# High-Quality Passive Facial Performance Capture using Anchor Frames

Thabo Beeler[1,2] Fabian Hahn[1] Derek Bradley[1] Bernd Bickel[1] Paul Beardsley[1]
Craig Gotsman[1,3] Robert W. Sumner[1] Markus Gross[1,2]
[1]Disney Research Zurich [2]ETH Zurich [3]Technion - Israel Institute of Technology

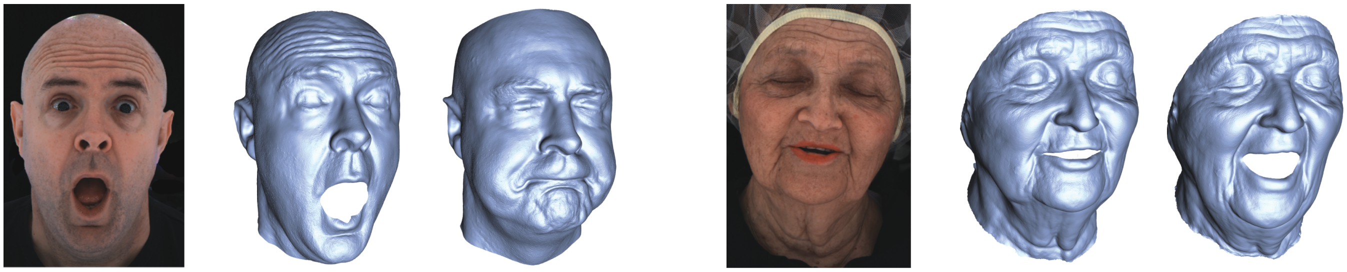

**Figure 1:** *High-quality facial performance capture for two actors. The resulting meshes are in full vertex correspondence.*

## Abstract

We present a new technique for passive and markerless facial performance capture based on *anchor frames*. Our method starts with high resolution per-frame geometry acquisition using state-of-the-art stereo reconstruction, and proceeds to establish a single triangle mesh that is propagated through the entire performance. Leveraging the fact that facial performances often contain repetitive subsequences, we identify *anchor frames* as those which contain similar facial expressions to a manually chosen reference expression. Anchor frames are automatically computed over one or even multiple performances. We introduce a robust image-space tracking method that computes pixel matches directly from the reference frame to all anchor frames, and thereby to the remaining frames in the sequence via sequential matching. This allows us to propagate one reconstructed frame to an entire sequence in parallel, in contrast to previous sequential methods. Our anchored reconstruction approach also limits tracker drift and robustly handles occlusions and motion blur. The parallel tracking and mesh propagation offer low computation times. Our technique will even automatically match anchor frames across different sequences captured on different occasions, propagating a single mesh to all performances.

**CR Categories:** I.3.3 [COMPUTER GRAPHICS]: Picture/Image Generation—Digitizing and scanning; I.3.5 [COMPUTER GRAPHICS]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems;

**Keywords:** Facial performance capture, space-time geometry reconstruction, motion capture.

**Links:** DL PDF Web Video

## 1 Introduction

The central role that facial motion plays in computer-generated animation, special effects, games, interactive environments, synthetic storytelling, and virtual reality makes facial performance capture a research topic of critical importance. However, the complexity of the human face as well as our skill and familiarity in interpreting real-life facial performances makes the problem exceptionally difficult. A performance capture result must exhibit a great deal of spatial fidelity and temporal accuracy in order to be an authentic reproduction of a real actor's performance. Numerous technical challenges such as robust tracking of facial features under extreme deformations and error accumulation over long capture sessions contribute to the problem's difficulty.

We present a reconstruction algorithm based on a multi-camera setup and passive illumination that delivers a single, consistent mesh deforming over time to precisely match an actor's performance. By incorporating a high-quality 3D reconstruction technique [Beeler et al. 2010], the mesh exhibits visually realistic pore-level geometric detail. Our results demonstrate that our system is robust to expressive and fast facial motions, reproducing extreme deformations with minimal drift. Our system requires no makeup so that temporally varying texture can be derived directly from the captured video. And, the computation is parallelizable so that long sequences can be reconstructed efficiently using a multi-core implementation.

Our high-quality results derive from two technical innovations. First, we employ a robust tracking algorithm that integrates tracking in image space and uses the integrated result to propagate a single reference mesh to each target frame. This strategy yields results superior to mesh-based tracking techniques for a number of reasons: (a) The image data typically contains much more detail, facilitating more accurate tracking. (b) The problem of error propagation due to inaccurate tracking in image space is dealt with in the same domain in which it occurs. (c) There is no complication of distortion due to parameterization, a technique used frequently in mesh processing algorithms. Additionally, because the image-space tracking is computed for each camera, multiple hypotheses are propagated forward in time. If one flow computation develops inaccuracies, the others can compensate.

Although our image-space tracker is accurate for short sequences, the eventual accumulation of integration error when reconstructing long capture sessions is unavoidable unless special care is taken.

Our second contribution addresses this issue by employing an "anchor frame" concept that relies on the observation that a lengthy facial performance will contain many frames that are similar in appearance. For example, when speaking, the face naturally returns to the resting pose between sentences or during speech pauses. Our method defines one frame as a reference frame and then marks all other frames similar to the reference as anchor frames. Due to the similarity, our image tracker can compute the flow from the reference to each anchor independently and with high accuracy. Our system can then treat each sequence between two consecutive anchors independently, integrating the tracking from both sides and enforcing continuity at the anchor boundaries. The accurate tracking of each anchor frame prevents error accumulation in lengthy performances. And, since the computation of the track between two anchors is independent, the algorithm can be parallelized across multiple cores or CPUs. Our method can use anchor frames that span multiple capture sessions of the same subject on different occasions without any additional special processing. This can be used to "splice" and "mix and match" unrelated clips, adding a powerful new capability to the editorial process.

## 2 Related Work

Data-driven facial animation has come a long way since the marker-based techniques introduced over two decades ago [Williams 1990; Guenter et al. 1998]. Current state of the art methods are now passive and almost fully automatic [Bradley et al. 2010; Popa et al. 2010]. However, a technique for capturing highly-detailed expressive performances that completely avoids temporal drift has yet to be realized. In this work, we take a step towards this goal with an anchored reconstruction approach. In this section we review the related work on facial geometry and motion capture, starting with the techniques that fit parametric face models to images, followed by active lighting and marker-based techniques, and finally discussing more recent passive reconstruction methods.

**Fitting Faces to Images.** One approach to face capture is to start with a deformable face model (or template) and then determine the parameters that best fit the model to images or videos of a performing actor [Li et al. 1993; Essa et al. 1996; DeCarlo and Metaxas 1996; Pighin et al. 1999; Blanz et al. 2003]. Using this approach the approximate 3D shape and pose of the deforming face can be determined. However the deformable face tends to be very generic, so the resulting animations often do not resemble the captured actor. The face model must also be low-resolution for the fitting methods to be tractable, so it is usually not possible to obtain the fine details that make a performance expressive and realistic.

**Markers and Active Light.** A common approach for performance capture is to track a sparse number of hand-placed markers or face paint using one or more video cameras [Williams 1990; Guenter et al. 1998; Lin and Ouhyoung 2005; Bickel et al. 2007; Furukawa and Ponce 2009]. While these techniques can yield robust tracking of very expressive performances and are usually suitable for a variety of lighting environments, the manual placement of markers can be tedious and invasive. Furthermore, the markers must be digitally removed from the videos if face color or texture is to be acquired. Also, the marker resolution is naturally limited, and detailed pore-scale performance capture has not been demonstrated with this approach.

