

# Facial Cartography: Interactive Scan Correspondence

Cyrus A. Wilson<sup>1</sup> Oleg Alexander<sup>1</sup> Borom Tunwattanapong<sup>1</sup> Pieter Peers<sup>1,2</sup>  
Abhijeet Ghosh<sup>1</sup> Jay Busch<sup>1</sup> Arno Hartholt<sup>1</sup> Paul Debevec<sup>1</sup>

<sup>1</sup>USC Institute for Creative Technologies

<sup>2</sup>The College of William & Mary

---

## Abstract

*We present a semi-automatic technique for computing surface correspondences between 3D facial scans in different expressions, such that scan data can be mapped into a common domain for facial animation. The technique can accurately correspond high-resolution scans of widely differing expressions – without requiring intermediate pose sequences – such that they can be used, together with reflectance maps, to create high-quality blendshape-based facial animation. We optimize correspondences through a combination of Image, Shape, and Internal forces, as well as Directable forces to allow a user to interactively guide and refine the solution. Key to our method is a novel representation, called an Active Visage, that balances the advantages of both deformable templates and correspondence computation in a 2D canonical domain. We show that our semi-automatic technique achieves more robust results than automated correspondence alone, and is more precise than is practical with unaided manual input.*

---

## 1. Introduction

Just as portraiture is one of the most challenging but important aspects of painting, rendering human faces is one of the most challenging but important aspects of computer graphics. Progress in this area has accelerated greatly, with recent results in research, movies, and video games building the first bridges across the *Uncanny Valley* towards believably realistic digitally rendered faces. However, creating a photorealistic digital actor remains complicated and time-consuming [ARL\*10], which prevents their widespread use.

Recent 3D scanning techniques provide useful data for creating digital faces based on real people, able to be rendered with any viewpoint and lighting. Since the appearance and deformation of emotive faces is complex, many digital characters are built from numerous scans of an actor making different expressions called *blendshapes*, interpolating between them to create facial animation. Blending between expressions, however, requires knowing how the surface points in each scan correspond to each other. Determining such correspondences can be made easier by placing markers on the actor's face, but this is time consuming and mars the appearance of the actor; moreover, even hundreds of markers yield little information about dynamic skin wrinkle behavior. A dynamic facial scanning system can also make the task easier, since the transition from one expression to another is recorded as a sequence of scans with little motion between them. However, dynamic scanning systems are typically very data intensive and provide far lower resolution geometry and reflectance than systems which record static facial expressions. As a result, building the highest-quality digital characters requires determining accurate correspon-

dences between scans of significantly differing expressions without the aid of facial markers. And typically, this is one of the most difficult stages of the character creation process.

An important step in creating an animatable character is to create a facial animation rig, often built on top of an animation mesh of moderate resolution augmented by high resolution detail textures (e.g., albedo, normals, etc.). Computing correspondences on the high resolution scan meshes, and subsequently downsampling to the animation mesh produces suboptimal results. To obtain the highest quality blendshapes, *both* the animation mesh and the detail textures need to be optimally corresponded. Downsampling a high resolution mesh does not guarantee the latter because important visual details in the textures do not necessarily align with vertices.

While exact 1 : 1 correspondences exist between the physical expressions, they might be difficult to uniquely identify between facial scans (e.g., appearance and disappearance of wrinkles). In such a case it is often difficult to find consistent 1 : 1 correspondences. Automated methods often rely on heuristics (e.g., smoothness) to handle such ambiguous cases. A more robust solution is to add case-specific user-defined constraints to these automatic methods. However, such an approach often results in a trial-and-error procedure, where the focus lies on avoiding undesired behavior of the automatic algorithms by tweaking and adding non-intuitive constraints. Instead, it would be better to have the user *participate* in the correspondence estimation process in a constructive manner by *directing* the computations rather than constraining and/or correcting them.

We propose a novel correspondence method, that we

call *Facial Cartography*, for computing correspondences between high resolution scans of an actor making different expressions, to produce mappings between a user-specified neutral mesh and all other expressions. A key difference between the proposed method and prior work is that we allow the user to participate and provide direction *during* the correspondence computations. Additionally, we compute correspondences on the final animation mesh and detail textures. To ensure optimal detail texture alignment and interactive user interaction, we leverage the GPU, and adopt an analysis-by-synthesis approach. A key component in our system is an *Active Visage*: a proxy for representing corresponded blendshapes that can be visualized from one or more viewpoints to evaluate the difference (i.e., error) with ground-truth visualizations of the surface properties of the target expression. While conceptually similar to a deformable template, a key difference is that a deformable template deforms a 3D mesh, while an active visage is constrained to the manifold of the non-neutral expression geometry, and hence only deforms the correspondences and not the 3D shape.

We have developed a modular optimization framework in which four different forces act on these active visages to compute accurate correspondences. These four forces are:

1. **Image Forces:** favoring correspondences providing the best alignment of fine-scale features in the detail maps;
2. **Shape Forces:** providing soft constraints on the 3D vertex positions of the optimization estimate;
3. **Internal Forces:** avoiding implausible deformations by promoting an as-rigid-as-possible deformation; and
4. **Directable Forces** that in conjunction with a GPU implementation of the other forces, enable user-participation in the optimization process, and direct the optimization.

## 2. Related Work

Computing correspondences between two surfaces is a classical problem in computer graphics and computer vision, and a large body of prior work investigates variants of this challenging problem. The ability to establish accurate surface correspondences enables a variety of applications such as the animation and morphing of shapes, shape recognition, and compression. Describing all prior work in correspondence computation is beyond the scope of this paper; we thus focus on the most relevant work to compute correspondences between a scan of a neutral reference pose and scans of a small set of extreme poses of a subject.

**3D Methods.** Allen et al. [ACP02] construct a kinematic skeleton model for articulated body deformations from range scan data and markers. Chang and Zwicker [CZ09] propose a markerless registration method that employs a reduced deformable model, and they decouple deformation from surface representation by formulating weight functions on a regular grid. Both methods are geared towards articulated

objects, and are not well suited for modeling facial animations.

