Leveraging Motion Capture and 3D Scanning for High-fidelity Facial Performance Acquisition

Haoda Huang[*] Jinxiang Chai[†] Xin Tong[*] Hsiang-Tao Wu[*]
[*]Microsoft Research Asia [†]Texas A&M University

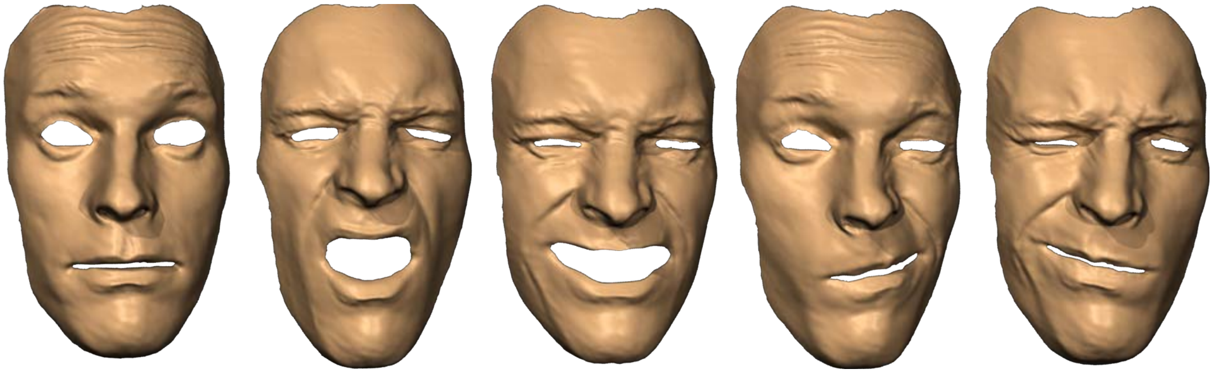


Figure 1: *Our system captures high-fidelity facial performances with realistic dynamic wrinkles and fine-scale facial details.*

Abstract

This paper introduces a new approach for acquiring high-fidelity 3D facial performances with realistic dynamic wrinkles and fine-scale facial details. Our approach leverages state-of-the-art motion capture technology and advanced 3D scanning technology for facial performance acquisition. We start the process by recording 3D facial performances of an actor using a marker-based motion capture system and perform facial analysis on the captured data, thereby determining a minimal set of face scans required for accurate facial reconstruction. We introduce a two-step registration process to efficiently build dense consistent surface correspondences across all the face scans. We reconstruct high-fidelity 3D facial performances by combining motion capture data with the minimal set of face scans in the blendshape interpolation framework. We have evaluated the performance of our system on both real and synthetic data. Our results show that the system can capture facial performances that match both the spatial resolution of static face scans and the acquisition speed of motion capture systems.

Keywords: Facial animation, face modeling, motion capture, facial data analysis, nonrigid surface registration, blendshape interpolation

Links: DL PDF

1 Introduction

One of the holy grail problems in computer graphics has been the realistic animation of the human face. Currently, creating realistic virtual faces often involves capturing facial performances of real people. A recent notable example is the movie *Beowulf* where prerecorded facial data were used to animate all characters in the film. Capturing detailed 3D facial performances, however, is difficult because it requires capturing complex facial movements at different scales. Large-scale deformations driven by muscles are paramount because they determine the overall shape and movement of the face. Medium-scale deformations such as skin wrinkling and folding are pivotal to understanding many of the expressive qualities in facial expressions. Finally, there is fine-scale stretching and compression of the skin mesostructure, producing subtle but perceptually significant cues.

Decades of research in computer graphics have explored a number of approaches to capturing 3D facial performances, including 3D scanning, marker-based motion capture, structured light systems, and image-based systems. Despite the efforts, acquiring high-fidelity facial performances remains a challenging task. For example, 3D face scanning systems (*e.g.*, [XYZ RGB Systems 2011]) can acquire high-resolution facial geometry such as pores, wrinkles, and age lines, but typically only for static poses. Marker-based motion capture systems such as Vicon [2011] can record dynamic facial movements with very high temporal resolution (up to 2000 Hz), but due to their low spatial resolution (usually 100 to 200 markers) they are not capable of capturing expressive facial details such as wrinkles. Recent progress in structured light systems [Zhang et al. 2004; Li et al. 2009] and multi-view stereo reconstruction systems [Bradley et al. 2010] have made it possible to capture 3D dynamic faces with moderate fidelity, resolution, and consistency, but their results still cannot match the spatial resolution of static face scans or the acquisition speed of marker-based motion capture systems.

The primary contribution of this paper is to introduce a novel acquisition framework for capturing high-fidelity facial performances with realistic dynamic wrinkles and fine-scale facial details (Figure 1). We leverage a marker-based motion capture system to record high-resolution dynamic facial movement and a 3D scanning system to record high-resolution static facial geometry of an actor performing a minimal set of preselected facial expressions. These two capturing technologies are complementary to each other as they focus on different aspects of facial performances. Though the general idea of combining 3D facial geometry with motion capture data is not new and dates back to [Williams 1990], our approach is unique in that we develop an automatic facial analysis process for selecting a minimal set of face expressions required for high-resolution facial reconstruction, thereby significantly reducing the time and effort required for 3D facial acquisition. Meanwhile, an optimal combination of high spatial resolution face scans and high temporal resolution motion capture data enables us to capture 3D facial performances that match both the spatial resolution of high-quality static face scans and the acquisition speed of marker-based motion capture systems.

Our high-fidelity facial performance acquisition is made possible by a number of technical contributions:

- A novel facial analysis technique that automatically determines a minimal set of face scans required for accurate facial performance reconstruction. This not only improves the accuracy of 3D facial reconstruction but also significantly reduces the time and effort required for 3D face scanning.
- A two-step registration process that builds dense, consistent surface correspondences across all the face scans. This is nontrivial because face scans often contain high-resolution facial details such as pores and wrinkles and a small misalignment between any two scans will result in unpleasant visual artifacts in the captured facial performance.
- Finally, an efficient facial reconstruction method that uses motion capture data to accurately interpolate the minimal set of the face scans in a blendshape interpolation framework. This step also requires the accurate registration of motion capture markers to all the face scans.

2 Background

In the past two decades, three distinctive methods have been developed for acquiring 3D facial performances: image-based facial capture, marker-based motion capture, and structured light based approach. Here we briefly discuss the advantages and disadvantages of each approach for 3D facial performance acquisition.

One appealing approach to capturing 3D dynamic faces is image-based facial capture, which deforms a 3D template mesh model to sequentially match input image sequences [Essa et al. 1996; DeCarlo and Metaxas 2000; Pighin et al. 1999]. Recent effort in this area has been focused on using prior models (*e.g.*, [Blanz et al. 2003; Vlasic et al. 2005]) to reduce the ambiguity of image-based facial deformations. Because these methods make use of generic templates or example-based priors models, the reconstructed geometry and motion do not approach the quality of person-specific captured data. More recent research has been focused on using multi-view stereo reconstruction techniques to improve the resolution and details of captured facial geometry. For example, Bradley and his colleagues [2010] reconstructed initial geometry using multi-view stereo reconstruction and used it to capture 3D facial movement by tracking the geometry and texture over time. While their approach produces much higher resolution than previous passive methods, their results still lack such details as pores and wrinkles. Concurrently, Beeler and his colleagues [2010] presented a different multi-view stereo reconstruction system for capturing the 3D geometry of a face in a single shot. Their system produces fine-scale facial details but the geometry that is recovered is qualitative and not metrically correct. More importantly, their system is limited to capturing static facial geometry and thus is not appropriate for our task.