An alternative to placing markers on the face is to project active illumination on the subject using one or more projectors [Wang et al. 2004; Zhang et al. 2004]. While this approach requires less manual setup, it can be equally invasive to the actor. Acquiring face color also poses a problem with these methods, as uniform illumination must be temporally interleaved with the structured light, sacrificing temporal resolution. A related active-light technique is proposed by Hernandez and Vogiatzis [Hernández and Vogiatzis 2010], who use tri-colored illumination with both photometric and multi-view stereo to obtain facial geometry in real-time, however without temporal correspondence. Finally, combining markers and structured light with a light stage has proven to yield impressive facial performance capture results [Ma et al. 2008; Alexander et al. 2009].

Other researchers have recently used controllable lighting in a light stage setup for facial performance capture. Wilson et al. [2010] establish temporal correspondence for images under the changing light conditions of spherical gradient illumination. This allows them to combine stereo with photometric normal maps in a temporally consistent way to generate detailed facial geometry and performance capture. Fyffe et al. [2011] propose a comprehensive facial performance capture system that aims to reconstruct both geometry as well as detailed reflectance information using gradient illumination and high-speed cameras. However they do not address temporal correspondence, which is the main focus of our work.

Unlike the marker-based and active light approaches, our method can capture detailed expressive performances in full temporal correspondence, with an entirely passive approach.

**Passive Capture.** Recently, research has focused on passive reconstruction, without requiring markers, structured light or expensive hardware. Beeler et al. [2010] reconstruct pore-scale facial geometry for static frames. Bradley et al. [2010] perform passive performance capture with automatic temporal alignment, but they lack pore-scale geometry and fail when confronted with expressive motions. Other passive deformable surface reconstruction techniques have been applied to faces [Wand et al. 2009; Popa et al. 2010]. Wand et al. [2009] reconstruct from point cloud data by fitting a template model. However, their method tends to lead to loss of geometric detail which is necessary for realistic facial animation. This is partially resolved in recent work by Popa et al. [2010] who use a gradual change prior in a hierarchical reconstruction framework to propagate a mesh structure across frames. Current industry leading systems for facial performance capture include Mova's CONTOUR Reality Capture[1] which requires fluorescent makeup, and the passive system of Dimension Imaging 3D[2].

**Pure Geometry Approaches.** While image data, due to its superior resolution and detail, can be an enormous help in tracking 3D points over time, it is not always available, and sometimes just a sequence of 3D point clouds or incompatibly-triangulated 3D meshes are the input. A number of authors have addressed the problem of tracking a 3D mesh over time based on pure geometry. An early approach was described by Anuar and Guskov [2004]. They start with an initial template mesh that is then propagated through the frames of the animation, based on a 3D adaptation of the Bayesian multi-scale differential optical flow algorithm. Since this flow is invariant to motion in the tangent plane, it is not able to eliminate "swimming" artifacts during the sequence.

Inspired by the deformation transfer method of Sumner and Popovic [2004], which allows to "copy and paste" mesh geometry from one shape to another, Winkler et al. [2008] present an optimization method to track triangle geometry over time combining terms measuring data fidelity, and preservation of triangle shape. Shape preservation is achieved by the use of mean-value barycentric coordinates. This also addresses motion in the tangent plane, eliminating the artifacts present in the results of Anuar and Guskov.

Tracking triangle geometry is intimately related to the problem of *cross-parameterization*, or *compatible remeshing* of

[1] www.mova.com

[2] www.di3d.com

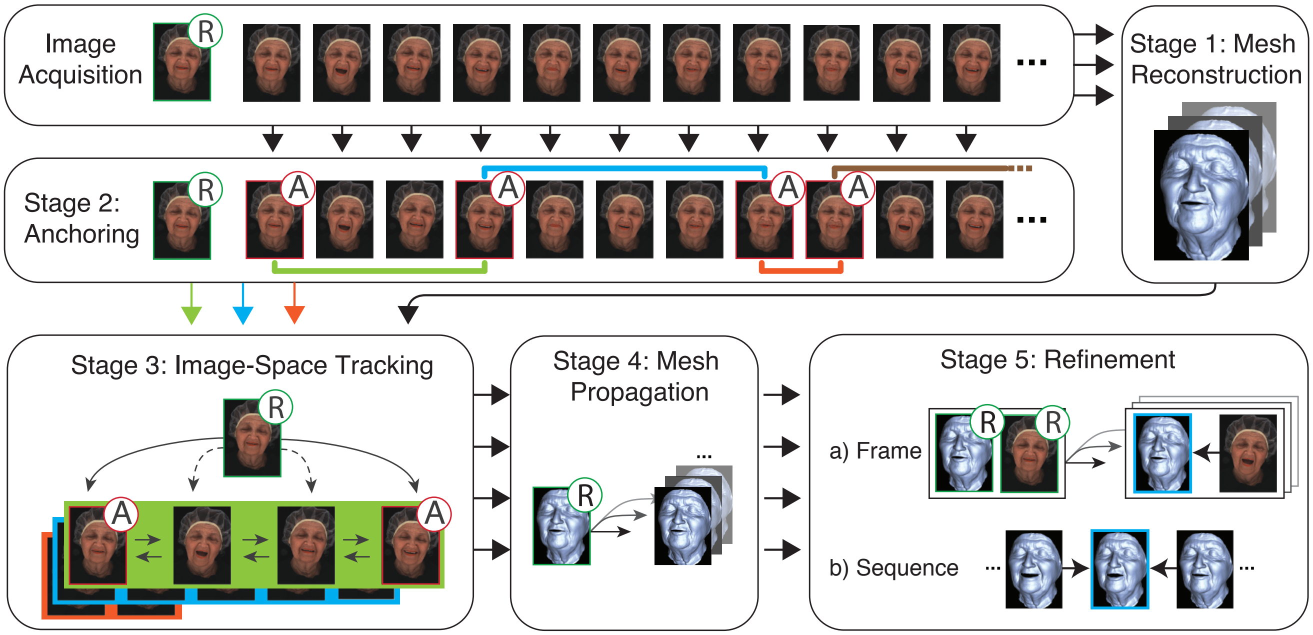

**Figure 2:** *System overview. Image acquisition is followed by mesh reconstruction (Stage 1) and anchor frames are detected to partition the sequence (Stage 2). The image-space tracking step matches the reference frame to all frames in the sequence (Stage 3), and then the reference mesh is propagated to each frame (Stage 4). Finally, the meshes are refined for a high-quality result (Stage 5).*

shapes [Kraevoy and Sheffer 2004], where the objective is to impose a triangle mesh representing one shape onto another, in a manner which minimizes distortion between the two. Bradley et al. [2008] have used this approach to track deforming garment geometry, however their method does not extend to faces since human skin does in fact distort during a performance.

More recent work by Sharf et al. [2008] tracks pure geometry over time using a volumetric representation. The main idea behind their method is that the volume of an object should be approximately constant over time, thus the flow must be "incompressible". This assumption, along with other standard continuity assumptions, regularizes the solution sufficiently to provide a good track.

In our work we explore anchor-based reconstruction with robust image-space tracking, where similar facial expressions in a sequence form anchor frames that allow us to limit temporal drift, match and reconstruct sequences in parallel, and establish correspondences between multiple sequences of the same actor captured on different occasions. To our knowledge, ours is the first approach to provide all of these features.