Brown and Rusinkiewicz [BR04, BR07] and Amberg et al. [ARV07] develop a non-rigid variant of the well known Iterative Closest Points (ICP) algorithm [BM92] to align rigid objects where calibration errors introduce small non-rigid deformations in the scans. A disadvantage of these approaches is that they can easily converge to a suboptimal solution due to their greedy nature.

Li et al. [LSP08] form correspondences by registering an embedded deformation graph [SP04] on the source surface to match the target surface. Key to their method is an optimization that robustly handles missing data, favors natural deformation, and maximizes rigidity and consistency. However, this method is limited to the resolution of the deformation graph, and cannot handle very small non-rigid deformations. [LAGP09] alleviates this for temporally dense input sequences by computing a displacement projection to the nearest surface point. However, this projection does not rely on surface features, and thus can introduce small errors.

**2D Methods.** Another strategy for establishing correspondences between two non-rigidly deformed shapes is to embed them in a 2D canonical domain, reducing the correspondence problem to a simpler 2D image matching problem. Anguelov et al. [AKS\*05] introduce *correlated correspondences* to compute an embedding of each shape that preserves geodesics and thus minimizes differences due to deformations. Anguelov et al. employ this algorithm to create SCAPE [ASK\*05], a data-driven model that spans both deformation as well as shape of a human body. Wang et al. [WWJ\*07] investigate several different types of quasi-conformal mappings with regard to 3D surface matching, and conclude that least squares conformal mapping is the best choice for this purpose. Zeng et al. [ZZW\*08] observe that most conformal mapping methods are not suited to deal with captured data due to inconsistent boundaries, complex topology, and distortions. To address these issues, they compute correspondences from multiple boundary-constraint conformal mappings. Recently, Lipman and Funkhouser [LF09] proposed Möbius voting, an algorithm for finding point correspondences between two near-isometric surfaces in polynomial-time. It selects the Möbius transform that minimizes deformation error of the two surfaces in a canonical domain. Besides the sensitivity to scanning noise and inconsistent boundaries, these embedding based strategies also assume that the deformed meshes are isometric, which is not truly the case for most surface deformations. Consequently, the resulting correspondences may be affected. To handle non-isometric surfaces robustly, Zeng et al. [ZWW\*10] apply a higher-order graph matching scheme on the combined embedding space determined by the Möbius transform and the Gaussian map of the surface. While mathematically elegant, the exact mapping to these embedded spaces are non-intuitive, making it difficult

for users to manipulate and correct the computed correspondences.

Litke et al. [LDRS05] propose a variational method that matches surfaces in a 2D canonical domain. Similar to the proposed technique, their matching energy takes into account various cues such as curvature and texture. Furthermore, it allows user-control in the form of feature lines (as opposed to point-wise constraints). While their system can produce impressive results, it is limited to surfaces that are homogeneous to a disk. Eyes and mouth need to be manually segmented out.

Blanz and Vetter [BV99] create a morphable face model from a database of static laser scans of several individuals in different expressions. To correspond the different expressions, a combination of optical flow and smoothing is employed, exploiting the native cylindrical output parameterization of the laser scanner. In subsequent work, a similar approach was used to create a morphable mouth model [BBVP03]. Huang et al. [HCTW11] employ a similar method for aligning small scale details, while large scale deformations are registered using a marker-based approach. A general problem with optical flow based methods is that they succeed in some cases but fail entirely on seemingly similar cases. User input is often employed to constrain the optical flow algorithm to produce a suitable solution. However, it is not always intuitive how a particular optical flow optimization trajectory will respond to specific constraints, leading to a trial-and-error procedure.

**Analysis-by-synthesis.** A key component in our system is the GPU-accelerated real-time visualization and feedback system. However, we are not the first to include graphics hardware in a registration pipeline. Pighin et al. [PSS99] and Pons et al. [PKF07] iteratively refine the correspondences by evaluating the error on the deformed target surface and the predicted deformed source surface. Blanz and Vetter [BV99] also employ an *analysis-by-synthesis* loop to match a morphable face model to one or more photographs of a subject. However, none of these methods provide a way for the user to correct erroneous correspondence estimates.

**User-interaction.** Finally, all prior methods have aimed at making the registration process as automated as possible. User interaction has only been employed to either initialize the computations [DM96, ACP02, LDRS05, ZZW\*08] or to constrain the solution space to avoid local minima [BBVP03, HCTW11]. However, none of the previous methods actually allows the user to *direct* the correspondence computations as part of the main optimization loop.

### 3. Algorithm

**Input.** Our correspondence algorithm takes as input a set of scanned meshes and high resolution detail maps of a subject, each exhibiting a different expression. One of these expressions, the most neutral one, is selected as a reference pose,

and an animation mesh is created for this expression, which can serve as the basis for an animation rig. This animation mesh can either be created by an artist based on the scanned mesh or can be a direct copy of the acquired neutral mesh. Additionally, the animation mesh is augmented by high resolution detail textures that can contain diffuse and specular albedo information, surface normal information (to compensate for the differences in fine geometric details between the animation mesh and the scanned mesh), etc.. This animation mesh is subsequently deformed to *roughly* match the target mesh. This can either be done by an artist, or by using any suitable automatic method discussed in Section 2. This deformation does not need to be exact; it is only used to bootstrap the optimization.

**Goal.** The goal of Facial Cartography is to create dense mappings between the user selected neutral mesh and all other expressions. A distinctive feature of Facial Cartography is that it was developed with the following two points in mind. First, to provide an intuitive and effective user experience, Facial Cartography allows the user to interactively participate during the computations of the correspondences. This allows the user to direct the computations away from local minima and even bias the computations to a subjectively preferred solution. Second, the need to correspond scans of different expressions is to aid in the creation of animation rigs. Therefore, to obtain high quality animation rigs, computed correspondences need to be optimal with respect to the actual animation mesh, the detail textures and their corresponding UV mappings.