An alternative approach for 3D facial capture is to use marker-based motion capture systems [Williams 1990; Guenter et al. 1998; Bickel et al. 2007], which robustly and accurately track a sparse set of facial markers and use them to deform a pre-scanned 3D facial mesh. Recent technological advances in motion capture equipment (*e.g.*, [Vicon Systems 2011]) have made it possible to acquire 3D motion data with stunningly high temporal resolution (up to 2000 Hz), but due to their low spatial resolution (usually less than 200 markers) they are not capable of capturing fine facial details such as wrinkles and bulges. Bickel and his colleges [2007] recently augmented the marker-based motion capture system with face paints and two synchronized video cameras for tracking medium-scale expression wrinkles. However, their approach, while powerful, is not appropriate for capturing small wrinkles (*e.g.*, nose wrinkles) and the fine-scale stretching and compression targeted in this paper.

Structured light systems are capable of capturing 3D models of dynamic faces in real time [Zhang et al. 2004; Ma et al. 2008; Li et al. 2009]. One notable example is the spacetime facial capture system developed by Zhang and his colleagues [2004]. They captured 3D facial geometry and texture over time and built the correspondences across all the facial geometries by deforming a generic face template to fit the acquired depth data using optical flow computed from image sequences. Recently, Ma and his colleagues [2008] achieved high-resolution facial reconstructions by interleaving structured light with spherical gradient photometric stereo using the USC Light Stage. More recently, Li and his colleagues [2009] captured dynamic depth maps with their realtime structured light system and fit a smooth template to the captured depth maps using embedded deformation techniques [Sumner et al.]. However, structured light systems cannot match the spatial resolution of static face scans (*e.g.*, XYZ RGB system [2011]) or the acquisition speed of marker-based systems (*e.g.*, Vicon [2011]).

A number of commercial systems have been developed for 3D facial performance capture in the entertainment industry. For example, Borshukov and his colleagues [2003] developed the *Universal Capture* system to recreate actors for *The Matrix Reloaded*. Their system deformed a laser-scanned 3D facial model by using optical flow fields computed from multiple image sequences. Alexander and his colleagues [2009] created a photo realistic facial modeling and animation system in the *Digital Emily Project*. Among all the systems, our approach is most similar to [Alexander et al. 2009]. Both systems combine motion capture data with a number of preselected face scans for high-fidelity facial performance capture. Our approach, however, is different in that we perform quantitative analysis on captured motion data and use it to automatically select a minimal set of facial expressions required for 3D facial performance capture, thereby minimizing the effort and time involved in the scanning process. In addition, instead of employing a manually intensive process for building surface correspondences across all the scans, which is not only timing consuming but also error prone [1], we develop a novel two-step registration process for building consistent dense surface correspondences across all the scans.

3 Overview

Our system acquires high-fidelity facial performances with realistic dynamic wrinkles and fine-scale facial details. The key idea of our system is to leverage high-fidelity motion capture data and high-resolution face scans for 3D facial performance acquisition.

[1] As reported in [Alexander et al. 2009], it takes an artist about three months to build all 75 blend shapes for 3D facial animation.

We start the process by recording facial performances of an actor using optical motion capture systems. We then perform facial analysis on the captured data and obtain a minimal set of face scans required for accurate facial reconstruction. Next, we register motion capture markers to all the face scans and build consistent dense surface correspondences across all the face scans. Lastly, we combine motion capture data with the minimal set of the face scans to reconstruct high-fidelity facial performances in the blendshape interpolation framework.

We choose to formulate the facial acquisition and reconstruction process in the blendshape interpolation framework because blendshape animation is one of the most widespread and successful facial animation techniques in both academia and industry [Pighin and Lewis 2006]. Mathematically, we represent high-fidelity facial performances $\mathbf{m}_t, t = 1,...,T$ as a weighted combination of the high-resolution face bases $\mathbf{b}_i, i = 1,...,K$:

$$\mathbf{m}_t = [\mathbf{b}_1...\mathbf{b}_K]\mathbf{w}_t, \quad \mathbf{w}_t \geq \mathbf{0}, \tag{1}$$

where the vector $\mathbf{w}_t$ represents K nonnegative weights to model the face mesh at frame t. One major benefit of blendshape interpolations is to decouple spatial details $\mathbf{b}_i, i = 1,...,K$ from temporal details $\mathbf{w}_t, t = 1,...,T$. This allows us to record high-resolution static facial geometry $\mathbf{b}_i, i = 1,...,K$ using 3D scanning systems and obtain high-resolution temporal details $\mathbf{w}(t), t = 1,...,T$ using marker-based motion capture systems.

Here we highlight the issues that are critical for the success of this endeavor and summarize our approach for addressing them.

Data Acquisition and Analysis. The first challenge is to scan a minimal set of the facial bases $\mathbf{b}_i, i = 1,...,K$ required for facial performance acquisition. Minimizing the number of face scans (K) is important because it reduces the time and effort spent on the scanning process. This problem, however, is challenging because ground truth facial performance data $\mathbf{m}_t, t = 1,...,T$ are not available. To address this challenge, we capture dynamic facial expressions $\mathbf{x}_t, t = 1,...,T$ using a marker-based motion capture system and utilize the captured data to automatically select a minimal set of face expressions $\mathbf{x}_{t_i}, i = 1,...,K$ required for accurate facial reconstruction. We scan the 3D geometry of the selected facial expressions $\mathbf{b}_i, i = 1,...,K$ by asking the actor to perform the same expressions as shown in the selected frames $\mathbf{x}_{t_i}, i = 1,...,K$.

Marker Mesh Registration. Combining motion capture data $\mathbf{x}_t, t = 1,...,T$ with face scans $\mathbf{b}_i, i = 1,...,K$ requires associating motion capture markers with every face scan. In other words, we need to register motion capture data $\mathbf{x}_{t_i}, i = 1,...,K$ at selected frames to the corresponding face scans $\mathbf{b}_i, i = 1,...,K$. This is a non-trivial registration problem because we need to consider both the rigid transformations between the two capturing systems and the non-rigid deformations caused by the possible differences between the "reference" expressions $\mathbf{x}_{t_i}$ and the "performed" expressions $\mathbf{b}_i$.

Face Scans Registration. Interpolations of the scanned faces require building dense, consistent surface correspondences across all the face scans $\mathbf{b}_i, i = 1,...,K$. Thus far nonrigid mesh registration remains difficult. Registering extreme facial expressions is particularly challenging because geometric details that appear in one scan might disappear in another one. In addition, face scans often contain high-resolution facial details such as pores and wrinkles. Even a small misalignment will result in unpleasant visual artifacts in the reconstructed facial performance. We propose a novel two-step registration algorithm for aligning all the face scans against each other. The two-step registration algorithm first registers large scale deformations across all the face scans and then refines the correspondences by segmenting each face scan into multiple regions and aligning each scan only with its closest neighbors (per region) via optical flow techniques.

Facial Performance Reconstruction. Our last challenge is how to combine the motion capture data with the face scans to reconstruct the facial performances $\mathbf{m}_t, t = 1,...,T$ in the blendshape interpolation framework.

We describe these components in more detail in the next sections.

4 Data Acquisition and Analysis

This section describes the process of facial data acquisition and analysis. We capture dynamic facial expressions $\mathbf{x}_t, t = 1,...,T$ using a marker-based motion capture system and analyze the captured data to determine a minimal set of facial expressions at selected frames $\mathbf{x}_{t_i}, i = 1,...K$, thereby obtaining a minimal set of high-resolution facial scans $\mathbf{b}_i, i = 1,...K$ for blendshape interpolation.

Dynamic Data Acquisition. We set up a twelve-camera Vicon motion capture system [2011] to record high-resolution dynamic facial movements $\mathbf{x}_t, t = 1,...,T$. We place about 100 retro-reflective markers on the face. The markers are arranged so that their movements capture the subtle nuances of the facial expressions. We set the acquisition rate to 240 frames per second. In addition, we synchronize a video camera with the optical motion capture system to record images of corresponding facial movements, which will later be used as references for scanning static facial expressions. The motion capture step produces a set of time-dependent 3D marker trajectories $\mathbf{x}_t, t = 1,2,...,T$ in the motion capture space, as well as synchronized reference facial expression images $\mathbf{I}_t, t = 1,...,T$.