## 3 Passive Facial Performance Capture

This section describes our method for passive facial performance capture. The input is a sequence of frames of the face, where a 'frame' is a set of $n$ images acquired at one timestep. The output is a sequence of 3D meshes, one per frame, which move and deform in correspondence with the physical activity of the face. Each mesh vertex corresponds to a fixed physical point on the face and maintains that correspondence throughout the sequence within the bounds of error. Computation of the output meshes takes into account the image data, a prior on spatial coherence of the surface, and a prior on temporal coherence of the dynamically deforming surface. Figure 2 illustrates the five stages in the method:

- **Stage 1: Computation of Initial Meshes.** Each frame in the sequence is processed independently to generate a first estimate of the mesh for that frame.
- **Stage 2: Anchoring.** One frame is manually identified as the reference frame (marked "R" in Fig. 2). Frames with similar image appearance (similar face expression and head orientation) are detected automatically and labelled as anchor frames (marked "A" in Fig. 2 and Fig. 3). Anchor frames will provide a way to partition the complete sequence into clips of frames for the processing in Stage 3.
- **Stage 3: Image-Space Tracking.** The goal of this stage is to track image pixels from the reference frame to each frame in the sequence. The process starts by tracking image pixels from the reference frame to the anchor frames, which is straightforward because the image appearance in both frames is, by definition, similar. This will guide the tracking of image pixels from the reference frame to all other frames in the sequence. Tracking to the non-anchor frames is performed sequentially, starting from the nearest anchor frames.
- **Stage 4: Mesh Propagation.** The tracked image pixels obtained in Stage 3 provide a way to propagate the reference mesh, which is the mesh computed for the reference frame in Stage 1, to all frames in the sequence. We use the term *mesh propagation* of the reference mesh to mean the computation of new positions in space of the mesh vertices, in correspondence with physical movement of the face.
- **Stage 5: Mesh Refinement.** Previous stages generated a propagation of the reference mesh to each frame in the sequence. This deforming mesh sequence provides an initial estimate of the face motion, which is refined to enforce consistency with the image data while applying priors on spatial and temporal coherence of the deforming surface.

The individual stages are described in more detail below.

*Terminology* - A frame $F^t$ is the collection of the images $I_c^t$ of all cameras $c = 1..n$ at time $t$. Tracking image pixels from frame $F^t$ to frame $F^{t'}$ will be shorthand for tracking image pixels in each image pair $(I^t, I^{t'})_c$ of all cameras $c$.

### 3.1 Stage 1: Computation of Initial Meshes

We begin by recording an actor's performance from $n$ different viewpoints, captured with uniform illumination. Each frame in the sequence is processed independently to generate a mesh for the face using the 3D reconstruction method in [Beeler et al. 2010]. This gives us single-shot geometry for each frame with visually realistic pore-level details. There is no temporal correspondence in the resulting sequence of meshes, i.e. their mesh structure (number of vertices and triangulation structure) is totally unrelated. The goal of subsequent stages will be to generate a mesh sequence that is compatible, i.e. share a common vertex set and mesh structure, based on the initial meshes.

### 3.2 Stage 2: Anchoring

One frame in the sequence is identified as the reference frame. This is currently performed manually, however this step could be automated by analyzing the data and then automatically picking a repetitive pose. Frames with similar image appearance (similar face expression and orientation) are detected and labelled as anchor frames. Anchor frames are used to partition a sequence of frames into clips as shown in Figure 3. For example, the reference frame can be chosen with the face in its natural rest expression, and this will produce anchor frames along the sequence whenever the face returns to a rest (or similar) expression. Anchoring will be utilized in Stage 3 where the image-space tracking operates at the level of clips, eliminating the need to track a lengthy sequence of frames.

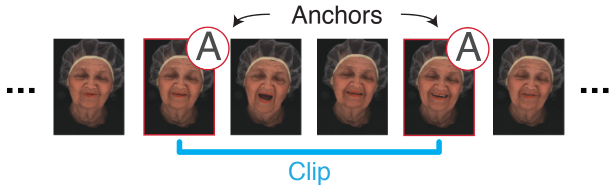

**Figure 3:** *Frames that are similar to the reference frame are labelled as anchors. A clip is a sequence of frames bounded by two anchor frames (an anchor frame might be both the last frame of one clip and the first frame of the next clip).*

#### 3.2.1 Motivation

Drift (or error accumulation) in tracking is a key issue when processing a long sequence. Anchor frames provide a way to decompose the sequence into clips, effectively allowing multiple starting points for the processing but still having a common root in the reference frame. An immediate benefit of this technique is that it naturally allows parallelization of the computation. But the more important benefit is that it results in periodic resets that prevent the accumulation of drift when tracking along the sequence. The technique also confines catastrophic failures in tracking to the frames in which they occur, which could arise from occlusion, motion blur, or the face outside the image.

#### 3.2.2 Identifying Anchor Frames

Our method for using the reference frame to determine anchor frames proceeds as follows:

- Detect a feature set $S_c$ in image $I_c^R$ in the reference frame, for all cameras $c = 1..n$. There is no requirement to detect the same physical face features in the different images. Our features correspond to a uniform sampling of the images with a sample rate of 0.05 (every 20th pixel). While we found these simple features sufficient, more sophisticated features such as SIFT [Lowe 2004] could be used.
- Perform correspondence matching of $S_c$ between $I_c^R$ and $I_c^t$ in the target frame $F^t$, for cameras $c = 1..n$. We employ normalized cross-correlation with $9 \times 9$ windows centered on each feature pixel.
- Compute an error score $E$ as the sum of the cross-correlation scores over all features in all feature sets $S_c$. Label the target frame as an anchor frame if $E$ is below a predefined threshold.

Figure 4 shows the variation in $E$ for a few example frames.

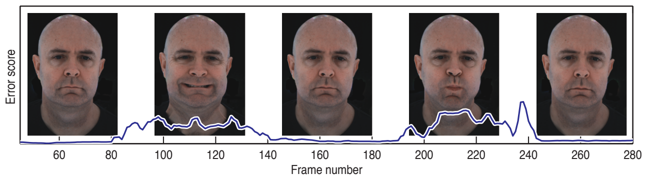

**Figure 4:** *The reference frame is compared with all frames in the sequence. Frames where the image appearance is similar are labelled as anchor frames.*

### 3.3 Stage 3: Image-Space Tracking

The goal of this stage is to track image pixels from the reference frame to each frame in the sequence. As stated previously, tracking pixels between frames is shorthand for tracking pixels in each camera in the frames. The basic method of finding correspondence is block-based matching using normalized cross-correlation.

However, there are two distinct situations for matching. The anchor frames identified in Stage 2 have, by definition, similar image appearance to the reference frame. Thus image correspondence is straightforward. The unanchored frames, however, may contain quite different facial expressions from the reference face and cannot be reliably matched directly.

Matching is done independently per clip, using the clips that were generated in Stage 3.2. Correspondence from the reference frame is first obtained for the anchor frames, as described in Section 3.3.1. These correspondences are then propagated from the anchor frames bounding the clip to the intermediate unanchored frames within the clip, both forwards and backwards, as described in Section 3.3.3.

