Blended Intrinsic Maps

Vladimir G. Kim   Yaron Lipman   Thomas Funkhouser

Princeton University

Abstract

This paper describes a fully automatic pipeline for finding an intrinsic map between two non-isometric, genus zero surfaces. Our approach is based on the observation that efficient methods exist to search for nearly isometric maps (e.g., Möbius Voting or Heat Kernel Maps), but no single solution found with these methods provides low-distortion everywhere for pairs of surfaces differing by large deformations. To address this problem, we suggest using a weighted combination of these maps to produce a "blended map." This approach enables algorithms that leverage efficient search procedures, yet can provide the flexibility to handle large deformations.

The main challenges of this approach lie in finding a set of candidate maps $\{m_i\}$ and their associated blending weights $\{b_i(p)\}$ for every point p on the surface. We address these challenges specifically for conformal maps by making the following contributions. First, we provide a way to blend maps, defining the image of p as the weighted geodesic centroid of $m_i(p)$. Second, we provide a definition for smooth blending weights at every point p that are proportional to the area preservation of m_i at p. Third, we solve a global optimization problem that selects candidate maps based both on their area preservation and consistency with other selected maps. During experiments with these methods, we find that our algorithm produces blended maps that align semantic features better than alternative approaches over a variety of data sets.

Keywords: inter-surface map, inter-surface correspondences

Links: ⬛DL 🅿️PDF ⬛WEB 📁DATA ⬇CODE

1 Introduction

Finding a map between two surfaces is a fundamental problem in computer graphics with applications in morphing, texture transfer, geometry synthesis, and animation. For many of these applications, the objective is to find an intrinsic map $f : \mathcal{M}_1 \to \mathcal{M}_2$, for a pair of non-isometric meshes $\mathcal{M}_1$ and $\mathcal{M}_2$, such that f is smooth and "low-distortion" everywhere (as isometric as possible). With such a map, it is possible to transfer attributes [Kraevoy and Sheffer 2004], study surface variations [Allen et al. 2003], and process meshes consistently [Golovinskiy and Funkhouser 2009].

The general approach to this problem is to search a discrete space of possible maps, selecting the one that minimizes a prescribed distortion measure. With this discrete formulation, the key challenge is to select a space of maps that is both small enough to search efficiently and large enough to contain useful maps between non-isometric
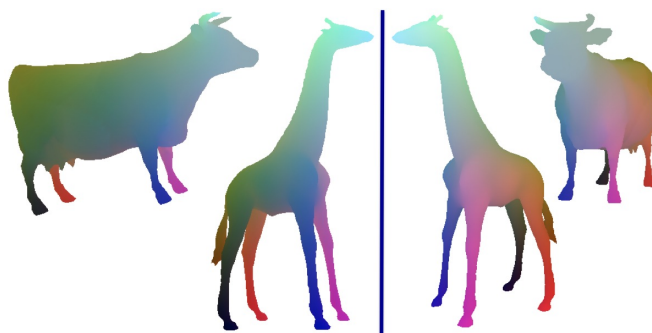

Figure 1: *Automatically-extracted map f between cow and giraffe (same map is rendered from two viewpoints). We color each vertex on giraffe's body by it's $\{x, y, z\}$ position. Then every vertex v on the cow's body is mapped to the giraffe by f, and colored the same as $f(v)$*

surfaces found in real-world problems. One approach is to search an exponentially large space of maps (e.g., all $N!$ sets of correspondences between N sparse feature points), which can include a wide variety of useful deformations, but requires an NP-Hard search algorithm. An alternative approach is to search a low-dimensional space of intrinsic maps (e.g, using geodesic feature vectors, Heat-Kernel maps, conformal maps, etc.), where polynomial-time search algorithms are available, but whose variety of deformations is limited. The problem is that no known space of maps is both polynomial in size and contains the deformations commonly found in real-world surface correspondence problems (e.g., even articulations of people and animals can deviate significantly from conformality or isometry), and so there is not an obvious solution to this problem.

Our approach is to search for a continuous blend of multiple low-dimensional maps. By combining maps with weights varying smoothly over the surface, we define a space of maps that includes a large range of deformations, yet still can be searched with polynomial-time algorithms. In this paper, we consider blends of conformal maps with weights that: 1) are proportional to the area-preservation of the map at every point, and 2) incorporate global similarity relations between different conformal maps. In this way, we favor maps that locally aim to preserve both angles and areas (i.e., near-isometries), but globally are consistent and can achieve extreme deformations.