Facial Data Analysis. Given the captured facial data $\mathbf{x}_t, t = 1,...,T$, our goal herein is to select a minimum set of static face scans $\mathbf{b}_i, i = 1,...K$ in such a way that high-fidelity facial performances $\mathbf{m}_t, t = 1,2,...T$ can accurately reconstructed via blendshape interpolations. This task is nontrivial because it requires determining not only the minimal number of blendshape bases (K) but also individual facial expressions ($\mathbf{b}_i, i = 1,...,K$) required for facial interpolations.

Our solution is to formulate the problem in an energy minimization framework and automatically select a minimum set of facial expressions by minimizing the reconstruction errors associated with the selected facial expressions:

$$\arg\min_{K,\{\mathbf{b}_i\},\{\mathbf{w}_t\}} \sum_{t=1}^{T} \|D[\mathbf{b}_1...\mathbf{b}_K]\mathbf{w}_t - \mathbf{x}_t\|^2 \quad \mathbf{w}_t \geq \mathbf{0}, \tag{2}$$

where the matrix D, which encodes the correspondences between the motion capture markers and the face scans, is used for converting the face scans into the motion capture data representation. However, the problem is still challenging because it requires optimizing both discrete parameters (K) and continuous parameters ($\mathbf{b}_i$). In addition, we need to estimate the unknown interpolation weights ($\mathbf{w}_t, t = 1,...,T$) for every frame.

To simplify the optimization problem, we choose to constrain the space of $\mathbf{b}_i, i = 1,...,K$ in a discrete space determined by the captured facial data $\mathbf{x}_t, t = 1,...,T$. This leads to the following optimization problem:

$$\arg\min_{K,\{t_i\},\{\mathbf{w}_t\}} \sum_{t=1}^{T} \|[\mathbf{x}_{t_1}...\mathbf{x}_{t_K}]\mathbf{w}_t - \mathbf{x}_t\|^2 \quad \mathbf{w}_t \geq \mathbf{0} \quad t_i \in 1,...,T. \tag{3}$$

In our implementation, we adopt a greedy strategy to find the optimal solution, initializing the bases number to zero and then incrementally increasing it by one until the reconstruction error falls below a user-specified threshold ϵ. The details of our analysis algorithm are described in Algorithm 1. We experimentally set the

Algorithm 1 : $\mathbf{B}^c$ = FacialPerformanceAnalysis($\mathbf{x}$, ε)

Input: dynamic facial data $\mathbf{x}_1,...,\mathbf{x}_T$ and tolerated reconstruction error ε

Output: a minimal set of static facial expressions $\mathbf{B}^c$=$\{\mathbf{x}_{t_1},\ldots,\mathbf{x}_{t_K}\}$

```
$\mathbf{B}^c = \{\}$ {initialize bases set}
while minError> $\varepsilon$ do
    for $t = 1 \ldots T$ do
        $\mathbf{B}^c_{tmp} = \mathbf{B}^c \cup \mathbf{x}_t$
        evaluate the error by applying NNLS to solve Equation (3)
        if error<minError then
            $minError \leftarrow error$
            $\mathbf{B}_{new} \leftarrow \mathbf{B}_{tmp}$
        end if
    end for
    $\mathbf{B}^c \leftarrow \mathbf{B}_{new}$
end while
return $\mathbf{B}^c$
```

threshold ε to 0.3% of the diagonal length of the bounding box containing all the facial markers. We start with the empty set $\mathbf{B}^c = \{\}$ and incrementally expand the set by minimizing the objective function described in Equation (3). We evaluate the objective function by computing the unknown weights via an efficient non-negative least squares solver (NNLS) described in [James and Twigg 2005].

Face Scans Acquisition. Given the minimal set of static facial expressions $\mathbf{B}^c = \{\mathbf{x}_{t_1},...,\mathbf{x}_{t_K}\}$ from the analysis algorithm, we can look up the synchronized reference images $\mathbf{I}_{t_1},...,\mathbf{I}_{t_K}$ and use them as reference facial expressions to scan the high-resolution facial meshes $\mathbf{b}_1,...,\mathbf{b}_K$. We use a Minolta VIVID 910 laser scanner to record high-resolution static facial geometry of an actor. During each scan, VIVID 910 acquires a face mesh with 100k to 200k vertices in about 2.5 seconds and achieves an accuracy of 0.008 mm on the x-y plane and 0.1 mm along the Z-axis. The scanned meshes $\mathbf{b}_i, i = 1,...,K$ are high-resolution and display subtle spatial details such as pores and wrinkles.

5 Marker Mesh Registration

This section discusses how to register motion capture data $\mathbf{x}_{t_i}, i = 1,...,K$ to the face scans $\mathbf{b}_i, i = 1,...,K$. This is a nontrivial task since we need to consider not only the rigid transformations between the two capturing systems but also the non-rigid deformations caused by differences between the "reference" expressions and the "performed" expressions.

In general, the nonrigid transformation between motion capture markers and face scans can be in an arbitrary form. In practice, the facial expressions performed by the actor are often very close to example expressions in the prerecorded facial data. Therefore, we assume the solution space of the nonrigid deformations lies in the subspace spanned by all the dynamic facial data. In other words, we regularize the deformation by modeling the low-resolution face scans $\mathbf{b}^c_i = D\mathbf{b}_i$ as a weighted combination of blendshape bases in the facial marker space: $[\mathbf{x}_{t_1}...\mathbf{x}_{t_K}]\mathbf{w}$.

We formulate the registration process as the following energy minimization problem:

$$\arg\min_{\mathbf{T},\mathbf{w}} dist^2([\mathbf{x}_{t_1}...\mathbf{x}_{t_K}]\mathbf{w}, \mathbf{T}(\mathbf{b}_i)), \quad (4)$$

where the function $\mathbf{T}$ is a rigid transformation function which models the global rotation and translation between the motion capture markers and the scanned meshes. The function $dist(\mathbf{x},\mathbf{b})$ evaluates the closest distances between the motion capture markers $\mathbf{x}$ and the surface of a face scan $\mathbf{b}$. However, direct estimation of rigid transformation parameters and deformation weights is prone to local minima and often produces poor results. We thus decouple the rigid transformation from nonrigid deformations and solve them in two sequential steps.

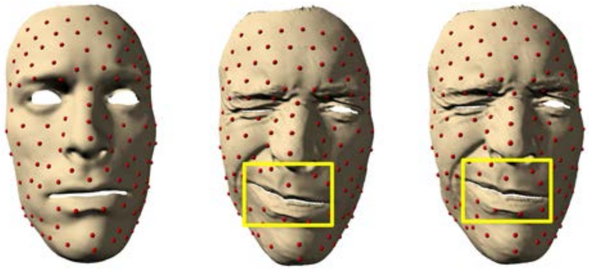

(a) (b) (c)

Figure 2: *Marker mesh registration: (a) motion capture markers located on a face scan with a neutral expression; (b) rigid registration results; (c) non-rigid registration results. Note that the non-rigid registration improves the registration results (e.g., highlighted region) because it considers the non-rigid deformations caused by differences between the reference expressions and the performed expressions.*

Rigid Registration. Rigid registration aligns the motion capture markers $\mathbf{x}_{t_i}$ to the corresponding face scan $\mathbf{b}_i$ by computing the rigid transformations T in such a way that minimizes the distances between the motion capture markers and the transformed face scan $\mathbf{T}(\mathbf{b}_i)$. This leads us to solve the following optimization problem:

$$\arg\min_{\mathbf{T}} dist^2(\mathbf{x}_t, \mathbf{T}(\mathbf{b}_i)). \quad (5)$$

We estimate the rigid transformations $\mathbf{T}$ with standard iterative closest point techniques (ICP). The corresponding point of each facial marker is found by its closest point on the transformed mesh. Figure 2(b) illustrates a transformed 3D mesh as well as the corresponding points of all the facial markers on the transformed mesh. Note that some markers on the bottom lip are misaligned because rigid registration does not consider the differences between the reference expressions and the performed expressions.