#### 3.3.1 Tracking from Reference Frame to Anchor Frames

The orientation and expression of the face in anchor frames is similar to that in the reference frame, but its location within the frame might differ substantially. Thus the matching uses an image pyramid to detect large motions. The process starts at the coarsest level of the pyramid and matches a sparse set of features (we again use uniform image samples) to estimate extremal motions $\mathbf{m}^{\pm}$, followed by the dense motion estimation method described in Section 3.3.2 using a search window of size $[(\mathbf{m}_x^+ - \mathbf{m}_x^-) \times (\mathbf{m}_y^+ - \mathbf{m}_y^-)]$. The resulting motion field is upsampled to the next higher resolution and the dense motion estimation is repeated but now with a search window of fixed size ($3 \times 3$). This is repeated until the highest resolution layer is reached. This is done for each anchor frame to provide the motion fields $\mathbf{u}_c^{R \to A}$ from the reference frame $R$ to anchor frame $A$ for cameras $c = 1..n$.

#### 3.3.2 Dense Motion Estimation

The motion estimation is an extension to 2D of the matching introduced in [Beeler et al. 2010] and has the following steps:

**Matching.** A pixel $\mathbf{x}$ in image $I_c^R$ is matched to its best match $\mathbf{x}'$ in image $I_c^A$ using $3 \times 3$ block-based normalized cross-correlation with the search window introduced in 3.3.1. This provides the forward motion estimation $\mathbf{u} = \mathbf{x}' - \mathbf{x}$. The matching is run also in the reverse direction from $I_c^A$ to $I_c^R$ starting from $\mathbf{x}'$ to provide the backward motion estimation $\mathbf{v} = \bar{\mathbf{x}} - \mathbf{x}'$, where $\bar{\mathbf{x}}$ is the pixel in $I_c^R$ that backward matches $\mathbf{x}'$.

**Filtering.** A match is not accepted for pixels where $\|\mathbf{u} + \mathbf{v}\|$ is larger than a threshold (one pixel).

**Re-Matching.** Unmatched pixels are re-matched using accepted neighbor matches for guidance. This process is iterated until all unmatched pixels are matched.

**Refining.** The computed matches are refined by combining two terms, one for photometric consistency of the match in the two images, and one which uses a depth map to prevent smoothing over depth discontinuities. Depth maps are obtained from the meshes computed in Stage 1, reprojected back onto the images.

The original formulation of the refined motion in [Beeler et al. 2010] is a convex combination $\mathbf{u}' = (w_p \mathbf{u}_p + w_s \mathbf{u}_s)/(w_p + w_s)$. The weights $w_p$ and $w_s$ and the photometric term $\mathbf{u}_p$ are as in the original reference, but the regularization term $\mathbf{u}_s$ is modified based on the depth map $\delta$ and the matching error $\xi$

$$\mathbf{u}_s\left(x, y\right) = \sum_{(x',y') \in \mathcal{N}(x,y)} w_{x',y'} \cdot \mathbf{u}_{x',y'}$$

where $w_{x',y'} := \exp\left(\frac{\|\delta_{x',y'} - \delta_{x,y}\|^2}{\sigma^2}\right)(1 - \xi_{x',y'})$ and $\mathcal{N}$ denotes the neighborhood of $(x, y)$, i.e. the 8 neighboring pixels. The value of $\sigma$ is 1mm in our experiments.

#### 3.3.3 Tracking from Reference Frame to Unanchored Frames

Direct tracking of pixels from the reference frame to the unanchored frames is not reliable because image appearance can differ substantially between the two. Instead the matching between the reference frame and the anchor frames obtained in Section 3.3.1 is used to aid the process. Frames are tracked incrementally within a clip starting from the relevant anchor frames. The pixel tracking information from the reference frame to the anchor frame, plus the incremental frame-to-frame matching, is used to infer the pixel tracking from the reference frame to the individual unanchored frames.

$$\mathbf{u}_c^{R \to t} = \mathbf{u}_c^{R \to A} + \sum_{i<t} \mathbf{u}_c^{i \to i+1}$$

This generates a motion field for each unanchored frame, which is refined as described in Section 3.3.2. This last step allows the motion field to self-correct for small drift with respect to the reference frame (track-to-first principle).

Each clip is bound by a start and end anchor frame, and the pixel tracking above is done from the start anchor in the forward direction and from the end anchor in the backward direction. The forward and backward motion fields may differ due to error. This is resolved by computing an error field for each motion field, smoothing the error fields to remove local perturbation, and then taking the lowest smoothed error to obtain the best match at each pixel. Individual pixels may vary in their assignment of either forward or backward propagation, however the local error (and thus, assignment of propagation) tends to be temporally coherent. Any inconsistencies are resolved in the refinement process described in Stage 5.

### 3.4 Stage 4: Mesh Propagation

The reference mesh consists of a set of vertices $\mathbf{X}_i^R$ for the reference frame, obtained in Stage 1. Each vertex represents a physical point on the face. The goal of mesh propagation to a frame $F^t$ in the sequence is to find the transformed 3D position $\mathbf{X}_i^t$ of each vertex $\mathbf{X}_i^R$ due to the motion and deformation of the face from the reference frame to frame $F^t$.

Stage 3 has provided the motion fields from reference frame $F^R$ to $F^t$. The method for using the motion fields to estimate the propagated vertices $\mathbf{X}_j^t$ is similar to [Bradley et al. 2010]. Each vertex $\mathbf{X}_i^R$ is projected onto the camera images in $F^R$, and the corresponding motion vectors from the per-camera motion fields are applied. Back-projecting from the new pixel locations onto the initial geometry for frame $F^t$ (obtained in Stage 1) gives a per-camera estimate of the propagated 3D position. Estimates are weighted by the dot product between the surface normal and the camera view vector. Spatial clustering is used to identify outliers, and then the final estimate is obtained from a weighted average within the best cluster.

Mesh propagation is now complete and vertices $\mathbf{X}_i^R$ are in correspondence with vertices $\mathbf{X}_i^t$ in every frame $F^t$.

### 3.5 Stage 5: Mesh Refinement

Previous stages have generated a propagation of the reference mesh to each frame in the sequence. This is a step closer to the goal stated earlier, to have temporal correspondence of meshes along the sequence. However the propagated meshes have been computed with different methods (for anchor frames and unanchored frames) and computed independently for each timestep. The refinement described in this section updates the meshes to ensure a uniform treatment in the computation of all frames and to apply temporal coherence. There are two stages—a refinement that acts independently on each frame and can thus be parallelized, and a refinement that aims for temporal coherence between frames.

#### 3.5.1 Refinement per Frame

The goal of the per-frame refinement is to find for each vertex the position in space that optimizes the following objectives:

- Spatial image fidelity - The reprojections in all visible cameras for frame $F^t$ should be similar.
- Temporal image fidelity - The reprojections in frames $F^t$ and $F^R$ for each visible camera should be similar.
- Mesh fidelity - The transformed mesh $M^t$ should locally be similar to the reference mesh $M^R$.
- Geometry smoothness - The transformed geometry should be locally smooth.

To render the process robust and efficient we follow the proposition of Furukawa *et al.* [2009] and treat motion and shape separately. The refinement is an iterative process in 2.5D that interleaves motion and shape refinement.

**Shape Refinement.** Shape is refined along the normal. We employ the shape refinement framework from [Beeler et al. 2010], which aims to find vertex displacements $\mathbf{X}'$ that jointly satisfy photometric consistency constraints, surface smoothness constraints, and mesoscopic position estimates,

$$\mathbf{X}' = (w_p \mathbf{X}_p + w_s \mathbf{X}_s + w_\mu \mathbf{X}_\mu)/(w_p + w_s + w_\mu).$$

We refer to [Beeler et al. 2010] for the derivation of the photometric $\mathbf{X}_p$, smooth $\mathbf{X}_s$ and mesoscopic $\mathbf{X}_\mu$ positions as well as the photometric $w_p$ and mesoscopic $w_\mu$ weights. To produce smoother solutions in areas of higher matching error (eye-brows, nostrils, etc.) we modify the smoothness weight to be

$$w_s = \lambda_0^s + \lambda_1^s \xi + \lambda_2^s \xi^2$$

where $\xi$ is the matching error and $\lambda^s$ is a user defined smoothness coefficient vector. For all examples in this paper we use $\lambda^{\mathbf{s}} = [0.03, 0.7, 1000]$.