This method finds a smooth map in polynomial time that empirically aligns semantic features of non-isometric meshes effectively. During experiments with a test set of 334 surface pairs, our blended map is able to align benchmark correspondence points on different meshes within the same object type better than several state-of-the-art methods. For example, a blended map between a cow and a giraffe is shown in Figure 1 (a failure case in [Lipman and Funkhouser 2009]) – note that the map is nearly-isometric locally, even though it provides a smooth map between significantly different shapes.

Our paper makes four main research contributions: 1) the idea of combining multiple low-dimensional intrinsic maps to produce a blended map, 2) an objective function for a weighted collection of maps that favors both confidence of maps and consistency between pairs of maps, 3) a method for estimating the consistency of two maps at a point, and 4) an optimization pipeline that produces a

globally optimal weight assignment for a set of maps. These methods are implemented in a test suite of code and data that is publicly available at http://www.cs.princeton.edu/~vk/CorrsBlended/.

2 Previous Work

Finding correspondences between surfaces is a long standing problem with a rich variety of previous methods and applications [van Kaick et al. 2010; Chang et al. 2010; Bronstein et al. 2008].

Inter-surface mapping Given a set of correct sparse correspondences (defined by a user or an algorithm), one can use a variety of methods to find a smooth map interpolating them. A common approach is to map both surfaces to a canonical domain where sparse feature points align and then interpolate the map in that domain [Alexa 2001]. For example, [Praun et al. 2001] used a base coarse mesh (provided by a user) as such a domain. In their approach, the surface is cut into triangular patches defined by three geodesic curves, such that each geodesic curve is mapped to a triangle on a coarse base mesh. Further, [Schreiner et al. 2004; Kraevoy and Sheffer 2004] developed an automatic approach for creating the base domain. These methods, however, were only evaluated with manually labeled sparse correspondences as their input, and they are too expensive to be used for finding sparse correspondences in a fully automatic algorithm.

Finding sparse correspondences Several methods have been proposed for automatically finding a small (sparse) set of feature correspondences, which could be used to produce an inter-surface map. The most common approach of this type is to first extract a set of feature points and then to explore permutations of them to find the correspondences implying an alignment with minimal deformation error [Huang et al. 2008; Zhang et al. 2008]. This approach is effective when local shape descriptors at the feature points are very distinctive, but quickly becomes too expensive when local shapes are different and measurement of global deformations implied by many points are required for discriminating the optimal solution. Even with pruning based on branch-and-bound [Gelfand et al. 2005] or priority-driven search [Funkhouser and Shilane 2006], the search space is simply too large to explore efficiently, and so heuristics are employed and/or few feature correspondences are found, which makes finding a good inter-surface map difficult.

Iterative closest points Some methods find surface correspondences through an iterative procedure that starts with an initial correspondence and then repeatedly improves it by computing an aligning transformation from the correspondences and then updating the correspondences based on the transformation (e.g., based on mutually closest points). This method is most commonly used for aligning surfaces related by a rigid transformation [Besl and McKay 1992], but has also been used for moderate non-rigid deformations [Allen et al. 2003; Brown and Rusinkiewicz 2007; Li et al. 2008; Pauly et al. 2005; Tevs et al. 2009; Ghosh et al. 2009]. Unfortunately, it does guarantee that the final map is smooth or bijective (two points on one surface may map to the same point on another), and it requires a good initial guess to succeed in most cases.

Finding dense correspondences Other methods directly find correspondences for all points on a surface. For example, the Gromov-Hausdorff distance motivated a purely intrinsic approach by [Mémoli and Sapiro 2004] to measure deviation from isometry between two surfaces, and [Bronstein et al. 2006; Bronstein et al. 2008] developed a Generalized Multidimensional Scaling (GMDS) framework to find the least distortion embedding of one surface onto another. These methods use an approximate search procedure to find an initial guess of point correspondences and thus may converge to a local minimum.