**Optimization.** We frame the computation of correspondences between two surfaces as a energy-minimization optimization in which the user can participate via a live, interactive simulation while the optimization is in progress. To achieve this we consider the objective function as an energy potential function  $U$ , indicating how well the two surfaces are in correspondence, associated with a conservative force  $\mathbf{f}$ , where  $\mathbf{f} = -\nabla U$ . This objective function can be minimized using a gradient-descent optimization, displaced by  $\mathbf{f}$  at every iteration. We identify the following four forces that play a role in the correspondence optimization:

- **Image Forces:** ensure proper registration of the features in the high resolution detail maps. Image forces are computed via an *analysis-by-synthesis* approach (Section 5).
- **Shape Forces:** allow the user to provide soft constraints on the 3D vertex positions of the animation mesh. (Section 6).
- **Internal Forces:** constrain the correspondence solution such that undesirable or impossible deformations (e.g., collapsing of triangles) are avoided by enforcing an as-rigid-as possible deformation (Section 7).
- **Directable Forces:** allow the user to direct the optimization (Section 8).

The key distinctive feature of Facial Cartography is not the general optimization above, but the combination of the

forces and the domain on which we apply the optimization. We define the optimization domain directly onto the target manifold (i.e., the non-neutral expression mesh). To facilitate the correspondence computations over this domain, taking into account the animation mesh and detail textures, and supporting the analysis-by-synthesis approach for the image forces, we create a proxy called the *Active Visage* that represents one of the possible deformations of the neutral expression constrained to the target surface (Section 4).

At every iteration in the optimization, we remap (and re-sample if necessary) the four forces to the vertices in the 2D optimization domain that comprise the active visage. The final resulting force acting on each vertex is then a weighted sum of each of the remapped forces. The user can enable and disable different forces during the optimization, and modulate their weights: typically we use an image weight between 4 and 8, shape weight = 1, internal weight = 1, and directable weight = 1.

#### 4. Active Visage

As noted in the previous section (Section 3), we define the optimization domain directly onto the target manifold, and optimize correspondences only, not shape. Intuitively, one can visualize this as gliding and stretching a rubber sheet over the target surface (i.e., maintaining contact with the target surface) until specific constraints are fulfilled (specified by forces in our optimization framework).

While it makes sense to define the optimization domain directly on the target manifold, it is not always the most straightforward parameterization to compute the forces that drive the optimization. To facilitate the force computations, we create a proxy called the *Active Visage*, where every point in the optimization domain has an associated 3D position (constrained to the target manifold) and corresponding surface properties borrowed from the neutral expression. An active visage represents one of the possible deformations of the neutral expression onto the target surface. As such, an active visage supports all operations that otherwise would have been performed on the neutral and/or target expression (e.g., visualization).

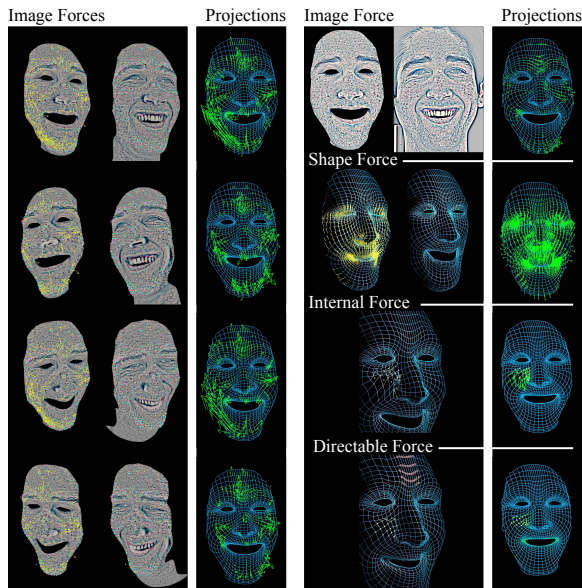
**Implementation.** Practically, we implement an active visage as follows. An active visage is a mesh that has the same connectivity as the animation mesh. Every vertex has a texture coordinate, which stays constant during optimization, that maps to the high resolution detail textures (obtained from the scanned neutral expression). Furthermore, every vertex has a correspondence estimate. This estimate is defined as a texture coordinate in the target expression mesh’s native texture parameterization or any other suitable 2D mapping (e.g., conformal mappings, or in case of 2.5D scans, this parameterization can be in the captured image space). The exact form of the 2D map is less important as long as we can map between surface coordinates and texture coordinates. For every correspondence estimate, we also

store its 3D coordinate on the target manifold and update it at every iteration of the optimization using a reverse lookup. The active visage can be easily rendered, mapped with any channel of information from the neutral expression and from any viewpoint.

**Discussion.** Conceptually, the active visage representation falls in between computing correspondences using a deformable template mesh and computing correspondences in an intermediate 2D domain.

The key difference between a deformable template and an active visage is that a deformable template deforms a 3D mesh, while an active visage is constrained to the manifold of the non-neutral expression geometry, and hence only deforms the correspondences. Solving the correspondence problem by finding a suitable deformation of an input or template mesh such that it matches a given target mesh subject to method-specific constraints (e.g., as-rigid-as-possible deformation, map constraints, etc.) spends significant resources in optimizing the shape (a 3D problem), and not the correspondences (a 2D problem).

At a high level, computing correspondences using an active visage is similar to computing correspondences in an intermediate 2D domain (e.g., conformal mapping): in both cases the optimization domain is 2D. However, a distinct difference is that an active visage represents a 2D manifold in 3D space defined by the final animation mesh. Often, the final animation mesh is an artist-tuned mesh where edges are aligned with specific features (e.g., ring-like structure around eyes and mouth), hence alignment of these features is important. Many intermediate 2D parameterizations are defined by optimizing vertex positions in a 2D domain to satisfy some preset constraint. Consequently, straight edges in 3D on the target manifold do not necessarily correspond to straight edges in the 2D parameterization, significantly complicating the alignment of edge features especially when the mesh resolution of the animation mesh and the scan differ. A similar problem occurs for the high resolution detail maps. Such detail textures are commonly employed to *compensate* for the lack of detail in the artist designed animation mesh. Since these detail maps greatly influence the appearance (e.g., normal maps) of the visualizations of the animated mesh, optimally corresponding the features in these maps is of utmost importance. Active visages provide a natural means of dealing with these issues. Although the optimization domain is 2D, edge and texture projections do not introduce discrepancies because the active visage is defined on a 3D manifold that corresponds to target manifold. Moreover, the active visage, in conjunction with the analysis-by-synthesis approach, allows Facial Cartography to take in account for the correspondence computations any “errors” in shape due to the lower resolution of the animation meshes. By employing an analysis-by-synthesis approach, we ensure that the features in the detail textures are visually registered consistently for the whole animation rig.