**Motion Refinement.** Motion is refined in the tangent plane of each vertex. Similar to the shape refinement process, we find vertex displacements $\mathbf{X}'$ that jointly satisfy photometric consistency and mesh regularization,

$$\mathbf{X}' = (w_p \mathbf{X}_p + w_s \mathbf{X}_s)/(w_p + w_s).$$

In this case, the photometric position estimate $\mathbf{X}_p$ is the position on the tangent plane that maximizes photometric consistency between current and reference frame. We use normalized cross-correlation as a measure of consistency and compute it by reprojecting corresponding surface patches into the reference image $I_c^R$ and the current image $I_c^t$ for all cameras $c$.

The regularized position estimate $\mathbf{X}_s$ for motion refinement assumes local rigidity of the surface and tries to preserve the local structure using Laplacian coordinates, as in [Bradley et al. 2010].

The *photometric confidence* $w_p$ is the sum of the matching errors for the neighboring positions on the tangent plane

$$w_p = 0.25(\xi_{x \pm dx, y} + \xi_{x, y \pm dy})$$

The *regularized confidence* $w_s$ employs a polynomial

$$w_s = \lambda_0^m + \lambda_1^m \xi_{x,y} + \lambda_2^m \xi_{x,y}^2$$

For all examples in this paper we use $\lambda^{\mathbf{m}} = [0.5, 1, 8000]$.

### 3.5.2 Refinement across Frames

The per-frame refinement in the previous section operates on each frame independently, and can thus be run in parallel. The results are not guaranteed to be temporally consistent but consecutive frames will be very similar by construction. While temporal error in the motion estimate is mostly imperceptible, small differences in shape between successive frames can cause changes in surface normals, which will produce subtle but noticeable flickering in the visualization. To avoid this we do a final pass over the complete sequence averaging the Laplacian coordinates in a [-1,+1] temporal window. This is implemented as an iterative process.

## 3.6 Acquisition Hardware

Our performance capture algorithm requires that the actor be illuminated by bright uniform light and recorded by $n \geq 2$ synchronized video cameras. While our technique does not depend on any specific hardware, we describe the acquisition setup used to generate the results in this paper for completeness.

All of our datasets were acquired using seven cameras (Dalsa Falcon 4M60) capturing images with a resolution of $1176 \times 864$ pixels at 42 frames per second and an exposure time of 8ms. Camera synchronization is essential for dynamic performance capture, and these cameras offer automatic hardware synchronization with a global shutter. Note that our method does not require professional high-speed video cameras (as used by Fyffe et al. [2011]), which would increase the cost of the system.

The actors were lit uniformly using an array of LED-lights. The lights are arranged in a spherical configuration for constant uniform illumination, akin to the USC light stage [Ma et al. 2007; Wilson et al. 2010; Fyffe et al. 2011]. However, unlike previous approaches we do not require polarized light or complex controllable light patterns. The simple LED-lights used by Bradley et al. [2010] would also be sufficient for our technique.

# 4 Results

We validate our method by reconstructing several performances given by three different actors. Our results cover a wide range of expressiveness, they include visually realistic pore-level geometry, and they demonstrate our method's robustness to motion blur and occlusions, outperforming previous approaches.

Figure 5 shows a number of frames from our first actor, including an input image (first row), resulting geometry (second row), the geometry rendered with a grid pattern to show the temporal motion over time (third row), and the final result rendered using per-frame color information projected from the video images. This result, while demonstrating the expressive quality of our reconstructions, also illustrates how we can match a reference frame across multiple capture sequences - the first three expressions in Figure 5 come from sequences that were captured on different occasions, however they are still in full vertex correspondence.

Reconstructed sequences from two additional actors are shown in Figure 6, demonstrating the versatility of our approach. Here we show several frames in chronological order, illustrating how the temporal reconstruction remains faithful to the true facial motion using an overlaid grid pattern. The result in the top row of Figure 6 particularly highlights the fine-scale spatio-temporal details that our method is able to produce.

One of the key innovations that our method relies on is our robust image-space tracking approach for deriving the temporal face motion. By dealing with error propagation directly in image space, we are able to produce more accurate motion reconstruction with less drift than techniques that rely on a potentially-distorted parameterization domain for drift correction, such as the approach of Bradley et al. [2010] (referred to just as 'Bradley et al' in the rest of this section). We illustrate this in Figure 7, where a short sequence of raising an actor's eyebrows to create wrinkles is reconstructed with both our approach and the method of Bradley et al. The initial geometry comes from Beeler et al. [2010] in both cases. Our method produces more detailed final geometry since we do not rely so heavily on spatial regularization, but also our method exhibits less drift accumulation. This can be seen in the latter three images, which show how the overlaid grid pattern deforms over time, from the first frame, to the most wrinkled frame, and then to a later frame after the wrinkles have dissipated. The temporal reconstruction of Bradley et al. is shown in yellow on the left half of the forehead, while our result is shown in blue on the right. A more regular grid pattern at the end of the sequence indicates that our approach is less susceptible to drift accumulation.

The second main contribution of our work is the application of anchor frames to address the problems associated with sequential motion processing, such as the unavoidable tracker drift over long sequences, and complete tracking failure caused by very fast motion or occlusions. With our anchor frame reconstruction framework we can recover from such tracking failure, as we demonstrate in Figure 8. This result shows a sequence of lip movements in which the upper lip is occluded by the lower lip in the third image. Since