Surface embedding Some methods find dense correspondences by embedding surfaces in feature space where similar points have similar coordinates and then produce a dense map based on nearest neighbors in that space. For example, [Ovsjanikov et al. 2010] showed that a single correspondence can define a Heat Kernel Map (HKM), a high dimensional embedding of a surface invariant under isometry. The disadvantage of these methods is that they are effective only for deformations that are nearly isometric – otherwise features do not align in the embedded space. [Ovsjanikov et al. 2010] suggested a simple extension of their approach to non-isometric cases: they concatenate features from two heat kernel maps generated by two correspondences into a single feature vector and search for nearest neighbors in that space. However, still it is not obvious how to best select multiple correspondences, and still the resulting map is not guaranteed to be smooth, or even continuous, when surfaces are not isometric.

Exploring Möbius Transformations The methods most similar to ours are the ones of [Lipman and Funkhouser 2009] and [Kim et al. 2010]. They both leverage the fact that isometries are a subspace of the conformal maps, which are low-dimensional and can be explored efficiently with Möbius Transformations. They differ in the way they search and combine the maps: [Kim et al. 2010] uses a RANSAC algorithm to discover the single "best" conformal map that maps a surface onto its reflection, while [Lipman and Funkhouser 2009] combines multiple maps with an algorithm that votes for correspondences. The former approach works only for nearly-isometric surfaces (e.g., intrinsic symmetries), while the latter approach may produce globally inconsistent correspondences (when votes for inconsistent maps combine in the correspondence matrix).

Overall, to our knowledge, there is no existing fully automated method that finds a smooth, low-distortion map between significantly non-isometric surfaces in polynomial time.

3 Key Idea

Our approach is to search for a map that smoothly blends multiple low-dimensional maps. This approach allows a search procedure to explore a polynomial size space of maps, while providing the flexibility to find smooth maps between surfaces differing by significant deformations.

A motivation for this approach is provided in Figure 2. Our goal in this example is to produce a smooth map between the surfaces of a person in two different poses, such that the map is "as isometric as possible." Although the surfaces are nearly isometric, there is no known low-dimensional map that takes one surface onto the other with small distortion everywhere. For example, the top row shows three conformal maps, each of which provides low distortion (blue) for most of the body, but has large distortions (red) on different parts of the arms and head. While none of these conformal maps provide a good solution for the entire body, they can be combined with weights (bottom left) to form a blended map with small distortion almost everywhere (bottom right).

In this paper, we investigate ways to define blended maps and compute them automatically for pairs of genus zero surfaces. Specifically, given a pair of surface meshes $\mathcal{M}_1$ and $\mathcal{M}_2$, our goal is to find a set of K candidate maps $\{m_i\}_{i=1}^{K} : \mathcal{M}_1 \to \mathcal{M}_2$ associated with smooth blending weights $b_i(p)$ for every point p, such that the blended map $f : \mathcal{M}_1 \to \mathcal{M}_2$ defined as follows has low distortion across the entire surface:

$$f(p) = \operatorname*{argmin}_{p' \in \mathcal{M}_2} \sum_{i=1}^{K} b_i(p) d_{\mathcal{M}_2} \left(p', m_i(p)\right)^2 , \qquad (1)$$
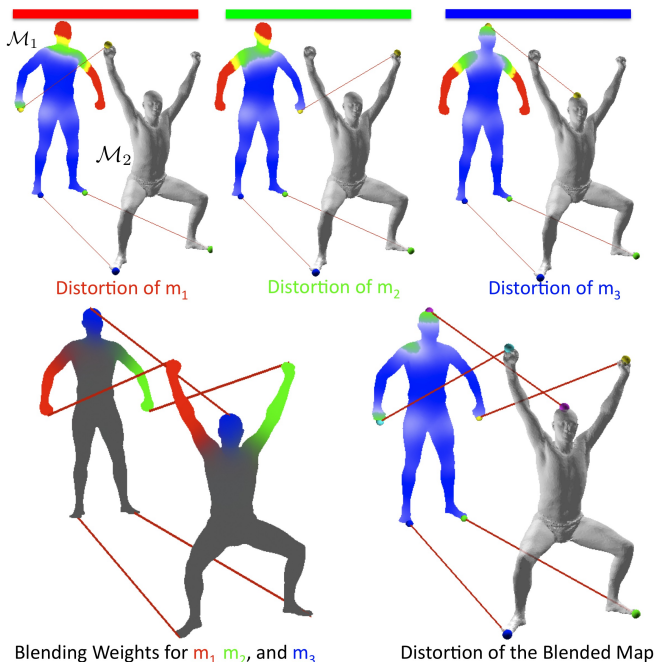
Figure 2: *Motivation for our approach. In the top row, three low-dimensional conformal maps $m_1, m_2,$ and m_3 are defined by three correspondences rendered as red lines. Map confidence is rendered in color on a surface, where blue corresponds to 0 area distortion, and red is high distortion. The three low dimensional maps are blended according to weights encoded as RGB colors in the bottom-left image. The resulting blended map in the bottom-right corner has lower local distortion than any of the three maps above.*

where $d_{\mathcal{M}_2}(\cdot, \cdot)$ denotes the geodesic distance on surface $\mathcal{M}_2$.