**Figure 1:** Forces acting on an active visage, shown in their native formulations, and after remapping (“Projections”). Arrows have been scaled up for visibility.

While a template or an intermediate 2D projection could arguably be extended to correctly handle all these conditions, we believe that the “active visage” makes this more convenient.

## 5. Image Forces

Image forces provide a mechanism to incorporate non-geometric cues, such as for example, texture information which is often available at a much finer granularity than the mesh resolution. Resampling this texture information on the mesh can result in a loss of important high frequency information. Instead, it is better to employ an analysis-by-synthesis approach, where image forces are computed on 2D visualizations of this high resolution data.

**Input.** The computation of an image force takes as input two 2D visualizations: a visualization of the active visage (i.e., the animation mesh deformed according to the current correspondence estimate) and a visualization of the target mesh. Both visualizations are from the same viewpoint, and display the same set of surface properties.

**Force Computation.** The main goal of the image force is to express how the active visage should be warped such that its visualization better matches the corresponding visualization of the target expression. There are many possible methods for computing such image forces (e.g., optical flow). However, to maintain interactive rates suitable for user participation, we harness the computing power of the GPU, and opt for a local cross-correlation window-matching approach.

A typical local cross-correlation window-matching ap-

proach works as follows. First, a number of feature locations are selected in the active visage visualization  $A$ . Then, a mean-subtracted discrete cross-correlation is computed between a given window centered around each feature location in  $A$  and in the target visualization  $B$ :

$$(A_s \star B_s)(p, q) \equiv \sum_{\tau=-\eta}^{\eta} \sum_{\nu=-\eta}^{\eta} [(A(s_u + \tau, s_v + \nu) - \langle A_s \rangle) \cdot (B(s_u + \tau + p, s_v + \nu + q) - \langle A_s \rangle)], \quad (1)$$

where  $\mathbf{s} = [s_u, s_v]^T$  is the window center (i.e., feature point location),  $\langle A_s \rangle$  is the mean value of  $A$  within the window, and the window size is  $(2\eta + 1) \times (2\eta + 1)$ . We found that a  $\eta$  of either 7 or 15 gave the best quality versus computational cost ratio. If the cross-correlation  $A_s \star B_s$  yields a significant peak in the window, then we compute the centroid  $[p, q]^T$  of the peak, which indicates the displacement of the feature between the visualizations.

We can optimize this computation by observing that both the texture coordinates and the content of the detail maps of the active visage do not change during the optimization. Hence, we can precompute the feature locations in texture space, and at each iteration project the precomputed feature locations into the visualization viewpoint, which can be done very efficiently.

**Surface Properties and Feature Selection.** The exact choice of surface properties should provide a maximum number of distinctive cues, and thus good image forces, to drive the correspondence optimization. In our implementation we employ a high-pass filtered version of the skin texture, and use Shi and Tomasi’s method to detect good features [ST94]. This ensures that significant changes in skin-texture are well aligned. These major changes in skin-texture are well located in space (i.e., a high-frequency peak) and correspond to the same surface location. However, other surface properties, such as Gauss curvature, can also provide additional cues in conjunction with the high-pass filtered skin-texture.

**Visualization Viewpoints.** While the vertices of the animation mesh are constrained to the target mesh, both meshes do not necessarily form the same manifold in 3D space, due to the differences in resolution between the animation mesh and the target mesh. In order to minimize visual artifacts because of differences in shape, we employ an analysis-by-synthesis (i.e., inverse rendering) approach for computing the image forces. In particular, for good coverage of the face, we compute image forces for five different viewpoints: frontal, upper left, upper right, lower left, and lower right.

**Aggregate Image Force.** Finally, we need to resample, remap and combine all computed image forces to the optimization domain (i.e., the target manifold):

- **Resample:** the locations of the sparse features at which the image forces are computed are most likely not the same as the locations of the mesh vertices (projected into

the virtual viewpoint). We therefore first need to resample the computed image forces to each vertex location. For this we employ a Gaussian weighted radial basis function based on the distance between the sparse feature location (on the mesh) and the vertex location.

- **Remapping:** Next we need to remap each of the computed (resampled) forces from the visualization domain to the optimization domain. Care has to be taken to correctly account for the change in differential measure by multiplying the resampled image forces by the Jacobian of this transform (i.e., from image pixels to a local tangential field around the target vertex).
- **Combine:** Finally, we add all image forces from the different virtual viewpoints into a single net image force.

Figure 1 (left) shows the visualizations of an active visage and corresponding target expression for the five different viewpoints, the image forces in the visualization domain and the corresponding remappings onto the animation mesh in the optimization domain.

## 6. Shape Forces

Shape forces allow the user to provide soft constraints on the 3D vertex positions of the active visage. This is an ancillary soft constraint as opposed to the primary hard constraint that the vertices of the active visage lie on the target manifold.

**Input.** The computation of a shape force takes as input a deformed 3D mesh that has the same connectivity as the animation mesh (and thus the active visage). While the 3D vertices of the active visage are constrained to the target manifold, we do not require the same from the input deformed mesh, because it is often more convenient for the user to specify constraints on the vertex positions in the form of a deformed animation mesh or by deforming the 3D shape of the (current) active visage.

**Force Computation.** Because we require the input deformed mesh to have the same connectivity, computation of this force is trivial. The shape force on a vertex of the active visage is proportional to the distance to the corresponding vertex in the input deformed mesh. The resulting 3D force is subsequently remapped onto the 2D optimization domain. For this we need to multiply the resulting 3D shape force vector with the Jacobian describing the difference in differential measure from 3D to the local tangent plane around the vertex on which the shape force acts. Figure 1, 2<sup>nd</sup> row right, shows the 3D shape force and its remapping.

**Application.** In our implementation, we allow the user to exert shape forces in two instances:

1. To bootstrap the method, a deformed version of the animation mesh, roughly matching the target shape, serves as input to the shape force. Depending on the accuracy of this initial guess, the user may opt to reduce the influence of this initial guess by gradually lowering the weight of the resulting shape force as the optimization progresses.