Non-rigid Registration. Non-rigid registration fixes the estimated transformations $\mathbf{T}$ and computes the nonrigid transformation $\mathbf{w}$ by minimizing the distances between the deformed motion capture markers, $[\mathbf{x}_{t_1}...\mathbf{x}_{t_K}]\mathbf{w}$, and the transformed mesh, $\mathbf{T}(\mathbf{b}_i)$. This can be achieved by solving the following optimization problem:

$$\arg\min_{\mathbf{w}} dist^2([\mathbf{x}_{t_1}...\mathbf{x}_{t_K}]\mathbf{w}, \mathbf{T}(\mathbf{b}_i)), \quad (6)$$

where the vector $\mathbf{w}$ models the nonrigid transform using a weighted combination of the blendshape bases in the mocap marker space. We extend the ICP techniques to iteratively estimate the combination vector $\mathbf{w}$. Briefly, we initialize the deformed facial markers $\mathbf{x}_w$ with $\mathbf{x}_{t_i}$. We then search the closest points of the deformed motion capture markers on the transformed mesh $\mathbf{T}(\mathbf{b}_i)$ and use them to update the combination vector $\mathbf{w}$ with least-squares fitting techniques. We iterate these two steps until convergence. Figure 2(b) and (c) show the corresponding motion capture markers on the face scan before and after the non-rigid registration step. As shown in the results, the nonrigid registration step improves the registration results between motion capture markers and face scans, *e.g.*, motion capture markers located on the bottom lip.

6 Face Scans Registration

After registering motion capture markers with the face scans, we obtain a sparse set of correspondences across all the face scans

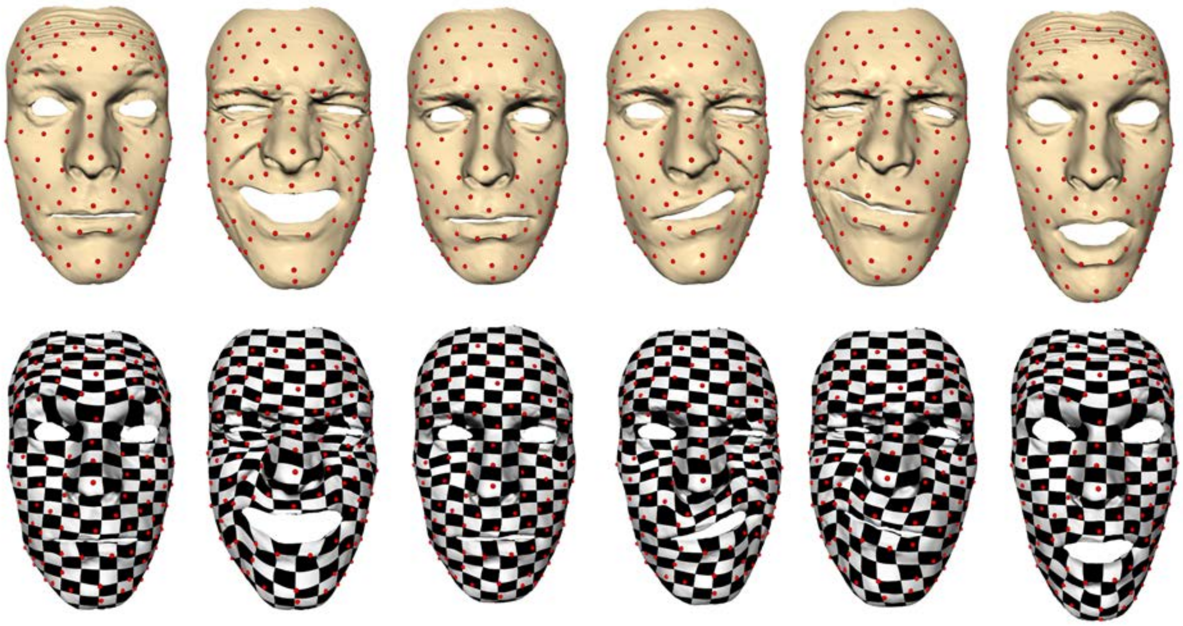


Figure 3: *(top) The marker mesh registration produces a sparse set of correspondences across all the face scans; (bottom) the two-step face scans registration produces dense consistent surface correspondences across all the face scans.*

(Figure 3(top)). However, blendshape interpolations require dense, consistent surface correspondences across all the scans. This section describes a novel two-step registration algorithm that achieves this goal (Figure 3(bottom)).

6.1 Large-scale Mesh Registration

We select one of the face scans as a template mesh $\mathbf{s}$ and build dense, consistent correspondences between the template mesh $\mathbf{s}$ and the face scans $\mathbf{b}_i, i = 1,...,K$ by deforming the template mesh to each face scan (Figure 4). Mathematically, we obtain the deformed template meshes $\mathbf{d}_i$ by minimizing the following objective function:

$$\arg\min_{\mathbf{d}_i} \; w_1|\mathbf{d}_i^c - \mathbf{b}_i^c|^2 + w_2\|L(\mathbf{v}_d) - L(\mathbf{v}_s)\|^2 + w_3 dist^2(\mathbf{d}_i, \mathbf{b}_i), \quad (7)$$

where the vectors $\mathbf{d}_i^c$ and $\mathbf{b}_i^c$ represent positions of facial markers on the deformed template mesh $\mathbf{d}_i$ and the target mesh $\mathbf{b}_i$, respectively. $\mathbf{v}_d$ and $\mathbf{v}_s$ represent the vertices on the deformed template mesh $\mathbf{d}_i$ and the template mesh $\mathbf{s}$, respectively. The operator $L(\mathbf{v})$ is the Laplace operator of the mesh model [Alexa 2003; Sorkine et al. 2004]. The function $dist(\mathbf{d}_i, \mathbf{b}_i)$ measures the distances between the deformed template mesh $\mathbf{d}_i$ and the target mesh $\mathbf{b}_i$, which can be evaluated by finding the closest points between the two mesh surfaces. Note that we factor out the global transformation of each scan before we perform the large-scale mesh registration process.

Intuitively, the first term measures how well motion capture markers on the deformed template mesh are mapped to the corresponding facial markers on the target mesh (*i.e.*, face scans). The second term is the Laplacian term, which preserves the fine details of the original template mesh and is mainly used to regularize the solution space. The last term ensures that the deformed template mesh $\mathbf{d}_i$ closely matches the target mesh $\mathbf{b}_i$. The weights w_1, w_2 and w_3 control the importance of each term, respectively. We initialize the deformed template mesh $\mathbf{d}_i$ by dropping the third term and deforming the template mesh to match the 3D positions of facial markers on the target mesh via Laplacian deformation techniques (Figure 4(b)). After that, we iteratively find the closest points between the deformed mesh and the target mesh and use them to deform the template mesh with least-squares techniques.