Intuitively, this definition maps every point p to the weighted geodesic centroid of its images $\{m_i(p)\}$ for all maps $\{m_i\}_{i=1}^K$. We use this definition because it guarantees smoothness of the blended map if the candidate maps $\{m_i(p)\}$ and blending weights $\{b_i(p)\}$ are both smooth, and because we believe that it contains most deformations commonly found in real-world surface correspondence problems. For instance, in the example shown in Figure 2, weights are chosen such that only one of the three candidate maps influences each of the arms and the head strongly, but they blend together smoothly to form a map with no discontinuities and low overall distortion on the body.

Given this definition, the key research challenge is to provide automatic methods to generate a set of candidate maps with associated blending weights. Intuitively, the ideal solution should guarantee smoothness of the blending weights $b_i(p)$, assign large weights at p only to maps m_i that induce small distortions at $m_i(p)$, and assign non-zero weights at p only for sets of maps that are consistent with one another (i.e., if $b_i(p) > 0$ and $b_j(p) > 0$, then $d_{\mathcal{M}_2}(m_i(p), m_j(p))$ should be small). If these constraints are satisfied, then the resulting map will be smooth and have low distortion everywhere.

In this paper, we focus on algorithms to address these challenges for blending *conformal* maps. Conformal maps are low-dimensional and thus efficient to search; they preserve angles and thus avoid distortions with shear; they contain isometries as a special case and thus they are a common type of map for non-rigid deformations; and, finally, they simplify the problem of finding blending weights in our formulation, because it is possible to estimate analytically the distortion of a conformal map at given point (e.g., how well area is

preserved in the point's neighborhood). Thus, we can partition the computation of blending weights into two factors:

$$b_i(p) = c_i(p) \cdot w_i(p),$$

where $c_i(p)$ measures the "confidence" of the conformal map m_i at point p based on an estimate of its area-preservation at p, and $w_i(p)$ are the "consistency" weights that indicate to what extent a map should be used for blending. The key observation is that the first factor, $c_i(p)$, already provides a smoothly varying estimate for the distortion of each map m_i at p (as shown in the top row of Figure 2), and thus captures the spatially varying aspects of $b_i(p)$. The second factor, $w_i(p)$, can then be treated as a constant across the surface, which greatly simplifies computation of optimal blending weights.

The following four sections describe our algorithm to compute blended maps between two surfaces automatically. Given two input surfaces, we first generate a set of candidate conformal maps (Section 4). Then, we estimate the confidence $c_i(p)$ for each map at every point p (Section 5). We next compute consistency weights w_i for every conformal map m_i by optimizing an objective function that favors non-zero weights only for sets of maps that are both high confidence and consistent with one another (Section 6). Finally, we produce a final blend with these weights using Equation (1) (Section 7). The methods employed in Sections 4, 5, and 7 are straightforward – detailed descriptions are included mainly for the sake of completeness and reproducibility. Our main algorithmic contribution is in Section 6, which describes a method for finding optimal consistency weights to be used for blending in our formulation.

4 Generating Maps: $\{m_i\}_{i=1}^K$

Our first step is to generate conformal maps that will form a candidate set for blending. Our goal is to provide a small set of maps such that at least one map achieves small distortion at every important feature point, and such that areas mapped with low distortion by different maps overlap significantly so that they can be blended without distortion.

To generate such a candidate set, we follow the procedure used in [Lipman and Funkhouser 2009; Kim et al. 2010]. We first compute a small collection of feature points, $\mathcal{P}_1 \subset \mathcal{M}_1$ and $\mathcal{P}_2 \subset \mathcal{M}_2$, on both surfaces. Then, we generate candidate conformal maps by enumerating triplets of three feature point correspondences, where each pair of triplets uniquely defines a Möbius transformation that maps one surface onto the other conformally while interpolating the feature point correspondences.