2. We furthermore allow the user to select an intermediate solution at any iteration as a soft constraint. This is useful when the user decides that the current solution provides a good base configuration; it can also serve as a tool to avoid getting stuck in a local optimum.

## 7. Internal Force

The above shape and image forces encourage similarity between corresponded points. However, by doing so, spatial ordering relationships are ignored. For example, the spatial relationship between two surface points (e.g., point A is above point B), is unlikely to reverse on a physical subject between different expressions, but swapping their position might (depending on the input data) numerically minimize the shape and image force. To avoid such physically implausible deformations, we introduce an *Internal Force* that promotes an as-rigid-as-possible deformation.

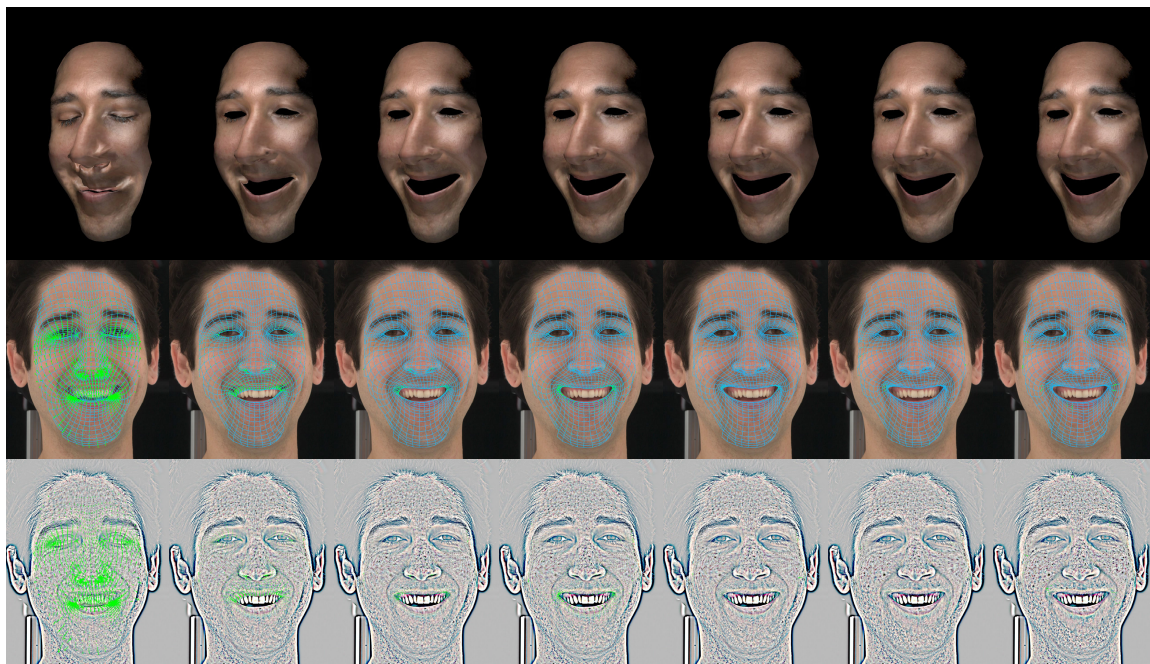
**Input.** The internal force only depends on the geometric ordering of the vertices in the animation mesh corresponding to the neutral expression (i.e., “resting state”), and on the current ordering in the active visage.

**Force Computation.** We define a strain stress imposed by the deformation of the active visage relative to the “resting state”. We treat edges of the active visage as springs, with equilibrium length defined as the length of the edges in their “resting state”. Displacements of mesh vertices representing non-rigid deformations will change the edge lengths, and result in a restoring force:

$$\rho_j = \begin{cases} -\frac{\epsilon_j}{\|\epsilon_j\|} \kappa (\|\epsilon_j\| - \lambda_j) & \text{if } \|\epsilon_j\| > \lambda_j, \\ 0 & \text{if } \|\epsilon_j\| \leq \lambda_j. \end{cases} \quad (2)$$

Here,  $\lambda_j$  is the equilibrium length of the edges, and  $\epsilon_j$  is the edge vector, and  $\kappa$  is a spring constant. The net resulting 3D internal force per vertex is then the sum of all restoring forces acting on that vertex. This 3D internal force can be easily and efficiently computed on the GPU. Finally, similar to the shape force, this 3D force vector needs to be remapped to the 2D optimization domain (Section 6).

**Orthogonal Forces.** In practice there are instances in which the computed 3D internal force is nearly orthogonal to the local tangent plane around the vertex on the target manifold on which the internal force acts, resulting in a negligible internal force (when mapped to the optimization domain) despite a potentially large deformation strain stress. We therefore also compute *2D internal forces* directly on the 2D correspondence estimate (i.e., a texture coordinate in the target expression mesh’s native texture parameterization (Section 4-Implementation)). This 2D internal force is remapped to the optimization domain and added to (remapped) 3D internal force. Figure 1, 3<sup>rd</sup> row right, shows the 3D internal force and its 2D mapping.



**Figure 2:** Intermediate states of the Active Visage as the simulation progresses, shown at an interval of 70 iteration steps. Top: 3D visualization of the active visage; Middle: active visage and corresponding forces overlaid on frontal camera view of the target subject; Bottom: band-pass filter texture overlaid with the aggregate forces (scaled up for visibility).

## 8. Directable Force

The final forces that drive the correspondences optimization are the user directable forces. These forces, in conjunction with the GPU implementation of the other forces, enable the user to participate in the optimization process, and guide the optimization. The motivation for bringing the user in the loop is two-fold:

1. It allows the user to prevent the correspondence computation from getting caught in a local optimum. In such a case, the user can pause the simulation (disabling other forces) and interact with the active visage, updating the optimization estimate through the directable force alone. Once the estimate is free of the local optimum, the optimization can be resumed to snap the solution into place for optimal correspondence.
2. It furthermore, allows to user to steer the solution (as the simulation is running) to a subjectively preferred solution: the correspondence with the lowest error does not necessarily correspond to artifact-free blendshapes.