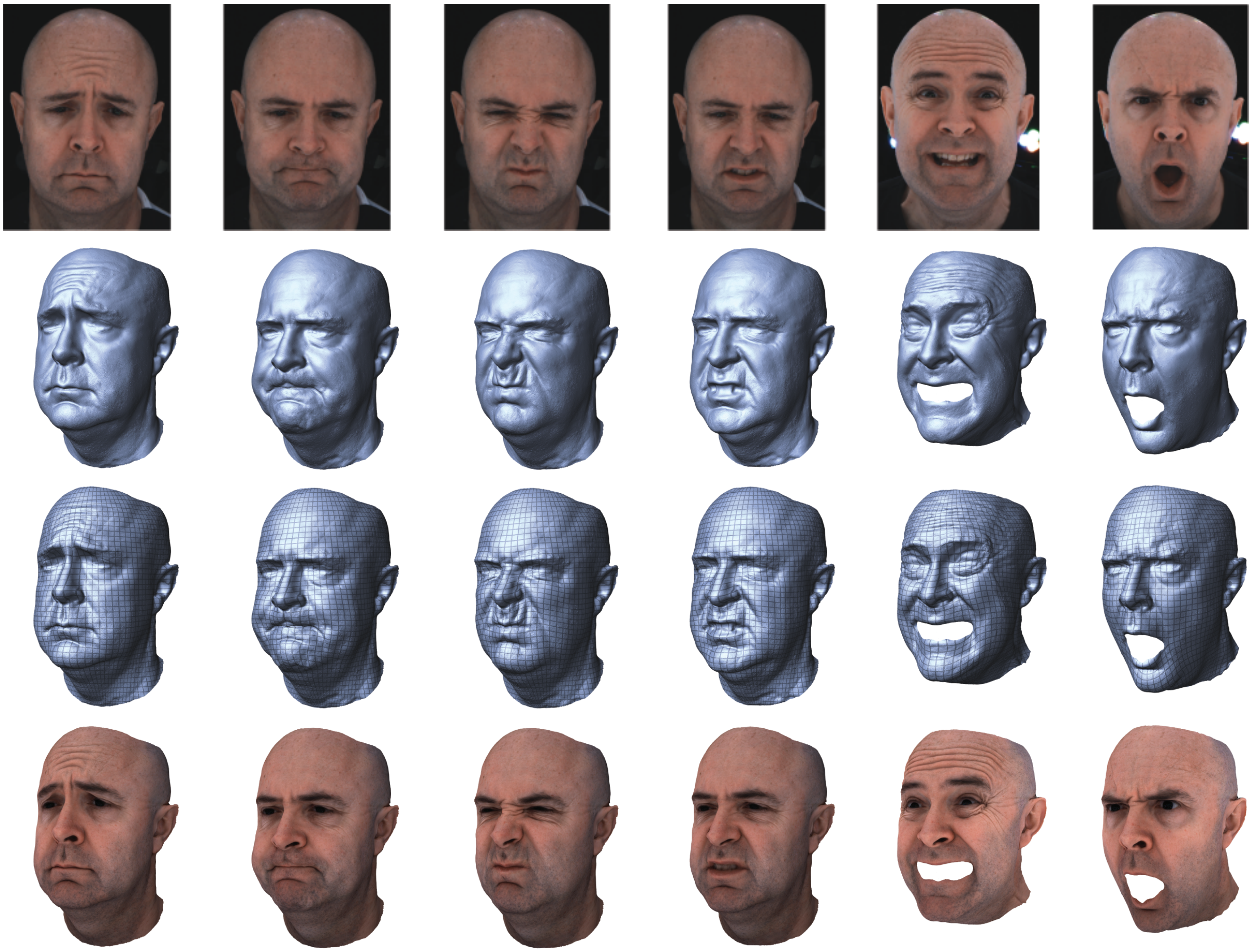

**Figure 5:** *Example frames taken from multiple different sequences of an actor. Top row: the images from one camera. Row 2: computed geometry. Row 3: geometry rendered with a grid pattern to enable a qualitative evaluation. Row 4: rendering of the computed mesh. In this figure, a single reference frame was the basis for processing multiple sequences of the actor, so the computed meshes are consistent (i.e. have corresponding mesh vertices) across all of the results.*

tracking of the upper lip fails, the system incorrectly predicts that the motion of the upper lip drifts down onto the lower lip, indicated by the overlaid grid. Sequential tracking methods would have trouble recovering from this situation. However, due to an anchor frame later in the sequence, our method is able to successfully track the upper lip backwards from the anchor frame to the occluded frame, automatically restoring tracking after the occlusion.

The combination of robust image space tracking and anchor frames allows us to successfully reconstruct very fast motions, even those containing motion blur. We demonstrate this in Figure 9, which contains a short sequence of an actor opening his mouth very quickly, and we compare our result again to the method of Bradley et al. Our method is able to produce a more accurate reconstruction.

**Analysis.** Here we assess the behavior of our algorithm under varying numbers of anchor frames. This assessment, depicted in Figure 10, demonstrates that our algorithm is relatively insensitive to the number of anchor frames, when present in typical amounts. For this analysis we ran our tracking method 9 times on the same sequence, using the same reference frame but varying the anchor frames. The sequence consists of 400 frames and contains a number of expressive as well as neutral poses, as illustrated by the images at the bottom of the figure. In 4 of the executions the anchor frames are manually selected (with 1, 3, 5 and 8 anchors), and in the other executions they are automatically selected based on the matching error (which is graphed in the background in purple); these 5 executions chose anchor frames that had the highest 10, 25, 50, 75 and 100 percentile matching error. At the low end, with only 1 anchor frame, we expect drift to accumulate since tracking is sequential (as in previous methods). At the high end, with 100% anchors, the method should degenerate completely because there is no frame-to-frame tracking at all. However, with a reasonable number of anchor frames we expect the results to be stable. We have no ground truth for measuring the difference between executions, so we chose the result with 8 anchors as the baseline for comparison. These anchor frames approximately partition the sequence into individual expressions bounded by neutral poses, which is exactly the situation where we expect our method to perform best. For each other result sequence, we measure the average image-space tracking error for each frame in pixels, compared to the baseline result, and accumulate the error across the sequence. This accumulated error is shown in the horizontal bar chart on the right side of the figure. As expected, using only 1 anchor frame produces significant error due to drift accumulation. When the number of anchor frames is very high the error is also large because many frames do not match

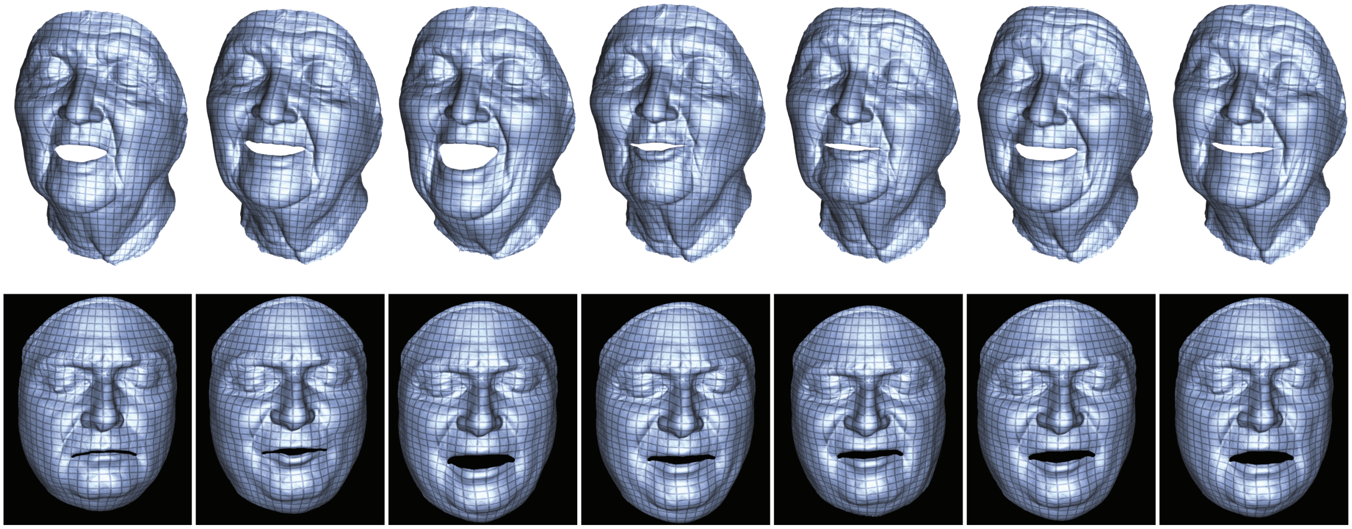

**Figure 6:** *Example on other subjects, showing frames pulled out at different time steps through a sequence. Low temporal drift is demonstrated by the consistent attachment of the superimposed grid pattern to the physical face surface.*

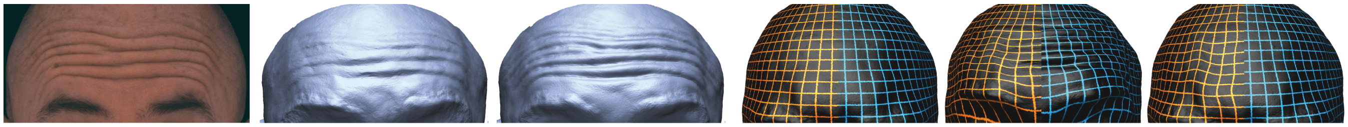

**Figure 7:** *A comparison with the method of Bradley et al. - From left to right: a zoom onto the forehead in an input image, 3D reconstruction of Bradley et al, our 3D reconstruction showing improved fidelity. The next three images show results for Bradley et al on the left-side of the head and our results on the right-side, for three time steps starting with unwrinkled forehead, then wrinkled forehead, then back to unwrinkled. Changes in the attachment of the superimposed grid to the face between the first and last images demonstrate drift in the tracking, with our method showing significantly less drift.*

well to the reference frame directly. However, using anywhere from 3 anchors to 10% of the frames as anchors produces similar results, indicating that our algorithm is relatively insensitive to the number of anchor frames.