Generating feature points: our first task is to generate sets of feature points $\mathcal{P}_1$ and $\mathcal{P}_2$ on $\mathcal{M}_1$ and $\mathcal{M}_1$. Our goal is to produce a small number of feature points with a large fraction of semantic correspondences. Although many methods are possible, we currently extract points at maxima of the Average Geodesic Distance function $AGD_{\mathcal{M}_\ell}(p) = \int_{\mathcal{M}_\ell} d_g(p, p')dA(p')$, where dA denotes the area element on the surface $\mathcal{M}_\ell$, $\ell = 1, 2$. This method provides corresponding feature sets particularly well for articulated figures (e.g., tips of extremities and top of head), and thus a small number of features is usually required to achieve multiple semantic correspondences spread throughout the surfaces. For most examples presented in this paper $|\mathcal{P}_\ell| \leq 10$.

Generating conformal maps: our next task is to generate a candidate set of conformal maps $\{m_i\}_{i=1}^K$. Following [Lipman and Funkhouser 2009], we first conformally map the two surfaces to the extended complex plane using mid-edge uniformization [Pinkall and Polthier 1993]. We then generate a set of conformal maps by

enumerating all possible combinations of three correspondences between feature points in $\mathcal{P}_1$ and $\mathcal{P}_2$ (*generating correspondences*) and construct a conformal map m_i for each one by computing the Möbius transformation that interpolates all three corresponding points. This procedure produces $K = \binom{|\mathcal{P}_1|}{3} \cdot \binom{|\mathcal{P}_2|}{3} \cdot 6$ distinct conformal maps that form the set $\{m_i\}_{i=1}^{K}$ that will be candidates for blending in the following steps. Please refer to [Lipman and Funkhouser 2009; Kim et al. 2010] for details.

5 Defining Confidence Weights: $\{c_i(p)\}_{i=1}^{K}$

Our second step is to compute a confidence value $c_i(p)$ that estimates how much distortion is induced by each map m_i at every point p. Though many formulations are possible to measure distortion at a point, in this work we aim to estimate deviations from isometry.

Since conformal maps preserve angles, we can estimate deviation from isometry simply by measuring the scale factor induced by the map at every point p (isometries are conformal maps that preserve scales). To do so, we define

$$c_i(p) = 2 \bigg/ \left[\frac{area(N_p)}{area(m_i(N_p))} + \frac{area(m_i(N_p))}{area(N_p)} \right], \quad (2)$$

where $area(N_p)$ is the area of a neighborhood, N_p, around point p on $\mathcal{M}_1$ and $area(m_i(N_p))$ is the area of its image, $m_i(N_p)$, on $\mathcal{M}_2$.

For computational efficiency, we calculate $c_i(p)$ only at a set $\mathcal{P}_{\text{even}}$ of 256 approximately evenly distributed points. The set is produced by starting with a random vertex and then iteratively adding vertices that are farthest from the set $\mathcal{P}_{\text{even}}$ until $|\mathcal{P}_{\text{even}}| = 256$ [Eldar et al. 1997]. Confidence weights $c_i(p)$ for all other vertices are calculated using smooth interpolation with Gaussian weights.

These estimates of $c_i(p)$ based on area preservation are quick to compute, vary smoothly across the surface, and correlate well with low-distortion in a map, and thus they provide the desired properties of the spatially varying factor in our blending weights.

6 Finding Consistency Weights: $\{w_i\}_{i=1}^{K}$

The next step is to compute a set of consistency weights $\{w_i\}_{i=1}^{K}$ for all conformal maps $\{m_i\}_{i=1}^{K}$. Ideally, this set will have weights with zero values for conformal maps that induce high distortion (e.g., the generating triplet of feature correspondences contains incorrect matches) and non-zero weights only for conformal maps that are consistent with one another.

6.1 Objective Function

Following this intuition, we define the consistency weights $\vec{w} := \{w_i\}_{i=1}^{K}$ as the minimizer of an objective function, $E(\vec{w})$:

$$E_{\mathcal{M}_1}(\vec{w}) = \sum_{i=1}^{K} \sum_{j=1}^{K} w_i w_j \int_{p \in \mathcal{M}_1} S_{i,j}(p) c_i(p) c_j(p) dA(p)$$
$$\text{subject to } \sum_{i=1}^{K} w_i^2 = 1, \quad (3)$$

where confidence values $c_i(p)$ are defined as in the previous section, and pairwise map consistency values $S_{i,j}(p) : \mathcal{M}_1 \rightarrow \mathbb{R}$ provide an estimate of how consistent two maps are at a point p. We constrain the L_2 norm of weights to be 1 since we want to favor global maps that include multiple similar maps.

