$ to obtain $\mathcal{M}$ using the sparse energy term $E_{\text{ref}} = \sum_j \left\|\tilde{\mathbf{v}}_j - \mathbf{r}_j\right\|_2^2$. Our manually determined correspondences are mostly concentrated in regions such as eyes, lips, and nose, but a few points are selected in featureless areas such as the forehead and chin to match the overall shape geometry. A total number of 24 reference points were sufficient for all our examples.

To warp the remaining vertices $\mathbf{v}_i \in \mathcal{M}_{\text{neutral}}$ toward $\mathcal{P}$, we add a dense fitting term based point-to-plane minimization with a small point-to-point regularization as described in [MGPG04]:

$$E_{\text{fit}} = \sum_{i=1}^{N} w_i (|\mathbf{n}_{\mathbf{c}_i}^\top (\tilde{\mathbf{v}}_i - \mathbf{c}_i)|^2 + 0.1 \, \|\tilde{\mathbf{v}}_i - \mathbf{c}_i\|_2^2). \tag{1}$$

The closest point on the input scan from $\tilde{\mathbf{v}}_i$ is denoted by $\mathbf{c}_i \in \mathcal{P}$ with corresponding surface normal $\mathbf{n}_{\mathbf{c}_i}$. We prune all closest point pairs with incompatible normals [RL01] or distance larger than 10 mm by setting the corresponding weights to $w_i = 0$ and $w_i = 1$ otherwise. Combining correspondence term and fitting term with the membrane model yields the total energy function $E_{\text{tot}} = E_{\text{fit}} + \alpha_{\text{ref}} E_{\text{ref}} + \alpha_{\text{memb}} E_{\text{memb}}$. The weights $\alpha_{\text{memb}} = 100$ and $\alpha_{\text{ref}} = 100$ are gradually reduced until $\alpha_{\text{memb}} = 5$ and $\alpha_{\text{ref}} = 1$. For all our examples we use the same scheduling for the energy weights (see also [LSP08]).

Texture Reconstruction. The diffuse texture map for the personalized face template is retrieved from the online rigid registration stage by averaging the textures of all input scans used for rigid reconstruction. The scan textures are the recorded video frames and have a resolution of 780 × 580 pixels. We use the projector's light source position to compensate for lighting variations assuming a dominantly diffuse reflectance model. Similarly, we remove points that are likely to be specular based on the half angle. The resulting texture map is oversmoothed, but sufficient for the tracking stage and has a resolution of 1024 × 768.

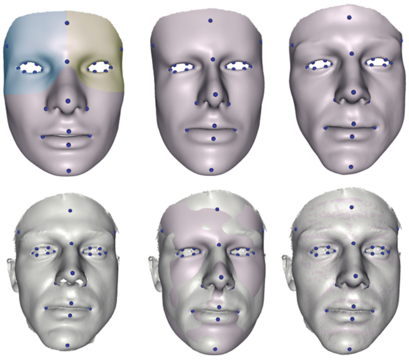

Figure 4: *Personalized template building: 24 manually labeled reference points are used for the rigid registration (left) and subsequent iterative non-rigid alignment (middle, right). Colors in the top left image indicate PCA segments.*

6. Facial Expression Recording

To generate the facial expression model we ask the actor to enact the different dynamic expressions that he plans to use for the puppetry. In the examples shown in our accompanying video, the actors perform a total of 26 facial expression sequences including the basic expressions (happy, sad, angry, surprised, disgusted, fear) with closed and open mouth as well as a few supplemental expressions (agitation, blowing, long spoken sentence, etc.). The personalized template $\mathcal{M}$ is then tracked through the entire scan sequence. For each input frame we use rigid ICP registration to compensate for global head motion yielding a rigid motion $(R, \mathbf{t})$. The generic non-rigid registration method described above then captures face deformations by adding to each rigidly aligned vertex $\bar{\mathbf{v}}_i = R\mathbf{v}_i + \mathbf{t}$ the displacement vector $\mathbf{d}_i = \tilde{\mathbf{v}}_i - \bar{\mathbf{v}}_i$. Note that a rigid head compensation is essential for robust tracking, since our globally elastic deformation model is a linear approximation of a non-linear shell deformation and thus cannot handle large rotations accurately [BS08]. Because of high temporal coherence between the scans, projective closest-point correspondences are used to compute $\mathbf{c}_i$ for Equation 1. In addition, we set w_i in E_{fit} to zero if $\mathbf{c}_i$ maps to a hole. Besides the dense geometric term E_{fit} and smoothness energy E_{memb}, we introduce a number of face specific additions, including dense and sparse optical flow texture constraints to improve accuracy and robustness of the tracking. Most notably, we explicitly track the mouth, chin, and eyelids.