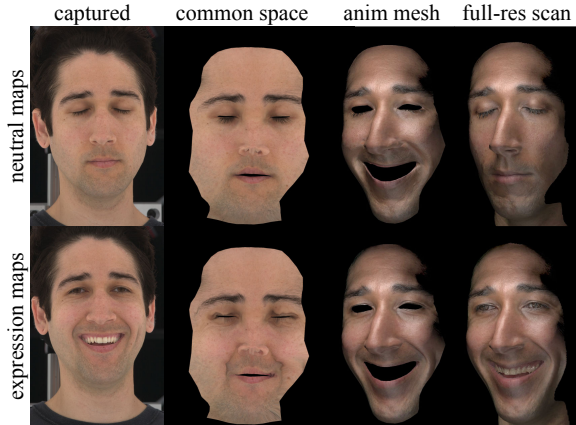
While there are many possible ways for the user to specify directional forces, we have implemented two: dragging of points on the manifold, and pinning (i.e., temporarily fixing) points on the manifold. As was the case with any of the prior forces, care has to be taken when remapping the corresponding directional force to the optimization domain. For example, if the user specifies directable forces in visualizations of the active visage, then the same Jacobian as in

Section 5 needs to be applied. Figure 1, last row right, shows an example of a directable force and its remapping to the target manifold. We refer the reader to the accompanying video and supplemental material for a demonstration of the user-interaction in Facial Cartography.

## 9. Results

We applied our technique to two facial expression datasets, each containing approximately 30 high resolution scans of a subject in various facial expressions. Each scan, acquired using the technique of [MHP\*07], includes a high-resolution geometry (approximately 1.5 M polygons) and photometric textures and normal maps (approximately 2K resolution), of diffuse and specular components. Note that our correspondence technique is equally suited to data obtained using other high-quality scanning methods such as [WMP\*06, BBB\*10].

Each dataset includes a neutral pose, which an artist remeshed into a low-polygon (approximately 2000 polygons) animation mesh along with a UV texture space layout for the detail maps. We then used our technique on each expression to find the correspondences to neutral which would provide the best visual consistency for the desired animation mesh. Figure 2 shows a sequence of intermediate results (visualized using active visages) of the correspondence optimization for a single expression at a selected number of iterations.



**Figure 3:** Facial expression corresponded to a subject’s neutral scan. Original captured maps (1<sup>st</sup> column) are mapped to the artist-defined common space (2<sup>nd</sup> column) using the computed correspondences. The animation mesh (3<sup>rd</sup> column), deformed according to the correspondences, is rendered with neutral and expression maps, producing a consistent result which is faithful to the rendering of the full-res scan data (lower right). Diffuse color texture maps are shown; renderings are performed using texture and photometric normal maps.

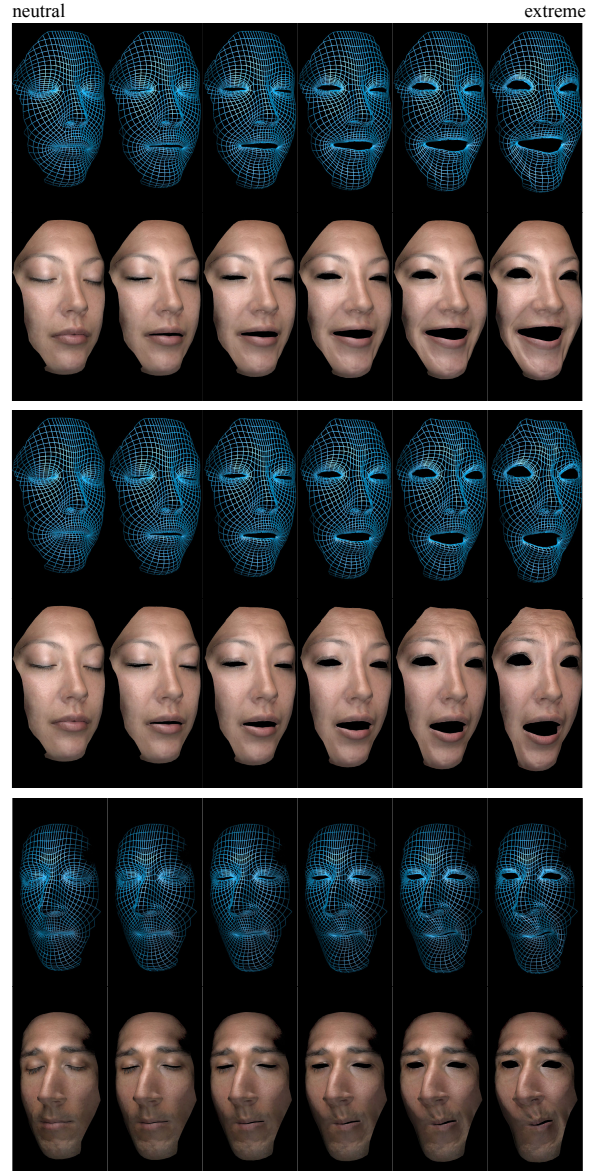
The obtained correspondences allow us to deform the animation mesh into each expression; remap the detail maps from each expression into the artist-defined UV texture space (Figure 3); and therefore blend high-resolution details from different expressions as we deform the low-resolution animation mesh (Figure 4). In facial animation applications, the interpolation weights (for both vertex displacements and detail maps) would come from rig controls and their associated weight maps, as in [ARL\*10].

**Comparison to other techniques.** We compare our technique with two alternatives for computing high-quality facial scan correspondences:

1. optical flow based alignment (based on [BBPW04]), and
2. manual (skilled) artistic alignment.

We compute correspondences on four different extreme facial expressions, each having one or more regions of pronounced deformation, as well as a change in topology compared to the neutral mesh (i.e., opening or closing of eyes and mouth). To evaluate how well the three correspondence methods line up fine-scale details, we remap the bandpass-filtered detail texture into a common, artist-defined UV texture space, and compute the difference between the remapped and the neutral detail texture. A visual comparison of the three different methods, applied to three of the selected extreme expressions, is shown in Figure 5. The average RMS errors, from the four selected extreme expressions, are summarized in Table 1.