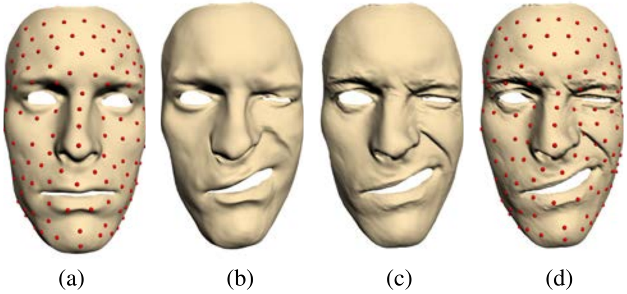


Figure 4: *Large-scale mesh registration deforms the template mesh to fit every face scan: (a) the template mesh* $\mathbf{s}$ *with motion capture markers; (b) the initial deformed template mesh* $\mathbf{d}_i$*; (c) the final deformed template mesh* $\mathbf{d}_i$*; (d) the target mesh* $\mathbf{b}_i$ *with motion capture markers obtained from marker mesh registration. Note that the final deformed mesh* $\mathbf{d}_i$ *preserves all the fine details in the target mesh* $\mathbf{b}_i$ *while still having the same topology as the template mesh* $\mathbf{s}$*.*

We experimentally set the weights of w_1 and w_2 to 10000 and 1.0, respectively. The optimization typically converges in three iterations because of very good initializations. During the iterations, we gradually increase the weight for the third term (w_3) from 0.1, to 0.5, to 2.0 in order to ensure that the final deformed template mesh can precisely match the target mesh. Figure 4(c) and (d) show a side-by-side comparison between the final deformed template mesh and the target mesh. The final deformed template mesh preserves all the fine details in the face scans. However, unlike the target meshes $\mathbf{b}_i, i = 1,...,K$, the final deformed template meshes $\mathbf{d}_i, i = 1,...,K$ have the same topology as the template mesh $\mathbf{s}$ and therefore are amenable for blendshape interpolations.

6.2 Fine-scale Mesh Registration

After large-scale mesh registration, we obtain dense, consistent surface correspondences across all the face scans $\mathbf{d}_i, i = 1,2,...K$.

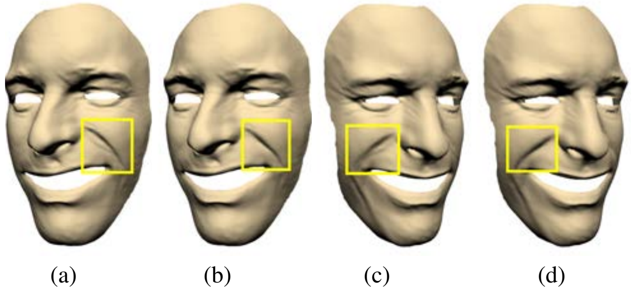


Figure 5: *Face scans registration with and without fine-scale mesh registration; (a)&(c) without fine scale mesh registration; (b)&(d) with fine scale mesh registration. Note the improvement of interpolation results in the highlighted region.*

However, large-scale mesh registration does not consider medium-scale deformations and fine-scale geometric details on the face scans, thereby producing unpleasant visual artifacts in facial interpolations. Figure 5(a) and (c) confirm this concern. To address this challenge, we introduce a new fine-scale mesh registration process to refine the dense correspondences across all the face scans. Figure 5(b) and (d) show fine-scale registration significantly reduces the interpolation artifacts in facial performance reconstruction.

So how can we refine the registration across all the face scans? One possible solution is to extract geometric details in all the face scans and minimize the misalignments of geometric details between every pair of scans. However, this idea does not work well for our application. First, pairwise registrations across all the face scans involve solving a large nonlinear optimization problem, which will inevitably be prone to local minima and is slow to converge. Second, geometric details are often expression dependent. The geometric details that appear in one face scan might disappear in another one and this causes problems for pairwise registrations because they often occur between two different facial expressions. Instead of performing pairwise registrations across all the face scans, we propose to register each scan only with its M closest scans. We hypothesize that scans close to each other often correspond to similar expressions and thus exhibit similar geometric details, which can then be used to register geometric details of the face scans.

This hypothesis, however, does not hold true in all instances, because two face scans close to each other over the whole face might still produce different geometric details in local regions. Figure 6(b) illustrates such an instance, where the face scan does not contain geometric details similar to its closest examples in the right forehead region. This leads us to partition the whole face into eight regions and perform geometric details registration in each local region. Figure 6(a) displays the eight partitioned regions on a neutral face. With region partitions, all the closest examples now exhibit similar geometric details in the forehead region (Figure 6(c)).

We formulate the region-based fine-scale mesh registration process as the following per-vertex optimization problem:

$$\arg\min_{\mathbf{d}'_1,\ldots,\mathbf{d}'_K} \sum_{i=1}^{K}\sum_{r=1}^{8}\sum_{m=1}^{M} \|\mathrm{G}(\mathbf{d}'_{i,r}) - \mathrm{G}(\mathbf{d}'_{i_m,r})\|^2, \;\; \mathbf{d}'_i \in Surf(\mathbf{d}_i), \quad (8)$$

where $\mathbf{d}'_i$ represents the refined mesh for the i-th face scan $\mathbf{d}_i$, $\mathbf{d}'_{i,r}$ represents the r-th region of the refined face scan $\mathbf{d}'_i$, and $\mathbf{d}'_{i_m,r}$ is the m-th neighbor of the refined face scan $\mathbf{d}'_{i,r}$. In our experiments, we set the number of closest examples (M) to three. Note that the goal of the fine-scale mesh registration is to refine the correspondences across all the face scans rather than change the underlying geometry and topology of the face scans. As a result, we keep the topology of the original face scans $\mathbf{d}_i$ and constrain the new vertices $\mathbf{d}'_i$ to a point on a surface of the old face scan $\mathbf{d}_i$: $\mathbf{d}'_i \in Surf(\mathbf{d}_i)$. The function $\mathrm{G}(\mathbf{d}')$ represents the geometry features extracted from the face mesh $\mathbf{d}'$. In our implementation, we choose the mean curvature values at each vertex as our geometry features. For each vertex, we fit a quadratic patch to all the vertices that are located within a particular distance to the vertex and then compute the mean curvature values as described in [Cazals and Pouget 2003]. To remove noise in the extracted geometry features, we threshold the feature values smaller than 0.05 to zero. Figure 7(a) exhibits the mean curvatures of one face scan.

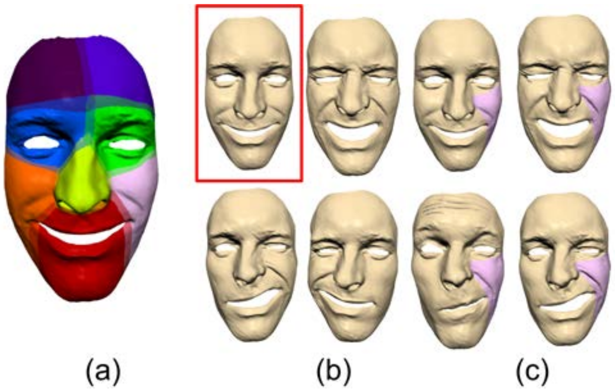


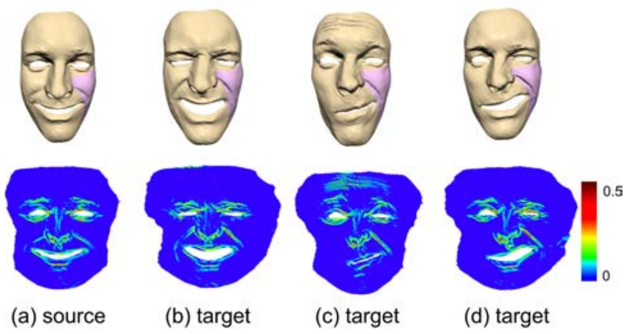
Figure 6: *Region-based mesh registration: (a) we partition the entire face into eight regions; (b)&(c) geometric details between one scan and its three closest scans without/with region partition.*

Given geometry features defined on the mesh, we solve the optimization described in Equation (8) iteratively with the overrelaxing algorithm. We initialize each refined mesh with the face scans obtained from large-scale mesh registration: $\mathbf{d}'_i = \mathbf{d}_i, i = 1,\ldots,K$. In each iteration, we sequentially update all the face meshes $\mathbf{d}'_i, i = 1,\ldots,K$ one by one. For each mesh $\mathbf{d}'_i$, the *LocalAlign* algorithm updates vertex positions of $\mathbf{d}'_i$ by fixing all the other meshes and minimizing the feature misalignments between the current mesh $\mathbf{d}'_i$ and its K-NN meshes:

$$\arg\min_{\mathbf{d}'_i} \sum_{r=1}^{8}\sum_{m=1}^{M} \|\mathrm{G}(\mathbf{d}'_{i,r}) - \mathrm{G}(\mathbf{d}'_{i_m,r})\|^2. \quad (9)$$

Due to the topology consistency between the different face scans $\mathbf{d}'_i$, adjusting vertex positions on one mesh changes the correspondences across all the meshes. The algorithm converges quickly. We stop the algorithm after three iterations in our experiments.