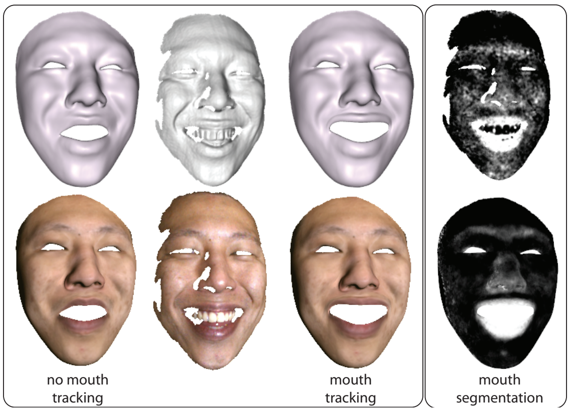

Figure 5: *Lip segmentation considerably improves the tracking results for the mouth region. The contrast enhancement due to lip classification can be seen on the right.*

Dense Optical Flow Constraints. Optical flow is used to enhance template tracking by establishing inter-frame correspondences from video data. Instead of using an independent optical flow procedure as in [ZSCS04], we directly include the optical flow constraints into the optimization, similar to model-based tracking methods [DM00]. We thus avoid solving the difficult 2D optical flow problem and integrate the constraints directly into the 3D optimization:

$$E_{\text{opt}} = \sum_{i=1}^{N} h_i \left(\nabla g_{t,i}^{\top} \, \Pi \left(\tilde{\mathbf{v}}_i^{t+1} - \tilde{\mathbf{v}}_i^{t} \right) + g_{t+1,i} - g_{t,i} \right) \quad (2)$$

where $g_{t,i} = g_t(\Pi(\tilde{\mathbf{v}}_i^t))$ is the image intensity at the projected image space position $\Pi(\tilde{\mathbf{v}}_i^t)$ of 3D vertex $\tilde{\mathbf{v}}_i^t$ at time t. Vertices at object boundaries and occlusions that pose problems in 2D optical flow are detected by projecting the template into both the camera and projector space and checking each vertex for visibility. We set the per vertex weight to $h_i = 1$ if visible and $h_i = 0$ otherwise. To ensure linearity in the optical flow energy E_{opt} we use a weak perspective camera model that we define as

$$\Pi(\mathbf{x}_i) = \frac{f}{\bar{z}_i} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} (R_{\text{cam}} \, \mathbf{x}_i + \mathbf{t}_{\text{cam}}), \quad (3)$$

where $\bar{z}_i$ is the fixed depth of the current template vertex $\tilde{\mathbf{v}}_i$, f the focal length, and $(R_{\text{cam}}, \mathbf{t}_{\text{cam}})$ the extrinsic camera parameters.

Optical flow is applied in a hierarchical fashion using a 3 level Gaussian pyramid, where low resolution video frames are processed first to allow for larger deformations. In each optimization step, we re-project all visible vertices to the image plane and recalculate the spatial image gradient $\nabla g_{t,i}$ using a standard Sobel filter, and the temporal derivative of the image intensity using forward differences.

Mouth Tracking. Optical flow is calculated sequentially and assumes that vertex positions in the previous frame are correct. This inevitably leads to drift, which is particularly noticeable in the mouth region as this part of the face typically deforms the most. We employ soft classification based on binary LDA [Fis36] to enhance the contrast between lips and skin. The normalized RGB space is used for illumination invariance. Soft classification is applied both to the scan video frame g_t and the rendering of the textured template g_t^* leading to two gray level images with strong contrast $\hat{g}_t$ and $\hat{g}_t^*$, respectively (Figure 5). Optical flow constraints between the template and the scan are then applied for the mouth region, in addition to the scan-to-scan optical flow constraints:

$$E_{\text{opt}}^{*} = \sum_{j \in \mathcal{V}_M} h_j \left(\nabla \hat{g}_{t,j}^{*\top} \; \Pi \left(\tilde{\mathbf{v}}_j^{t+1} - \tilde{\mathbf{v}}_j^{t} \right) + \hat{g}_{t+1,j} - \hat{g}_{t,j}^{*} \right) \quad (4)$$

where $\mathcal{V}_M$ is the set of vertices of manually segmented mouth and lips regions in the generic face template.

Thus mouth segmentation not only improves optical flow but also prevents drift as it is employed between scan and template texture which does not vary over time. The Fisher LDA is trained automatically on the template texture as both skin and lip vertices have been manually marked on the template mesh, which only needs to be performed once.

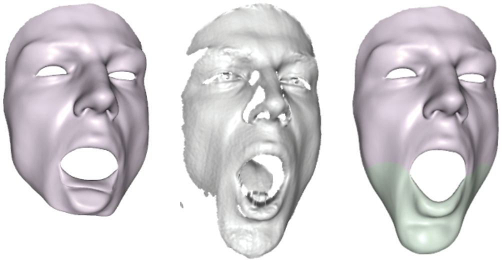

Figure 6: *Fast motion can lead to registration failure (left). Explicit rigid tracking of the chin significantly improves the robustness and convergence of the non-rigid registration algorithm (right). The chin region is marked in green and needs only be determined once on the generic template face.*

Chin Alignment. The chin often exhibits fast and abrupt motion, e.g., when speaking, and hence the deformable registration method can fail to track the chin correctly (Figure 6). However, the chin typically exhibits little deformation, which we exploit in an independent rigid registration for the chin part to better initialize the correspondence search for both geometry and texture. As a result, fast chin motion can be tracked very robustly.