While the optical flow approach is automatic and offers

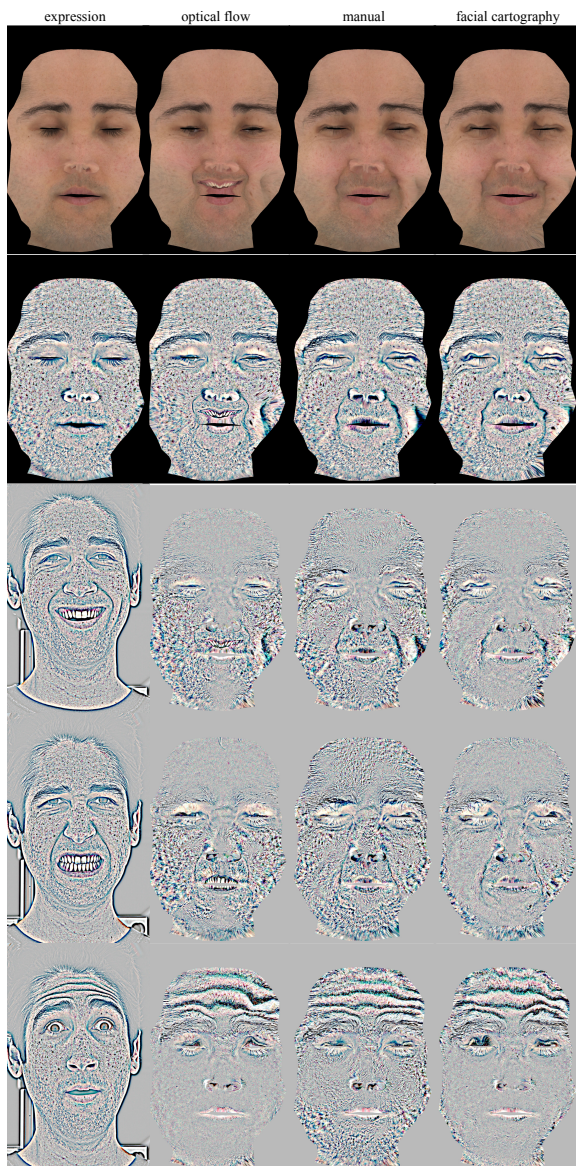


**Figure 4:** Synthesis of intermediate expressions by combining scans which have been mapped into a common domain.

precise alignment (low average RMS error) it can fail on particularly challenging cases such as those in Figure 5. We used the optical flow algorithm of Brox et al. [BBPW04], which we have found to be accurate and robust, albeit computationally intensive. Optical flow systems can be finicky in general, succeeding in some cases but failing entirely on seemingly similar cases. Typical failure cases are expressions in which substantial areas of one image (e.g., mouth interior) are not present in the target image (e.g., mouth closed).

The manual approach consistently achieves good corre-





**Figure 5:** Comparison. The diffuse texture of each expression is mapped to the common UV space using each technique (1<sup>st</sup> row). We assess fine-scale alignment by high-pass filtering the textures (2<sup>nd</sup> row) and computing the difference between expression and neutral textures after registration (rows 3 – 5).

spondences even in challenging cases. However with a manual approach it is impractical to precisely align fine-scale details (such as skin pores), as seen in Figure 5. Alignment of such features is important if high-resolution detail maps are to be blended seamlessly, without introducing ghosting artifacts (see video). The results shown for the manual approach were obtained after approximately 30 minutes of work per expression by a skilled artist. An interesting observation is

method	average RMS error
optical flow	0.16
manual	0.21
Facial Cartography	0.16

**Table 1:** Average RMS error on corresponded expressions computed using three different registration algorithms.

that the artist reported diminishing returns: additional small improvements would require increased amounts of time, indicating that there is a practical upper-limit to how well an artist can correspond high resolution blendshapes.

Like the manual approach, the interactive Facial Cartography approach achieves good correspondences even in challenging cases. Furthermore it greatly outperforms the manual approach on precision, achieving significantly lower average RMS error, and doing so in a fraction of the time (approximately 8 minutes per expression).

**Limitations.** Compared to a fully automatic method, giving the user more control, and thus responsibility, results in a tradeoff. On one hand, it is now up to the user to decide whether convergence is reached, as it is hard to formulate a good metric given the element of unpredictability introduced by bringing the user in the loop. On the other hand, this also implies that the user can decide to continue working on the result until a satisfactory result is reached. Furthermore, we also rely on the user to help the algorithm to do the “right thing” when the algorithm does not “know” what the right thing is. However in a production context, having this ability to correct and direct, is preferable to being fully automatic but unable to change a “wrong” result.

Another limitation of Facial Cartography is that deep wrinkles can occlude small areas of the face for some image force viewpoints. Consequently, the image forces in these hidden areas may not constrain the correspondence computations sufficiently to obtain preferable results. Using more viewpoints or using an advanced physical model of skin buckling as a prior could mitigate this.

Finally, on less-challenging correspondence cases, fully automated methods are more likely to converge to a desirable result faster than Facial Cartography. The current optimization scheme employed in Facial Cartography is not as sophisticated as some of the schemes used in advanced automatic correspondence methods. However, the goal of Facial Cartography is not to obtain correspondences in the fewest number of iterations, but to make the optimization accessible, intuitive, and directable.

## 10. Conclusion

In this work we introduced a novel technique, called Facial Cartography, for determining correspondences between facial scans such that they can be mapped to a common domain for use in animation. Our approach allows the user to

participate, interactively, in the optimization as an integral part of the computations. This provides practical benefits: empowering the artist to guide the solution to a subjectively preferred solution; assisted by computation which maintains a consistent solution which retains detail from the original measurements.

To make this interplay possible, we propose a novel representation, called the Active Visage, which maintains the advantages of deformable template meshes and computations in a 2D canonical domain, while avoiding their disadvantages. Furthermore, the components of our system such as the analysis-by-synthesis component, together with the tightly coupled interaction, may prove useful for a variety of applications. Therefore in future work we plan to investigate how our flexible framework can be extended to related problems in facial animation, such as using captured performance data to generate control curves for a specific animation rig.