One remaining issue is how to update the mesh $\mathbf{d}'_i$ by minimizing the misalignments of geometric details between the current mesh and the rest of the meshes (Equation (8)). Our idea is to project the 3D face meshes onto the 2D image space and use optical flow algorithms to register the geometric details between the source image (*i.e.*, the current mesh) and the target image (*i.e.*, the K-NN meshes). Briefly, we project 3D vertices on the mesh onto a cylindrical surface and then unfold the cylindrical surface into an image. This allows us to build a one-to-one mapping between the image space and the mesh surface. For each region $r = 1,\ldots,8$, we sample geometry features of the current mesh to form a source image $I_{s,r}$ (Figure 7(a)). The geometry features are conceptually similar to color values in standard images. Similarly, we form a target image $I_{t,r}$ based on its three closest meshes (Figure 7(b)–(d)). Next, we register the source image with the target image using standard high-dimensional optical flow algorithms [Papenberg et al. 2006] by minimizing the color differences (*i.e.*, geometric feature misalignments) between the source and target images. Lastly, we project the computed optical flow back to the mesh surface and use



(a) source (b) target (c) target (d) target

Figure 7: *Cylindrical image mapping: (a) the source image obtained from a source face scan; (b)–(d) the target images obtained from three closest face scans. The pixel colors indicate the magnitude of mean curvatures.*

it to update the vertices of the current mesh accordingly. We explain the details of the algorithm below.

Cylindrical Mapping. Cylindrical mapping transforms the source mesh $\mathbf{d}'_{i,r}$ and its three closest meshes $\mathbf{d}'_{i_m,r}, m = 1,...,3$ into the source image $I_{s,r}$ and the target images $I_{t,r}$, respectively. We choose cylindrical mapping because cylindrical projection achieves maximal coverage of face features and introduces minimal distortions. Based on the bijective cylindrical projection mapping from the mesh surface to the cylindrical surface, we can construct the source image $I_{s,r}$ for region r by mapping all triangles within the region to the image plane and filling the mean curvature values of the projected pixels with scanline conversion techniques (Figure 7(a)). We copy the mean curvature value in each pixel three times to generate a three-dimensional vector so that the source image has the same dimensions as the target image.

To construct the target image $I_{t,r}$, we first pre-warp the geometry features on each of three closest meshes $\mathbf{d}'_{i_m,r}, m = 1,...,3$ to the source mesh $\mathbf{d}'_{i,r}$ by using the current correspondences between the source and the neighboring meshes. More specifically, we assign the mean curvature value of each vertex on a neighboring mesh to the corresponding vertex on the source mesh. After that, we fill the target image for each neighboring mesh with the same rasterization method described above (Figure 7(b)–(d)). Geometric feature prewarping achieves two major benefits. First, it allows us to stack geometric features from all three neighboring meshes into a single color image, where each channel encodes geometric features from one neighboring mesh. More importantly, prewarping reduces the magnitude of optical flow between the source image and the target image, thereby reducing the difficulty in image registration.

Image Registration. Given the source image $I_{s,r}$ and the target image $I_{t,r}$, we apply the optical flow algorithm described in [Papenberg et al. 2006] to solve the pixel correspondences between the two images. We choose the optical flow algorithm developed by [Papenberg et al. 2006] because of its ability to deal with large displacements, its robustness to noise, and its insensitivity to parameter variations. The output of the optical flow algorithm is an offset image $I_{o,r}(x,y) = (\delta x, \delta y)$, which records the offset of each pixel (x,y) in the source image to its corresponding pixel $(x+\delta x, y+\delta y)$ in the target image. We compute the optical flow for each region and obtain the offset images $I_{o,r}, r = 1,...,8$.

We compose the offset images of each region to form an offset image for the whole face. In the overlapping regions, we compute the offset of each pixel (x,y) as follows:

$$I_o(x,y) = \sum_r w_r(x,y) I_{o,r}(x,y) / \sum_r w_r(x,y), \qquad (10)$$

where $w_r(x,y)$ are the blending weights to ensure a smooth transition from one region to another one. We set the blending weights $w_r(x,y)$ to $1.0 - cos(\frac{\pi d}{2 l_d})$, where d is the closest distance from the pixel (x,y) to the boundary of the corresponding overlapping region and the threshold l_d is automatically set to the maximum width of the corresponding overlapping region.

3D Mesh Update. This step updates the vertex positions of the current mesh with the computed image displacements $I_o(x,y)$. For each vertex $\mathbf{v}$ on the current face mesh $\mathbf{d}'_i$, we project it onto the image space and obtain the corresponding image displacement $\boldsymbol{\delta}\mathbf{p} = [\delta x, \delta y]$ by looking up the offset image $I_o(x,y) = (\delta x, \delta y)$. We then project the offset vector $\boldsymbol{\delta}\mathbf{p}$ from the image space back to the mesh surface to obtain the 3D offset vector $\boldsymbol{\delta}\mathbf{v}$. During the projection process, we ensure the updated vertices are located on the original mesh $\mathbf{d}_i$. We now can update the mesh by moving its vertices $\mathbf{v}$ to the new positions $\mathbf{v} + \boldsymbol{\delta}\mathbf{v}$. To avoid triangle flips, we update the vertex positions by constraining the mesh update step with Laplacian deformation. We again solve a least-squares Laplacian deformation problem. Thus, we generate the final mesh by solving the following quadratic optimization problem:

$$argmin_{\mathbf{v}'} \|\mathbf{v}' - \mathbf{v} - \boldsymbol{\delta}\mathbf{v}\|^2 + \alpha \|L\mathbf{v}' - L\mathbf{v}\|^2, \qquad (11)$$

where $\mathbf{v}$ and $\mathbf{v}'$ are vertex positions on the current mesh and the refined mesh, respectively. L is the cotangent Laplacian matrix. We experimentally set the weight α to 1.0.

User Interaction. Automatic registration of fine-scale geometric details across all the face scans will not always produce high quality results, especially for face scans with high-frequency geometric details (*e.g.*, small wrinkles). To ensure high quality facial reconstruction, we constantly monitor the automatically registered face scans. When image registration errors are larger than a specific threshold, the registration process is not considered reliable. The user can always manually refine the registration results. The image registration process provides an efficient way to combine user interactions with an automatic registration process. The user can refine the registration result by specifying point correspondences or curve correspondences between any two face meshes, including the correspondence constraints as a part of the objective function for image registration, and restarting the optical flow estimation process.

7 Facial Performance Reconstruction

We now discuss how to combine the captured facial data $\mathbf{x}_t, t = 1,...,T$ with the topology-consistent face scans $\mathbf{d}'_i, i = 1,...,K$ to reconstruct the facial performance $\mathbf{m}_t, t = 1,...,T$. Because the face scans $\mathbf{d}'_i, i = 1,...,K$ might be slightly different from facial expressions shown in reference images $I_1,...,I_K$ or reference motion capture data $\mathbf{x}_{t_i}, i = 1,...,K$, we apply Laplacian deformation techniques to deform the face scans $\mathbf{d}'_i, i = 1,...,K$ to precisely match 3D positions of all facial markers at the selected frames $\mathbf{x}_{t_i}, i = 1,...,K$. The updated face scans $\mathbf{d}'_i, i = 1,...,K$ are then used as blendshape bases.