Eyelid Tracking. Eyelids move very quickly and eye blinks appear often just for a single frame. Neither optical flow nor closest point search give the appropriate constraints in that case (Figure 7). However, the locations of the eye corners can be determined by a rigid transformation of the face. Assuming a parabolic shape of the eyelid on the eyeball, we can explicitly search for the best eyelid alignment us-

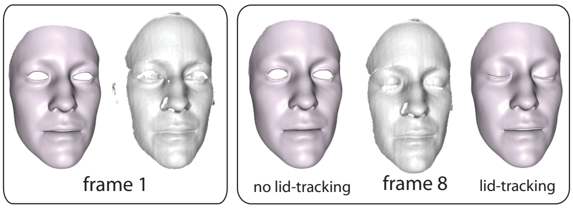

Figure 7: *Eyelid tracking enables the system to track eye blinks correctly. Without explicit eyelid tracking the system fails to track the closing of the eyes.*

ing texture correlation. The resulting correspondences are included into the optimization using a specific fitting term E_{eye} of closest-point constraints, similar to E_{fit}. A full statistical model [HIWZ05] was not required in our experiments, but could be easily incorporated into the framework.

Border constraints. The structured light scanner observes the geometry only from a single viewpoint. The sides of the face are mostly hidden and thus underconstrained in the optimization. For stability we fix the border vertices to the positions as determined by rigid registration.

Iterative Optimization. To improve convergence in the facial expression recording, we schedule $M = 5$ optimization steps for each input scan by recalculating closest points and using a coarse-to-fine video frame resolutions. After rigid alignment, we perform three steps of optimization with increasing resolution in the Gaussian pyramid for estimating image gradients and two optimization at the highest resolution. Each optimization step minimizes the total energy $E_{\text{tot}} = E_{\text{fit}} + \alpha_{\text{opt}} E_{\text{opt}} + \alpha^*_{\text{opt}} E^*_{\text{opt}} + \alpha_{\text{eye}} E_{\text{eye}} + \alpha_{\text{memb}} E_{\text{memb}}$ with constant energy weights $\alpha_{\text{opt}} = 5$, $\alpha^*_{\text{opt}} = 100$, $\alpha_{\text{eye}} = 0.5$, and $\alpha_{\text{memb}} = 10$.

7. Live Facial Puppetry

7.1. Online Face Tracking

Face tracking using the deformable face model is very accurate and robust, but computationally too expensive for online performance. Even though all constraints are linear and the resulting least-squares problem is sparse, solving the optimization requires approximately 2 seconds per iteration and 5 iterations per frame since the left hand side of a sparse but large linear system need to be updated in each step. In order to achieve real-time performance we employ PCA dimensionality reduction in facial expression space similar to [BV99]. We also manually segment the face into several subparts to break global dependencies. In our case this is the mouth and chin region, and symmetrically each eye and forehead (Figure 4).

The effectiveness of PCA depends on the quantity, quality, and linearity of the underlying data. Linearity has been demonstrated in previous PCA-based face models [Sir87, BV99, VBPP05]. One important advantage of our system is that we can easily generate a large number of high-quality training samples by recording a continuous sequence of facial expression tracked using our offline registration method (Figure 8). This allows us to accurately sample the dynamic expression space of the actor, which is essential for live puppetry. As opposed to previous methods based on linear dimension reduction, our approach uses dense batches of scans for the recording of each sequence.

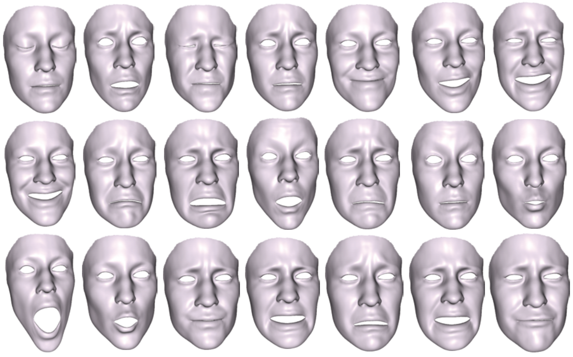

Figure 8: *A small subset of the roughly 250 expressions used for the generation of the PCA expression model for a specific actor.*

Online Registration. PCA represents the expression space by a mean face and a set of K components. At run-time, the system registers the facial expression by searching for the K coefficients that best match the data.