**Acknowledgments.** We thank N. Palmer-Kelly, M. Liewer, G. Storm, B. Garcia, and T. Jones for assistance; and S. Mordijk, K. Haase, G. Benn, J. Williams, M. Trimmer, K. LeMasters, B. Swartout, R. Hill, and R. Hall for generous support. The work was partly supported by the Office of Naval Research, NSF grant IIS-1016703, the University of Southern California Office of the Provost and the U.S. Army Research, Development, and Engineering Command (RDECOM). The content of the information does not necessarily reflect the position or the policy of the US Government, and no official endorsement should be inferred.

## References

- [ACP02] ALLEN B., CURLESS B., POPOVIĆ Z.: Articulated body deformation from range scan data. *ACM Trans. Graph.* 21, 3 (2002), 612–619. 2, 3
- [AKS\*05] ANGUELOV D., KOLLER D., SRINIVASAN P., THRUN S., PANG H.-C., DAVIS J.: The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces. In *NIPS* (2005). 2
- [ARL\*10] ALEXANDER O., ROGERS M., LAMBETH W., CHIANG M., MA W.-C., WANG C., DEBEVEC P.: The digital emily project: Achieving a photorealistic digital actor. *IEEE Comp. Graph. and App.* 30 (July/Aug. 2010). 1, 8
- [ARV07] AMBERG B., ROMDHANI S., VETTER T.: Optimal step nonrigid icp algorithms for surface registration. *IEEE CVPR 0* (2007), 1–8. 2
- [ASK\*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: Scape: shape completion and animation of people. *ACM Trans. Graph.* 24, 3 (2005), 408–416. 2
- [BBB\*10] BEELER T., BICKEL B., BEARDSLEY P., SUMNER B., GROSS M.: High-quality single-shot capture of facial geometry. *ACM Trans. Graph.* 29, 4 (July 2010), 40:1–40:9. 7
- [BBPW04] BROX T., BRUHN A., PAPANBERG N., WEICKERT J.: High accuracy optical flow estimation based on a theory for warping. In *ECCV* (2004), pp. 25–36. 8
- [BBVP03] BLANZ, VOLKER, BASSO, CURZIO, VETTER, THOMAS, POGGIO, TOMASO: Reanimating Faces in Images and Video. *Comp. Graph. Forum* 22, 3 (2003), 641–650. 3
- [BM92] BESL P. J., MCKAY N. D.: A method for registration of 3-d shapes. *IEEE PAMI* 14 (Feb. 1992), 239–256. 2
- [BR04] BROWN B. J., RUSINKIEWICZ S.: Non-rigid range-scan alignment using thin-plate splines. In *3DPVT* (2004), pp. 759–765. 2
- [BR07] BROWN B., RUSINKIEWICZ S.: Global non-rigid alignment of 3-D scans. *ACM Trans. Graphs.* 26, 3 (Aug. 2007). 2
- [BV99] BLANZ V., VETTER T.: A morphable model for the synthesis of 3d faces. In *Proc. SIGGRAPH* (1999), pp. 187–194. 3
- [CZ09] CHANG W., ZWICKER M.: Range scan registration using reduced deformable models. *Comp. Graph. Forum* 28, 2 (2009), 447–456. 2
- [DM96] DECARLO D., METAXAS D.: The integration of optical flow and deformable models with applications to human face shape and motion estimation. In *IEEE CVPR* (1996), p. 231. 3
- [HCTW11] HUANG H., CHAI J.-X., TONG X., WU H.-T.: Leveraging motion capture and 3d scanning for high-fidelity performance acquisition. *ACM Trans. Graph.* 30, 4 (Aug. 2011), 74:1–74:10. 3
- [LAGP09] LI H., ADAMS B., GUIBAS L. J., PAULY M.: Robust single-view geometry and motion reconstruction. *ACM Trans. Graph.* 28, 5 (Dec. 2009). 2
- [LDRS05] LITKE N., DROSKE M., RUMPF M., SCHRÖDER P.: An image processing approach to surface matching. In *Proc. SGP* (2005). 3
- [LF09] LIPMAN Y., FUNKHOUSER T.: Mobius voting for surface correspondence. *ACM Trans. Graph.* 28, 3 (Aug. 2009). 2
- [LSP08] LI H., SUMNER R. W., PAULY M.: Global correspondence optimization for non-rigid registration of depth scans. *Proc. SGP* 27, 5 (July 2008). 2
- [MHP\*07] MA W.-C., HAWKINS T., PEERS P., CHABERT C.-F., WEISS M., DEBEVEC P.: Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination. In *Rendering Techniques* (2007), pp. 183–194. 7
- [PKF07] PONS J.-P., KERIVEN R., FAUGERAS O.: Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *IJCV* 72 (April 2007), 179–193. 3
- [PSS99] PIGHIN F., SZELISKI R., SALESIN D. H.: Resynthesizing facial animation through 3d model-based tracking. In *In ICCV* (1999), pp. 143–150. 3
- [SP04] SUMNER R. W., POPOVIĆ J.: Deformation transfer for triangle meshes. *ACM Trans. Graph.* 23 (Aug. 2004), 399–405. 2
- [ST94] SHI J., TOMASI C.: Good features to track. In *IEEE CVPR* (1994), pp. 593 – 600. 5
- [WMP\*06] WEYRICH T., MATUSIK W., PFISTER H., BICKEL B., DONNER C., TU C., MCANDLESS J., LEE J., NGAN A., JENSEN H. W., GROSS M.: Analysis of human faces using a measurement-based skin reflectance model. *ACM Trans. Graph.* 25, 3 (2006), 1013–1024. 7
- [WWJ\*07] WANG S., WANG Y., JIN M., GU X. D., SAMARAS D.: Conformal geometry and its applications on 3d shape matching, recognition, and stitching. *IEEE PAMI* 29 (2007). 2
- [ZWW\*10] ZENG Y., WANG C., WANG Y., GU X., SAMARAS D., PARAGIOS N.: Dense non-rigid surface registration using high-order graph matching. In *CVPR* (2010), pp. 382–389. 2
- [ZZW\*08] ZENG W., ZENG Y., WANG Y., YIN X., GU X., SAMARAS D.: 3d non-rigid surface matching and registration based on holomorphic differentials. In *ECCV* (2008), pp. 1–14. 2, 3