Blendshape interpolation also requires determining the blendshape weights $\mathbf{w}_t$ for every frame. We compute the blendshape weights by solving the following non-negative least squares problem:

$$\bar{\mathbf{w}}_t = \arg\min_{\mathbf{w}_t} \|D[\mathbf{d}'_1 ... \mathbf{d}'_K]\mathbf{w}_t - \mathbf{x}_t\|^2 \quad \mathbf{w}_t \geq \mathbf{0}, \qquad (12)$$

where the matrix D encodes the correspondences between the mo-

Subject	Sequence length	# of markers	# of meshes	Mesh resolution
Abu I	54s	97	30	80K
Abu II	46s	97	21	80K
Matt	40s	111	20	80K
Robert	49s	89	20	80K

Table 1: *Statistics of our data set.*

tion capture markers and the detailed facial meshes, which is obtained by the marker mesh registration process described in Section 5. Finally, we reconstruct 3D facial performances in the blendshape framework: $\mathbf{m}_t = [\mathbf{d}'_1 ... \mathbf{d}'_K]\bar{\mathbf{w}}_t, t = 1, ..., T$.

8 Experimental Results

We have tested our system on acquiring 3D facial performances of three subjects. Table 1 lists the parameters of all the data sets shown in the accompanying video, including the length of each motion sequence, the number of motion capture markers used for capturing, the number of face scans required for 3D reconstruction, and the resolution of the scanned meshes.

Validation. We validate our facial data analysis method with one set of facial performance data captured by [Bradley et al. 2010], which consists of 1620 frames. We select 100 markers in the ground truth facial performance data and extract the corresponding motion trajectories across the entire sequence. We then analyze the synthesized trajectories and select a minimal set of facial bases for facial performance reconstruction. For comparison's sake, we also reconstruct the facial data with a varying number of bases and compute the numerical errors between the reconstructed motion and the original data. We evaluate the error as the sum of Euclidean distances normalized by the diagonal length of the mesh bounding box. Figure 8 shows the reconstruction errors corresponding to both facial markers (shown in blue) and full-resolution 3D meshes (shown in red). As the number of bases increases, both reconstruction errors decrease accordingly. However, the errors stabilize after the number of bases is higher than 15. This confirms our facial analysis result. Please refer to the accompanying video for a side-by-side comparison between the ground truth data and the reconstruction result generated by 15 bases.

Results. Our system can capture realistic dynamic wrinkles and fine-scale facial details. Our results are best seen in the accompanying video. Figure 9 show several sample frames from our reconstruction results. In the accompanying video, we also show a side-by-side comparison between the captured 3D facial performance and the recorded video data. Our experiment results show that the reconstructed facial performance is consistent with the recorded video data and retains a lot of spatial-temporal facial details shown in the original video data.

Computational Time. We implemented our system in C++ on a PC with Intel Xeon E5520 2.27GHZ CPU and 12GB memory. For a typical data set that contains 1500 frames and 20 face meshes with 80K vertices, our system takes about 2-3 hours for facial data analysis, 0.2 hours for marker mesh registration, and about 5 hours for face scan registration. After the data is aligned, the high-fidelity face geometry for each frame can be reconstructed in real time.

9 Conclusion and Future Work

We present an end-to-end system for acquiring high-fidelity 3D facial performances. The proposed system combines the power of motion capture and 3D scanning technology. The quantitative analysis of motion capture data allows us to obtain a minimal set of face scans required for spatial-temporal facial performance reconstruction, thereby minimizing the time and effort for 3D scanning. Our results show that the system can capture high-fidelity 3D facial performances with large-scale facial deformations, realistic dynamic wrinkles, and subtleties of facial expressions.

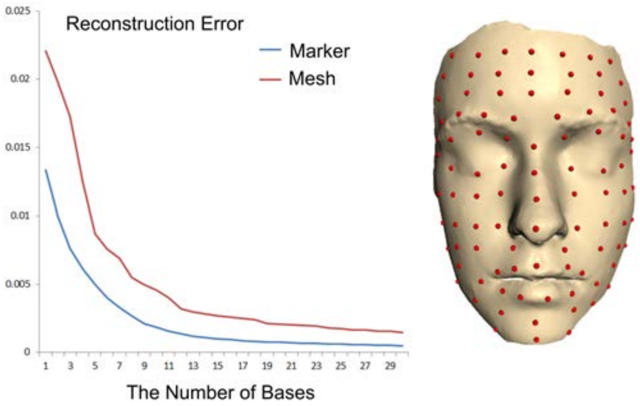


Figure 8: *Reconstruction errors of 100 dynamic marker points (in blue) and full facial geometry (in red) with a different number of face scans.*

Our system allows for capturing high-fidelity facial performances that match both the spatial resolution of static face scans and the acquisition speed of motion capture systems. The current system uses a Minolta VIVID 910 laser scanner to record high-resolution static facial geometry of an actor. We believe the quality of the final results can be further improved with a more accurate and higher resolution 3D scanning system such as XYZ RGB systems [2011] or Light Stage 5 [2009].

We perform our facial analysis on the captured dynamic facial data and select a minimal set of static facial expressions required for accurate facial performance reconstruction. However, an optimal combination of motion capture and 3D scanning also requires us to determine the minimal number of facial markers. Currently, we use about 100 markers to capture dynamic facial movement. However, we have not rigorously studied the minimal set of markers required to capture the high-resolution facial movement. In the future, we plan to study how increasing and decreasing the number of facial capture markers influences the quality of the captured motion.

The quality of the reconstructed facial performances highly depends on the accuracy of both marker mesh registration and face scans registration–even a small misalignment will result in unpleasant visual artifacts in the captured facial performance. Our experiments show that the fine-scale mesh registration algorithm, which minimizes the misalignments of geometric details across all the face scans, is extremely important to retaining geometric features (*e.g.*, facial lines and wrinkles) obtained from static face scans. In the future, we will continue to improve the accuracy of the registration algorithm. One possibility is to extract more effective geometric features for aligning fine-scale facial details. Another solution is to complement geometric features with image features (*e.g.*, textures, gradients, edges, or corners) by incorporating texture images into the current framework.

While this work focuses on capturing high-fidelity facial performances with realistic dynamic wrinkles and fine-scale facial details, in the future we are interested in modifying the captured facial data for new applications. For example, the captured facial performance data can be interactively edited to generate new facial expressions with direct manipulation interfaces or sketching interfaces [Lau et al. 2009], statistically generalized to achieve the