In principle, all the constraints used in offline face tracking can be included in the optimization. We found that due to the much lower dimensionality of the problem, projective closest-point correspondence search with point-plane constraints is usually sufficient to faithfully capture the dynamics of the face. However, we include rigid chin tracking to improve stability. We currently use $K = 32$ PCA components divided appropriately between the three face segments, which proved to be sufficient for representing more than 98% of the variability of the training data. More components did not add any significant benefit in tracking quality. We avoid discontinuities at the segment borders by pulling the solution towards the mean of the PCA model [BV99]. Online registration is achieved by optimizing $E_{\text{tot}} = E_{\text{fit}} + 0.1 \sum_{i=1}^{K} \|k_i\|_2^2$ where k_i are the PCA coefficients replacing the previous optimization variables $\mathbf{d}_i$.

All algorithms except the rigid registration are implemented on the GPU using shading languages and CUDA. With all these optimizations in place, our system achieves 15 frames per second, which includes the calculation of the structured light scanning system, rigid registration, chin registration, PCA-based deformation, and display.

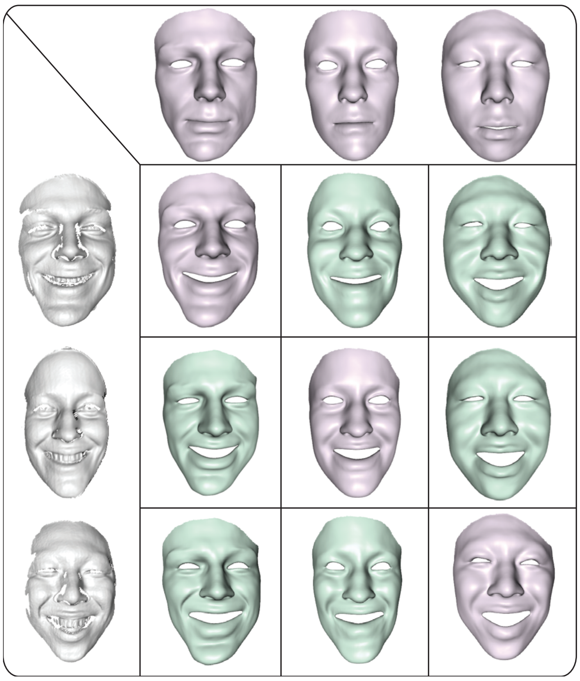

Figure 9: *Bringing a smile onto another face with real-time expression transfer. The tracked face is shown in magenta, the respective transfer onto the other faces is shown in green.*

7.2. Expression Transfer

Online face tracking allows the actor to control an accurate digital representation of his own face. Expression transfer additionally enables mapping expressions onto another person's face in real-time. The actor becomes a puppeteer.

Deformation Transfer. Sumner and Popovic [SP04] introduced a general method for mapping the deformation of a source mesh $\mathbf{S}$ onto an arbitrary target mesh $\mathbf{T}$ that we adapt to our setting. The deformation is expressed by the non-translational component of the affine transformation, i.e., the *deformation gradients* between a source mesh in its rest pose $\mathbf{S}$ and deformed state $\tilde{\mathbf{S}}$. The deformation gradients are then transfered to $\mathbf{T}$ by enforcing mesh connectivity via linear least-squares optimization. Since the template mesh provides correspondences, we can directly determine the deformation gradients between a face in neutral pose $\mathbf{S}_{\text{neutral}}$ and each captured expression $\mathbf{S}_i$. Thus, only a single target pose in neutral position $\mathbf{T}_{\text{neutral}}$ is required to determine all corresponding target expressions $\mathbf{T}_i$.

In our experiments we found that deformation transfer from one face to another yields very plausible face animations (Figure 9), giving the impression that the target face has the mimics of the actor. We note that we are not considering the problem of animating a different character with a non-human face. In that case models based on blendshapes [Chu04] seem more appropriate as deformations in the source and target face may not correlate geometrically.

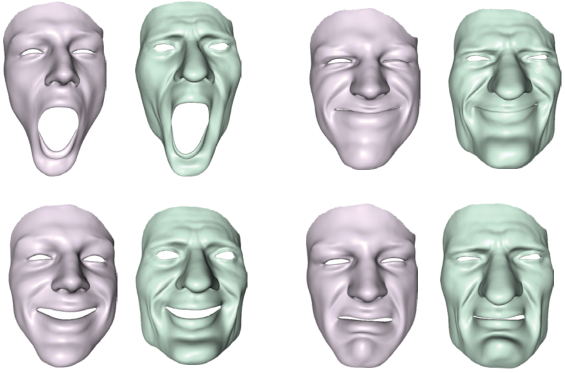

Figure 10: *Bringing an ancient Roman statue to live. The actor (magenta) can control the face of Caesar (green) that has been extracted from a laser scan of the statue.*

Linear Deformation Basis. Unfortunately, deformation transfer on the high resolution template mesh (25 K vertices) is too inefficient for real-time performance. To enable live puppetry, we generate a linear subspace that optimally spans the space of deformation transfer. For this purpose we compute the PCA bases of all $\bar{\mathbf{S}} = [\mathbf{S}_1 \ldots \mathbf{S}_n]$ and find the least squares optimal linear basis for the target face $\bar{\mathbf{T}} = [\mathbf{T}_1 \ldots \mathbf{T}_n]$ that is driven by the same coefficients $\mathbf{W}$ as the actor's PCA model. Thus, expression transfer is reduced to applying the coefficients of the actor PCA model to a linear model of the target shape.