Roughly speaking, for every choice of weights $\vec{w}$ giving non zero weights to some subset of maps, the functional measures how pairwise consistent is this set and how well each individual in this set

preserves area. The weights achieving the minimum of this objective function will signal out the correct set of maps to be used in the blending.

Map Consistency The most important term in the objective function is the similarity measure $S_{i,j}(p)$ for a pair of maps m_i, m_j. Intuitively, $S_{i,j}(p)$ should be high if maps m_i and m_j are similar at point p. To model this intuition, we define the consistency for a pair of maps m_i and m_j at a point p to be inversely related to the geodesic distance between images of the point p under the two maps, $m_i(p)$ and $m_j(p)$:

$$S_{i,j}(p) = \exp\left(-\frac{d_{\mathcal{M}_2}(m_i(p), m_j(p))}{\sigma^2} \right) \quad (4)$$

Note that $0 \leq S_{i,j}(p) \leq 1$ at any point p, $S_{i,j}(p) = 1$ iff $m_i(p) = m_j(p)$, and σ is a controllable parameter that controls how close images of a mapped point should be in order for a pair of maps to be considered similar (we use $\sigma = 0.5$ for all results in this paper).

Since calculating $S_{i,j}$ is at the core of our objective function we need to make this computation as efficient as possible. However, for high-resolution meshes, calculating geodesic distances between any arbitrary pair of vertices can be expensive. Instead of calculating Equation (4) directly, we replace it with:

$$S_{i,j}(p) = \exp\left(-\frac{d_{\mathcal{M}_1}\left(p, m_j^{-1}(m_i(p))\right)}{\sigma^2} \right) \quad (5)$$

The reason is that we can then evaluate the consistency function $S_{i,j}$ only for a subset of points $\mathcal{P}_{\text{even}}$ and therefore geodesic distances from point p to every other vertex on $\mathcal{M}_1$ can be precomputed and stored. For example, Figure 4 shows similarity values for two pairs of maps.
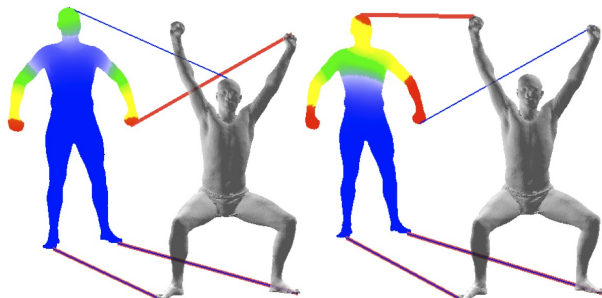

Figure 4: *These images show the similarity function $S_{i,j}(p)$ for a pair of conformal maps m_i, m_j (values range from 0 (red) to 1 (blue)). The generating correspondences are depicted by red lines for map m_i and by blue lines for map m_j.*

6.2 Optimizing for map consistency weights

In this step, we optimize Equation (3) for the consistency weights: $\{w_i\}$. To do this, we define a blending matrix

$$\mathbf{S}_{i,j} = \int_{\mathcal{M}_1} c_i(p) c_j(p) S_{i,j}(p) dA(p).$$

with which Equation (3) can be written as

$$E_{\mathcal{M}_1}(\vec{w}) = \vec{w}^T \mathbf{S} \vec{w}, \ \|\vec{w}\|_2 = 1. \quad (6)$$