Our initial presentation of the concept of anchoring involved a single reference frame. It may happen that a given reference frame does not yield a good distribution of anchor frames in the whole sequence, so that the benefits of anchor-based reconstruction are lost in some places. In this case, it is not necessary to maintain the same reference frame for the entire sequence. A subset of the sequence can be matched to one reference frame, followed by a change of the reference frame to one of the processed frames, if the switch would yield a better anchor frame distribution for the remainder of the sequence. The result shown in the latter three frames of Figure 5 is generated in this way, where the reference frame starts off as a neutral expression and then switches to a pose with the mouth open, since the mouth remains open for most of the sequence. The two reference frames would need to be in full vertex correspondence, but by switching the reference to a frame that has already been processed, the correspondence between the two frames is directly available.

Please see the supplemental material for video results of all datasets and comparisons in the paper.

## 5 Discussion

This paper presents a performance capture algorithm that can acquire expressive facial performances with visually-realistic pore-level geometric details. Here we discuss some of the research opportunities for extending the system.

In common with previous methods, we do not expect to obtain a faithful 3D reconstruction of the eye geometry or facial hair, since these tend to violate stereo and temporal brightness constancy assumptions in image space. Visual artifacts may also be seen at the mesh boundaries, however these can easily be cleaned in a post-process.

Our image-space tracking requires a stereo deployment where all cameras capture the full face. This contrasts with a stereo deployment like that of Bradley et al [2010], where the cameras are optically zoomed to capture small patches of the face, and a single point on the face often migrates between stereo views during a sequence[3]. Our approach could be extended to this situation if we were to combine the stereo images into one image, for example using the "unwrap mosaics" method of Rav-Acha et al. [2008].

The reference frame is matched to the anchor frames in image space, as described in Section 3.2. A more flexible approach, and a straightforward extension, would be to do the matching in 3D via the meshes that are computed in Stage 1 of the pipeline. Thus matching would succeed whenever facial expression is the same in the reference and anchor frames, and would no longer require a similar orientation of the head relative to the cameras.

Anchoring can be a powerful tool for integrating multiple face performances of an actor over an extended period. As shown in the results section, a reference frame can be taken from one sequence but used to generate anchor frames in another sequence. This provides a way to propagate a single mesh across different capture sessions

[3] For this reason we are not able to reconstruct the original datasets of Bradley et al. Instead we ran their tracking algorithm on our capture data for the comparison.

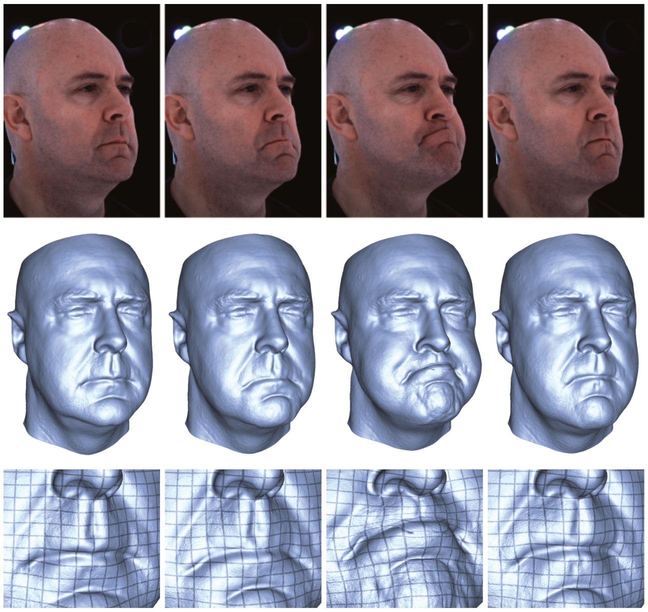

**Figure 8:** *Several frames from a sequence in which there is significant occlusion and reappearance of structure around the mouth. Our method of anchor frames combats this by partitioning the sequence into clips, and doing image-space tracking from the start and end of each clip.*

for an actor (including the case where the camera positions or calibration may have changed somewhat between the sessions), and to embed the full corpus of facial performance data for the actor into a single coordinate frame.

An extension of the work here would be to use multiple reference frames simultaneously. For example, facial performance capture could be applied to a sequence in which an actor adapts a set of FACS poses [Ekman and Friesen 1978] with careful supervision to yield best possible results. The frames with the FACS poses could then be used as a set of high quality reference frames that could be used simultaneously when processing subsequent sequences (because the meshes have consistent triangulation).

Finally we believe that there is an interesting new avenue for research in segmenting the face, and applying the concept of anchored reconstruction at the level of individual parts of the face. This relates to a contribution of our work, which was to show how a long image sequence can be decomposed into anchored clips for 3D reconstruction. Face segmentation will provide an orthogonal way of decomposing the process, and we plan to explore this extension.

## 6 Conclusion

We have presented a new passive technique for high-quality facial performance capture, based on two key technical innovations. First, we employ a robust tracking algorithm that integrates all of the pixel tracking in image space and uses the integrated result to propagate a single reference mesh to each target frame in parallel. Second, leveraging the fact that facial performances tend to contain repetitive motions, we introduce "anchor frames" defined as those where the facial expression is similar to the reference frame. After locating the anchor frames automatically, we compute pixel tracking directly from the reference frame to anchor frames. By using the anchor frames to partition the sequence into clips and independently matching clips, we are able to bound tracker drift, correctly handle occlusion and motion blur, and process capture sequences in parallel. We can even match frames between multiple capture sessions recorded on different occasions, yielding a single deformable mesh that corresponds to every performance an actor gives.