Assume the training shapes of the actor can be expressed by the linear combination of PCA basis vectors $\tilde{\mathbf{S}}_i$:

$$\begin{bmatrix} \mathbf{S}_1 \\ \ldots \\ \mathbf{S}_n \end{bmatrix} = \begin{bmatrix} w_{11} & \ldots & w_{1k} \\ & \ldots & \\ w_{n1} & \ldots & w_{nk} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{S}}_1 \\ \ldots \\ \tilde{\mathbf{S}}_k \end{bmatrix} \tag{5}$$

We look for the linear basis $\left[\tilde{\mathbf{T}}_1 \ldots \tilde{\mathbf{T}}_k\right]^\top$ that best generates the target shapes $[\mathbf{T}_1 \ldots \mathbf{T}_n]^\top$ using the same weights:

$$\begin{bmatrix} \mathbf{T}_1 \\ \ldots \\ \mathbf{T}_n \end{bmatrix} = \begin{bmatrix} w_{11} & \ldots & w_{1k} \\ & \ldots & \\ w_{n1} & \ldots & w_{nk} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{T}}_1 \\ \ldots \\ \tilde{\mathbf{T}}_k \end{bmatrix} \tag{6}$$

We solve this linear least-squares problem using normal equations, where $\mathbf{W}$ is determined by simple projection of $\mathbf{S}_i$ onto the PCA bases $\tilde{\mathbf{S}}_i$ and $\left[\tilde{\mathbf{T}}_1 \ldots \tilde{\mathbf{T}}_k\right]^\top = \left[\mathbf{W}^\top \mathbf{W}\right]^{-1} \mathbf{W}^\top \bar{\mathbf{T}}$. The resulting basis vectors of the linear model are not orthogonal, but this is irrelevant for transfer. The training samples are already available from the offline facial expression tracking, and thus all expressions that are captured by the PCA model can also be transfered to the target face. For segmented PCA, each segment is transfered independently.

8. Results and Evaluation

Our system achieves accurate 3D facial tracking and real-time reconstruction at 15 fps of a complete textured 3D model of the scanned actor. In addition, we can transfer expressions of the actor at the same rate onto different face geometries. All computations were performed on an Intel Core Duo 3.0 Ghz with 2 GB RAM and a GeForce 280 GTX.

We demonstrate the performance of our approach with two male actors (Caucasian and Asian) and one female actress (Caucasian) as shown in Figure 9. Live puppetry is conducted between each actor and with two additional target models, a 3-D scanned ancient statue of Caesar (Figure 10) and a digitally sculpted face of the asian actor to impersonate the Joker (Figure 1). For both supplemental target meshes, no dynamic models were available. Building the personalized template requires rigid reconstruction of the actor's face and interactive reference point selection in order to warp the generic template onto the reconstruction. This whole process takes approximately 5 minutes. For each actor we capture 26 different facial expressions (another 5 minutes) as described in Section 6 resulting in approximately 2000 frames. We track the deformation of the personalized template over all input frames (10 seconds per scan) and sample 200 shapes at regular intervals. These are then used to compute the reduced PCA bases which requires additional 5 minutes. The extracted 200 face meshes are also used for deformation transfer on an arbitrary target model to generate the entire set of target expressions (about 30 minutes). All results are obtained with a fixed set of parameters and no manual intervention as described in the previous sections. Once set up, the online system can run indefinitely for extended live puppetry performances. Figure 12 shows an evaluation of the accuracy of the online tracking algorithm for a typical sequence with considerable facial deformations. The maximum error between the online registered template and the noisy scans mostly vary between 2 and 4 mm, while the root-mean-square error lies below 0.5 mm.

As illustrated in Figures 1 and 9 to 11, expression transfer achieves plausible facial expressions even though the target face geometries can differ substantially. Especially the facial dynamics are convincingly captured, which is best appreciated in the accompanying video. Note that all expression transfers are created with a single 3D mesh of the target face. No physical model, animation controls, or additional example shapes are used or required to create the animations.

Limitations. Our tracking algorithm is based on the assumption that the acquisition rate is sufficiently high relative to the motion of the scanned actor. Very fast motion or large occlusions can lead to acquisition artifacts that yield inaccurate tracking results. However, as Figure 13 illustrates, our system quickly recovers from these inaccuracies. Since online tracking can be achieved real-time, slight matching inaccuracies between the input scans and the template as illustrated in Figure 12 are visually not apparent.