Since $\mathbf{S}$ is symmetric, the top eigenvector is the optimal maximal solution (maximizing the Rayleigh quotient $E_{\mathcal{M}_1}$). The Perron-Frobenius theorem assures us that all the entries of this optimal
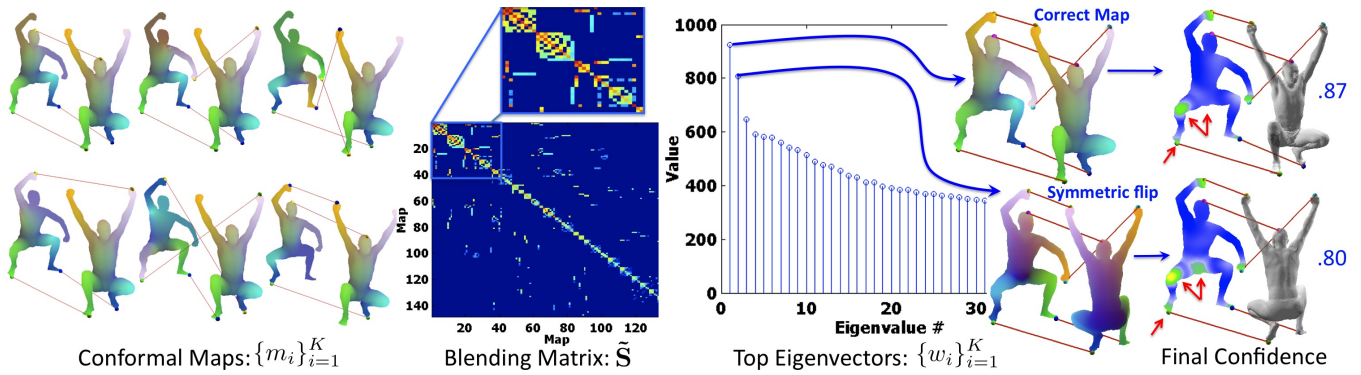
Figure 3: *This figure depicts the blending matrix $\mathbf{S}_{i,j}$ rearranged by consistent blocks of decreasing size. Each row (and column) corresponds to a conformal map. Several conformal maps are illustrated on the left. Note that $\mathbf{S}_{i,j}$ is sparse and contains approximately two large blocks that correspond to the correct near-isometric map and its symmetric flip. We also show the spectrum of this matrix. Note the spectral gap separating the two eigenvectors corresponding to the near-isometries from the rest of the spectrum. On the right we show the two blended maps corresponding to these top two eigenvectors, together with their confidence functions and its integral (single number printed in blue). Note the red arrows that indicate small regions that allow us to distinguish between the correct map and the symmetric flip. For instance, mapping front of a human to the back results in more area distortion at feet (due to heels), at knees, and at buttocks.*

$\vec{w}$ are of constant sign and therefore can be chosen to be positive. Thus, a simple optimization to achieve the consistency weights is to find the top eigenvector $\vec{w}$ of $\mathbf{S}$, define the blending weights $b_i(p) = c_i(p)w_i$, and construct the blended map f as described in Equation (1).

However, there are two issues: 1) the matrix $\mathbf{S}$ for many feature points is large (remember we have $\binom{|\mathcal{P}_1|}{3} \cdot \binom{|\mathcal{P}_2|}{3} \cdot 6$ distinct conformal maps), and 2) in a presence of intrinsic symmetries (or near-intrinsic-symmetries) there is more than one "correct" (i.e., near-isometry) map between the surfaces. We address these issues as follows.

Computing the Blending Matrix Filling in the matrix is the most computationally involved step of our approach. For example, given N feature points on both surfaces, one can construct $\binom{N}{3} \cdot \binom{N}{3} \cdot 6 = O(N^6)$ distinct conformal maps, thus filling the matrix $\tilde{\mathbf{S}}$ requires $O(N^{12}) \cdot |\mathcal{P}_{\text{even}}|$ operations. Fortunately, this matrix is extremely sparse (see Figure 3), and the sparsity can be exploited with a few simple observations. Practically, highly consistent values between conformal maps are possible mainly when they share consistent subsets of generating correspondences. Thus, we only calculate $S_{i,j}$ for pairs of maps that share two (out of a possible three) generating correspondences and do not have any "conflicting correspondences" (i.e., when a feature point on one surface is in correspondence with two different feature points on the other surface), and set $S_{i,j}$ for others to zero. Furthermore, we restrict the maximal number of conformal maps to $10,000$. If more maps generated we randomly remove maps until we are left with $10,000$.

To further speed the computation, we approximate $S_{i,j}$ using a uniform point sampling. Specifically, for each pair of maps, we compute $\mathbf{S}_{i,j}$ by summing over a discrete set of 256 evenly distributed points ($\mathcal{P}_{\text{even}}$):

$$\mathbf{S}_{i,j} \approx \sum_{p \in \mathcal{P}_{\text{even}}} c_i(p)c_j(p)S_{i,j}(p)A_i,$$

where A_i are the constant units of area $\frac{area(\mathcal{M}_1)}{256} = \frac{1}{256}$.