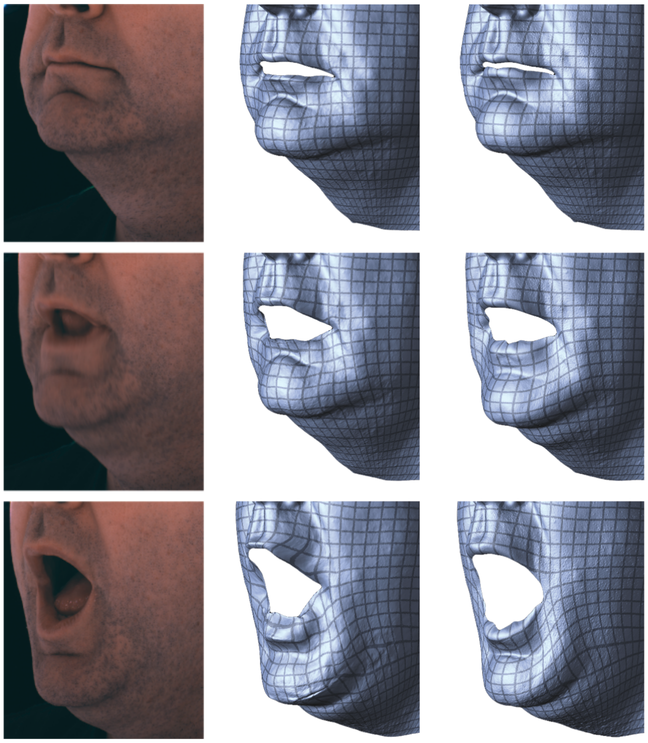

**Figure 9:** *Left: a zoom onto the mouth for several frames in a sequence in which there is fast facial motion and motion blur in the images. Center: the reconstruction of Bradley et al. Right: our reconstruction, with greater fidelity around the mouth.*

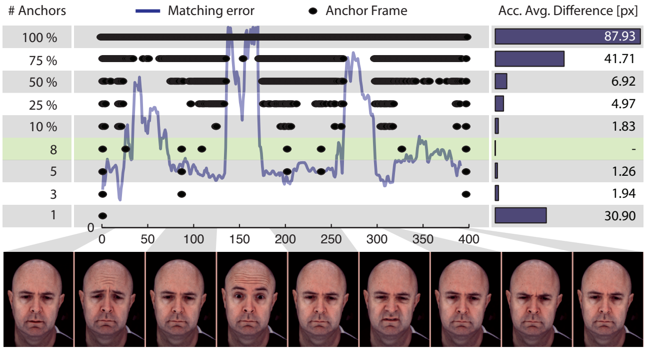

**Figure 10:** *Analysis of quantity and placement of anchor frames.*

Our method produces detailed 3D geometry in full temporal correspondence, even for the most expressive of performances undergoing very fast motion, without the requirement of hand-placed markers or face makeup. We have demonstrated our technique on a number of example performances given by different actors, and have also shown how our anchored-reconstruction approach combined with our robust image-space tracking method can yield more accurate results than a current state-of-the-art technique [Bradley et al. 2010], particularly in the presence of motion blur and highly expressive wrinkles where drift tends to accumulate faster.

To our knowledge, ours is the first method to passively reconstruct 3D facial performances with visually realistic pore-level geometric details, while demonstrating robustness to fast motions. A system of this type would be very useful for facial animation applications, such as performance transfer from one actor to another, in particular given the high-resolution geometry and expressive motions our method is able to reconstruct.

## Acknowledgements

We would like to thank our actors: Sean Sutton, Leila Gangji and Johann Gangji. Many thanks also to Maurizio Nitti and Gioacchino Noris for the artistic help with our results and figures.

## References

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

ANUAR, N., AND GUSKOV, I. 2004. Extracting animated meshes with adaptive motion estimation. In *Proc. Vision, Modeling, and Visualization*, 63–71.

BEELER, T., BICKEL, B., SUMNER, R., BEARDSLEY, P., AND GROSS, M. 2010. High-quality single-shot capture of facial geometry. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 40.

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. Graphics (Proc. SIGGRAPH)*, 33.

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

BRADLEY, D., POPA, T., SHEFFER, A., HEIDRICH, W., AND BOUBEKEUR, T. 2008. Markerless garment capture. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 99.

BRADLEY, D., HEIDRICH, W., POPA, T., AND SHEFFER, A. 2010. High resolution passive facial performance capture. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 41.

DECARLO, D., AND METAXAS, D. 1996. The integration of optical flow and deformable models with applications to human face shape and motion estimation. In *Proc. CVPR*, 231–238.

EKMAN, P., AND FRIESEN, W. 1978. The facial action coding system: A technique for the measurement of facial movement. In *Consulting Psychologists*.

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

FURUKAWA, Y., AND PONCE, J. 2009. Dense 3D motion capture for human faces. In *Proc. CVPR*, 1674–1681.

FYFFE, G., HAWKINS, T., WATTS, C., MA, W.-C., AND DEBEVEC, P. 2011. Comprehensive facial performance capture. *Comp. Graphics Forum (Proc. Eurographics) 30*, 2, 425–434.

GUENTER, B., GRIMM, C., WOOD, D., MALVAR, H., AND PIGHIN, F. 1998. Making faces. In *Comp. Graphics*, 55–66.

HERNÁNDEZ, C., AND VOGIATZIS, G. 2010. Self-calibrating a real-time monocular 3D facial capture system. In *Proceedings International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*.

KRAEVOY, V., AND SHEFFER, A. 2004. Cross-parameterization and compatible remeshing of 3D models. *ACM Trans. Graph. 23*, 861–869.

LI, H., ROIVAINEN, P., AND FORCHEIMER, R. 1993. 3-D motion estimation in model-based facial image coding. *IEEE Trans. Pattern Anal. Mach. Intell. 15*, 6, 545–555.

LIN, I.-C., AND OUHYOUNG, M. 2005. Mirror mocap: Automatic and efficient capture of dense 3D facial motion parameters from video. *The Visual Computer 21*, 6, 355–372.

LOWE, D. G. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision 60*, 2, 91–110.

MA, W.-C., HAWKINS, T., PEERS, P., CHABERT, C.-F., WEISS, M., AND DEBEVEC, P. 2007. Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination. In *Eurographics Symposium on Rendering*, 183–194.

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. Graphics (Proc. SIGGRAPH Asia) 27*, 5, 121.

PIGHIN, F. H., SZELISKI, R., AND SALESIN, D. 1999. Resynthesizing facial animation through 3D model-based tracking. In *Proc. ICCV*, 143–150.

POPA, T., SOUTH-DICKINSON, I., BRADLEY, D., SHEFFER, A., AND HEIDRICH, W. 2010. Globally consistent space-time reconstruction. *Comp. Graphics Forum (Proc. SGP)*, 1633–1642.

RAV-ACHA, A., KOHLI, P., ROTHER, C., AND FITZGIBBON, A. 2008. Unwrap mosaics: A new representation for video editing. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 17.

SHARF, A., ALCANTARA, D. A., LEWINER, T., GREIF, C., SHEFFER, A., AMENTA, N., AND COHEN-OR, D. 2008. Space-time surface reconstruction using incompressible flow. *ACM Trans. Graphics 27*, 110.

SUMNER, R. W., AND POPOVIĆ, J. 2004. Deformation transfer for triangle meshes. *ACM Trans. Graphics 23*, 399–405.

WAND, M., ADAMS, B., OVSJANIKOV, M., BERNER, A., BOKELOH, M., JENKE, P., GUIBAS, L., SEIDEL, H.-P., AND SCHILLING, A. 2009. Efficient reconstruction of nonrigid shape and motion from real-time 3D scanner data. *ACM Trans. Graph. 28*, 2, 1–15.

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

WILLIAMS, L. 1990. Performance-driven facial animation. In *Computer Graphics (Proc. SIGGRAPH)*, vol. 24, 235–242.

WILSON, C. A., GHOSH, A., PEERS, P., CHIANG, J.-Y., BUSCH, J., AND DEBEVEC, P. 2010. Temporal upsampling of performance geometry using photometric alignment. *ACM Trans. Graphics 29*, 2.

WINKLER, T., HORMANN, K., AND GOTSMAN, C. 2008. Mesh massage. *The Visual Computer 24*, 775–785.

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