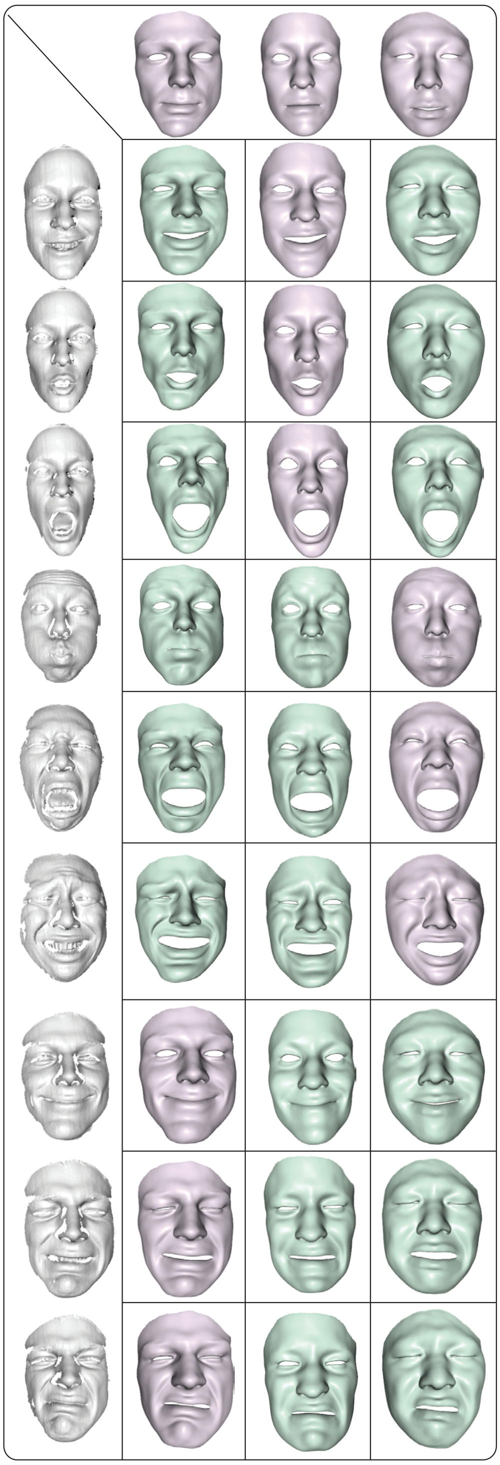

Figure 11: *Real-time expression transfer: The tracked face is shown in magenta, the transferred expression in green.*

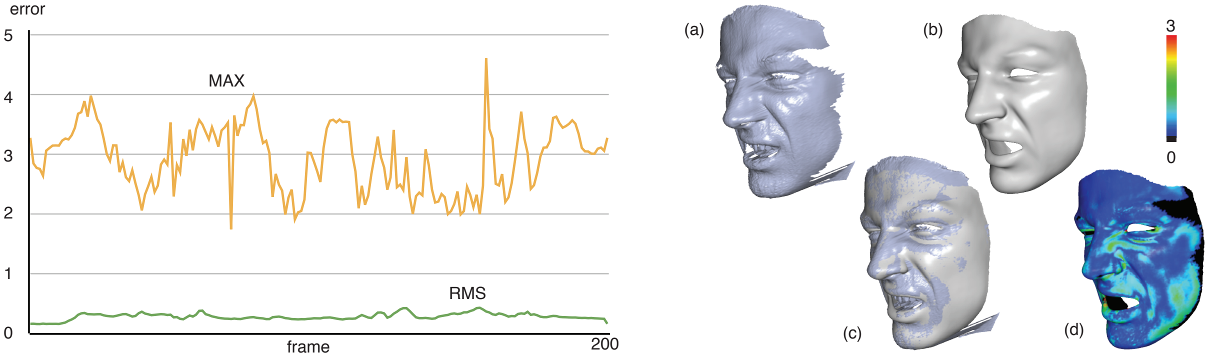

Figure 12: *Online tracking accuracy for a sequence of 200 frames of a speaking actor. The graph shows the maximum (MAX) and root-mean-square (RMS) distance between the input scans and the warped template. On the right we show the comparison between the scan (a) and corresponding template (b) that differs the most in the entire sequence. Their overlap is shown in (c) and the distance for each vertex is visualized in (d), where black denotes a hole region. Error measurements are in mm.*

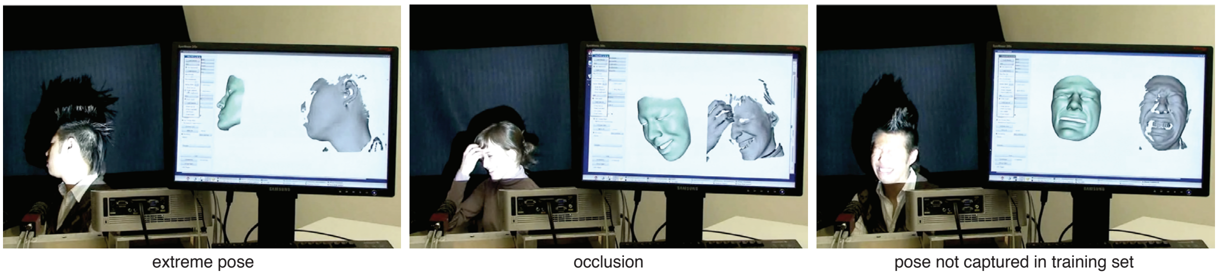

Figure 13: *The online tracking algorithm robustly handles difficult cases such as poses where the actor faces away from the camera (left), or occlusions that invalidate parts of the scan (middle). If the actor's expression is substantially different than any of the training samples, a plausible, but not necessarily accurate reconstruction is created (right). The gray image on the screen shows the acquired depth map, the green rendering is the reconstructed expression transferred to a different face.*