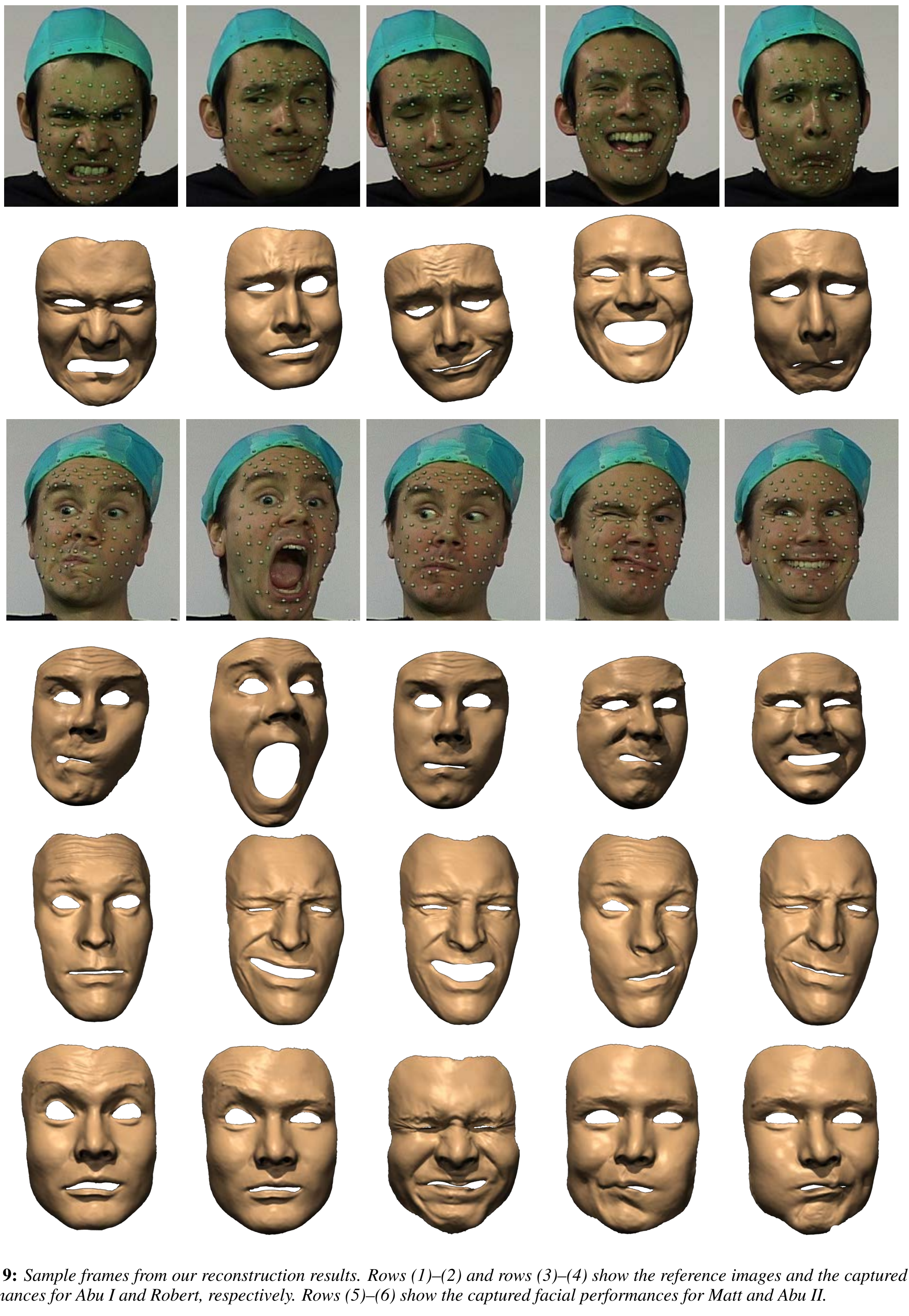


Figure 9: *Sample frames from our reconstruction results. Rows (1)–(2) and rows (3)–(4) show the reference images and the captured facial performances for Abu I and Robert, respectively. Rows (5)–(6) show the captured facial performances for Matt and Abu II.*

goals specified by the user [Chai and Hodgins 2007], interpolated to match low-dimensional signals extracted from vision-based interfaces [Chai et al. 2003], or retargeted to animate a different human avatar model [Li et al. 2010]. Direct applications of previous techniques might not work well for high-fidelity datasets because they are mainly focused on prerecorded facial data without dynamic wrinkles or fine scale facial details. One of the immediate directions for future work is, therefore, to investigate new methods for editing, retargeting, interpolating, and understanding high-fidelity facial data with dynamic winkles and fine-scale facial details.

Believable facial animation also requires realistic movements of the eyes and lips. The current system, however, is not appropriate to capture synchronized eye and lip movements. In the future, we are interested in extending the current system to capture lip synchronization and eye behavior. In addition, extending the method to full-body skin deformation acquisition is another interesting research topic.

Acknowledgement

The authors would like to thank Matt Callcut, Robert Walker and Abu for their facial performances and Professor Wencheng Wang from Institute of Software, Chinese Academy of Sciences for offering free access to their Minolta VIVID 910 laser scanner.

References

ALEXA, M. 2003. Differential coordinates for local mesh morphing and deformation. *The Visual Computer*. 19(2):105–114.

ALEXANDER, O., ROGERS, M., LAMBETH, W., CHIANG, M., AND DEBEVEC, P. 2009. The digital emily project: photoreal facial modeling and animation. In *ACM SIGGRAPH 2009 Courses*, 12:1–12:15.

BEELER, T., BICKEL, B., BEARDSLEY, P., SUMNER, B., AND GROSS, M. 2010. High-quality single-shot capture of facial geometry. *ACM Trans. Graph. 29*, 4, 40:1–40:9.

BICKEL, B., BOTSCH, M., ANGST, R., MATUSIK, W., OTADUY, M., PFISTER, H., AND GROSS, M. 2007. Multi-scale capture of facial geometry and motion. *ACM Trans. Graph. 26*, 3, 33:1–33:10.

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

BORSHUKOV, G., PIPONI, D., LARSEN, O., LEWIS, J. P., AND TEMPELAAR-LIETZ, C. 2003. Universal capture: image-based facial animation for "the matrix reloaded". In *ACM SIGGRAPH 2003 Sketches & Applications*, 1–1.

BRADLEY, D., HEIDRICH, W., POPA, T., AND SHEFFER, A. 2010. High resolution passive facial performance capture. *ACM Trans. Graph. 29*, 4, 41:1–41:10.

CAZALS, F., AND POUGET, M. 2003. Estimating differential quantities using polynomial fitting of osculating jets. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, 177–187.

CHAI, J., AND HODGINS, J. 2007. Constraint-based motion optimization using a statistical dynamic model. *ACM Transactions on Graphics 26*, 3, 8:1–8:9.

CHAI, J., XIAO, J., AND HODGINS, J. 2003. Vision-based control of 3D facial animation. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 193-206.

DECARLO, D., AND METAXAS, D. 2000. Optical flow constraints on deformable models with applications to face tracking. *International Journal of Computer Vision*. 38(2):99-127.

ESSA, I., BASU, S., DARRELL, T., AND PENTLAND, A. 1996. Modeling, tracking and interactive animation of faces and heads using input from video. In *Proceedings of Computer Animation Conference*. 68-79.

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

JAMES, D. L., AND TWIGG, C. D. 2005. Skinning mesh animations. *ACM Transactions on Graphics 24*, 3, 399–407.

LAU, M., CHAI, J., XU, Y.-Q., AND SHUM, H. 2009. Face poser: Interactive modeling of 3d facial expressions using facial priors. *ACM Transactions on Graphics 29*, 1, 3:1–3:17.

LI, H., ADAMS, B., GUIBAS, L. J., AND PAULY, M. 2009. Robust single-view geometry and motion reconstruction. *ACM Trans. Graph. 28*, 5, 175:1–175:10.

LI, H., WEISE, T., AND PAULY, M. 2010. Example-based facial rigging. *ACM Transactions on Graphics 29*, 4, 32:1–32:6.

MA, W.-C., JONES, A., CHIANG, J.-Y., HAWKINS, T., FREDERIKSEN, S., PEERS, P., VUKOVIC, M., OUHYOUNG, M., AND DEBEVEC, P. 2008. Facial performance synthesis using deformation-driven polynomial displacement maps. *ACM Trans. Graph. 27*, 5, 121:1–121:10.

PAPENBERG, N., BRUHN, A., BROX, T., DIDAS, S., AND WEICKERT, J. 2006. Highly accurate optic flow computation with theoretically justified warping. *Int. J. Comput. Vision 67*, 2, 141–158.

PIGHIN, F., AND LEWIS, J. P. 2006. Facial motion retargeting. In *ACM SIGGRAPH 2006 Courses*, SIGGRAPH '06.

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

SORKINE, O., COHEN-OR, D., LIPMAN, Y., ALEXA, M., RÖSSL, C., AND SEIDEL, H.-P. 2004. Laplacian surface editing. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, 175–184.

SUMNER, R. W., SCHMID, J., AND PAULY, M. Embedded deformation for shape manipulation. *ACM Trans. Graph. 26*, 3, 80:1–80:7.

VICON SYSTEMS, 2011. http://www.vicon.com.

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

WILLIAMS, L. 1990. Performance driven facial animation. In *Proceedings of ACM SIGGRAPH 1990*. 24(4):235-242.

XYZ RGB SYSTEMS, 2011. http://www.xyzrgb.com/.

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