Our system does not capture all aspects of a real-live facial performance. For example, we do not explicitly track eyes, or represent the tongue or teeth of the actor. Similarly, secondary effects such as hair motion are not modeled in our system due to the substantial computation overhead that currently prevents real-time computations in the context of facial puppetry.

Facial expressions that are not recorded in the preprocessing step are in general not reproduced accurately in the online stage (Figure 13 right). This general limitation of our reduced linear model is mitigated to some extent by our face segmentation that can handle missing asymmetric expression. Nevertheless, high-quality results commonly require more than one hundred reconstructed scans to build an expression model that covers a wide variety of expressions suitable for online tracking. Fortunately, a 5-minute recording session per actor is typically sufficient, since the expression model can be reconstructed offline from a continuous stream of input scans.

Conclusion and Future Work. Our system is the first markerless live puppetry system using a real-time 3D scanner. We have demonstrated that high-quality real-time facial expression capture and transfer is possible without costly studio infrastructure, face markers, or extensive user assistance. Markerless acquisition, robust tracking and transfer algorithms, and the simplicity of the hardware setup, are crucial factors that make our tool readily deployable in practical applications. In future work we plan to enrich our system with a number of components that would increase the realism of the results. Realistic modeling of eyes, tongue, teeth, and hair, are challenging future tasks, in particular in the context of real-time puppetry. In addition, we want to investigate transfer methods that allow live control of substantially different models, such as non-human faces or cartoon characters.

Acknowledgments. This research has been funded in parts by SNF grant 200021-112122 and EU project HERMES FP6-027110. We are grateful to Pieter Peers, Leo Guibas, Bastian Leibe, and Chris Twigg for their valuable comments, and to Bine for being a great actress.

References

[ACP03] ALLEN B., CURLESS B., POPOVIĆ Z.: The space of human body shapes: reconstruction and parameterization from range scans. *ACM Trans. Graph. 22* (2003), 587–594.

[ARV07] AMBERG B., ROMDHANI S., VETTER T.: Optimal step nonrigid icp algorithms for surface registration. In *CVPR'07* (2007).

[BBPV03] BLANZ V., BASSO C., POGGIO T., VETTER T.: Reanimating faces in images and video. In *EUROGRAPHICS* (2003).

[BHB00] BREGLER C., HERTZMANN A., BIERMANN H.: Recovering non-rigid 3d shape from image streams. *CVPR'02 2* (2000), 2690.

[BLB*08] BICKEL B., LANG M., BOTSCH M., OTADUY M. A., GROSS M.: Pose-space animation and transfer of facial details. In *Proc. of SCA'08* (2008).

[BS08] BOTSCH M., SORKINE O.: On linear variational surface deformation methods. *IEEE Trans. on Visualization and Computer Graphics 14* (2008), 213–230.

[BV99] BLANZ V., VETTER T.: A morphable model for the synthesis of 3d faces. In *SIGGRAPH '99* (1999).

[CB05] CHUANG E., BREGLER C.: Mood swings: expressive speech animation. *ACM Trans. Graph. 24*, 2 (2005), 331–347.

[Chu04] CHUANG E.: *Analysis, Synthesis, and Retargeting of Facial Expressions*. PhD thesis, Stanford University, 2004.

[CM93] COHEN M. M., MASSARO D. W.: Modeling coarticulation in synthetic visual speech. In *Models and Techniques in Computer Animation* (1993).

[CXH03] CHAI J. X., XIAO J., HODGINS J.: Vision-based control of 3d facial animation. In *Proc. of SCA '03* (2003).

[DCFN06] DENG Z., CHIANG P.-Y., FOX P., NEUMANN U.: Animating blendshape faces by cross-mapping motion capture data. In *I3D '06: Proc. of the Symp. on Interactive 3D graphics and games* (2006).

[DM00] DECARLO D., METAXAS D.: Optical flow constraints on deformable models with applications to face tracking. *Int. J. Comput. Vision 38* (2000), 99–127.

[DN07] DENG Z., NEUMANN U.: *Computer Facial Animation: A Survey*. Springer London, 2007.

[Fis36] FISHER R. A.: The use of multiple measurements in taxonomic problems. *Annals Eugen. 7* (1936), 179–188.

[FKY08] FENG W.-W., KIM B.-U., YU Y.: Real-time data driven deformation using kernel canonical correlation analysis. *ACM Trans. Graph. 27* (2008), 1–9.

[Hav06] HAVALDAR P.: Sony pictures imageworks. In *SIGGRAPH '06: Courses* (2006).

[HIWZ05] HYNEMAN W., ITOKAZU H., WILLIAMS L., ZHAO X.: Human face project. In *SIGGRAPH '05: Courses* (2005).

[HMN03] HIWADA K., MAKI A., NAKASHIMA A.: Mimicking video: real-time morphable 3d model fitting. In *VRST '03: Proc. of the Symp. on Virtual Reality Software and Technology* (2003).

[HS81] HORN B. K. P., SCHUNK B. G.: Determining optical flow. *Artificial Intelligence 17* (1981), 185–203.

[JTDP03] JOSHI P., TIEN W. C., DESBRUN M., PIGHIN F.: Learning controls for blend shape based realistic facial animation. In *Proc. of SCA '03* (2003).

[KMG04] KALBERER G. A., MUELLER P., GOOL L. V.: Animation pipeline: Realistic speech based on observed 3d face dynamics. In *1st Europ. Conf. on Visual Media Prod.* (2004).

[LCXS07] LAU M., CHAI J., XU Y.-Q., SHUM H.-Y.: Face poser: interactive modeling of 3d facial expressions using model priors. In *Proc. of SCA '07* (2007).

[LSP08] LI H., SUMNER R. W., PAULY M.: Global correspondence optimization for non-rigid registration of depth scans. *SGP'08 27* (2008).

[MGPG04] MITRA N. J., GELFAND N., POTTMANN H., GUIBAS L.: Registration of point cloud data from a geometric optimization perspective. In *SGP '04* (2004).

[MJC*08] MA W.-C., JONES A., CHIANG J.-Y., HAWKINS T., FREDERIKSEN S., PEERS P., VUKOVIC M., OUHYOUNG M., DEBEVEC P.: Facial performance synthesis using deformation-driven polynomial displacement maps. *ACM Trans. Graph. 27* (2008), 1–10.

[NN01] NOH J.-Y., NEUMANN U.: Expression cloning. In *SIGGRAPH '01* (2001).

[Par72] PARKE F. I.: Computer generated animation of faces. In *ACM'72: Proceedings of the ACM annual conference* (1972).

[Par82] PARKE F.: Parameterized models for facial animation. *Computer Graphics and Applications, IEEE 2* (1982), 61–68.

[PB81] PLATT S. M., BADLER N. I.: Animating facial expressions. *SIGGRAPH Comput. Graph. 15*, 3 (1981), 245–252.

[PSS99] PIGHIN F., SZELISKI R., SALESIN D.: Resynthesizing facial animation through 3d model-based tracking. *In Proc. 7th IEEE Int. Conf. on Computer Vision 1* (1999), 143–150 vol.1.

[PW96] PARKE F. I., WATERS K.: *Computer facial animation*. A. K. Peters, Ltd., 1996.

[RHHL02] RUSINKIEWICZ S., HALL-HOLT O., LEVOY M.: Real-time 3d model acquisition. *ACM Trans. Graph. 21* (2002).

[RL01] RUSINKIEWICZ S., LEVOY M.: Efficient variants of the ICP algorithm. In *3DIM'01* (2001).

[SG04] SCHENK O., GÄRTNER K.: Solving unsymmetric sparse systems of linear equations with pardiso. *Future Gener. Comput. Syst. 20* (2004), 475–487.

[Sir87] SIROVICH L.; KIRBY M.: Low-dimensional precedure for the characterization of human faces. *Journal of the Optical Society of America A 4* (1987), 519–524.

[SP04] SUMNER R. W., POPOVIĆ J.: Deformation transfer for triangle meshes. In *SIGGRAPH '04* (2004).

[SSRMF06] SIFAKIS E., SELLE A., ROBINSON-MOSHER A., FEDKIW R.: Simulating speech with a physics-based facial muscle model. In *Proc. of SCA '06* (2006).

[TD90] TERZOPOULOS D. W. K.: Physically-based facial modeling, analysis and anmiation. *Journal of Visualization and Computer Animation 1* (1990), 73–80.

[VBPP05] VLASIC D., BRAND M., PFISTER H., POPOVIĆ J.: Face transfer with multilinear models. *ACM Trans. Graph. 24* (2005), 426–433.

[Wil90] WILLIAMS L.: Performance-driven facial animation. In *SIGGRAPH '90* (1990).

[WLG07] WEISE T., LEIBE B., GOOL L. V.: Fast 3d scanning with automatic motion compensation. In *CVPR'07* (2007).

[ZLG*06] ZHANG Q., LIU Z., GUO B., TERZOPOULOS D., SHUM H.-Y.: Geometry-driven photorealistic facial expression synthesis. *IEEE Trans. on Vis. and Comp. Graph. 12*, 1 (2006), 48–60.

[ZSCS04] ZHANG L., SNAVELY N., CURLESS B., SEITZ S. M.: Spacetime faces: High-resolution capture for modeling and animation. In *ACM Annual Conf. on Comp. Graphics* (2004).