Processing Eigenvectors If either of the two surfaces has an intrinsic near-symmetry, there may be more than one near-isometry

between the surfaces. Hence, there will be more than one group of Möbius transformations such that the corresponding consistency weights $\{w_i\}$ yield a high energy value in Equation (3). In case one of the groups of Möbius transformations is (even slightly) better than the rest (in the sense that its consistency vector produce higher energy), the eigenspaces of $\mathbf{S}$ will naturally separate this better map from the other candidate maps. For example, Figure 3 shows two humans where there are two possible near isometric solutions corresponding to the two top eigenvectors. In this case the better map was characterized by higher energy level (eigenvalue).

Nevertheless, sometimes the different near isometries have very close energy level and the corresponding eigenvectors are blended. In this case any top eigenvector can contain a linear combination of good weight vectors $\vec{w}$ that are originated from different near-isometries of the two surfaces. To avoid blending inconsistent maps we follow the next steps to extract sets of candidate consistent weight vectors $\vec{w}^1, \vec{w}^2, ..., \vec{w}^n$ and analyze them to select the best one.

First we recognize the top eigenvectors by taking all eigenvectors with eigenvalues separated by the spectral gap to the rest of the spectrum of $\mathbf{S}$, see Figure 3. Practically, we take eigenvectors $\vec{w}$ that correspond to eigenvalues within 75% of the top eigenvalue.

Second, we construct the weights $\vec{w}^1, \vec{w}^2, ..., \vec{w}^n$ by separating the different conformal maps with high values in these eigenvectors to different groups $G^1, G^2, ..., G^n$ (clusters) as follows. We start by seeding the first group G^1 to contain the conformal map that corresponds to the top entry (measured in absolute value of the top eigenvector). Then, we traverse the rest of the conformal maps corresponding to high entries (top 25% of that eigenvector). For each conformal map, we check whether its generating correspondences are consistent with the maps chosen already in G^1 (i.e., has no conflicting correspondences). If so, it is not conflicting, and we add it to G^1. Otherwise, we start a new group G^2 seeded with this map. We continue in this fashion until all the eigenvectors belonging to top eigenvalues are processed. We then threshold the weights to $\{0, 1\}$ to enforce the expected block structure of the group in the matrix and to eliminate maps with nearly zero weight from further processing, yielding a set of groups $G_1, .., G_n$ with corresponding binary weights $\vec{w}^1, .., \vec{w}^n$.

The last step is to choose among the different candidate weights

$\vec{w}^1, ..., \vec{w}^n$ the best one. The above procedure generates weights corresponding to clusters of consistent maps, all of which provide an approximately optimal solution to the objective function defined in Equation (3). To select the best among them, we construct from each candidate vector the final blended map $\vec{w}^j \rightarrow f^j$, and pick the blended map that is most confident overall – i.e., globally preserves area best over the whole surface:

$$f = \underset{\{f_j\}_{j=1}^n}{\operatorname{argmin}} \int_{\mathcal{M}_1} c_{f_j}(p) dA(p).$$

For example, the mapping between two human body surfaces would usually generate two weight vectors $\vec{w}^1, \vec{w}^2$: one corresponds to the correct map, and one to an intrinsic rotation by $180°$ under which the front of a human goes to the back, left side maps to right side, etc... Note that both these assignments are globally consistent, and each of the maps have similar confidence to their symmetric counterparts. However, the blended map corresponding to the intrinsic rotation (flip) usually introduces more area distortion for some parts of a body like feet or knees, see Figure 3. This allows us to distinguish between good and flipped solution.

7 The Blended Map

Now we can use low-dimensional maps m_i defined in Section 4 with the importance weights w_i obtained in Section 6 to construct the blending map defined in Equation (1). Note that as long as our confidence $c_i(p)$ changes smoothly over the surface the resulting blending map will also be smooth.

Figure 5 shows how $f(p)$ is found at a point p by blending $m_i(p)$ with the calculated weights. For efficiency sake, we find it useful to approximate the geodesic centroid by projecting the weighted Euclidean centroid $\tilde{f}(p) = \sum_i b_i(p)m_i(p) \big/ \sum_i b_i(p)$ onto the closest point on the surface $\mathcal{M}_2$, an alternative that trades efficiency for accuracy. This is a favorable trade-off because weights used for blending tend to be non-negligible only for a small number of points concentrated very close to one another, in practice, in which case centroids based on Euclidean distances provide a good approximation.
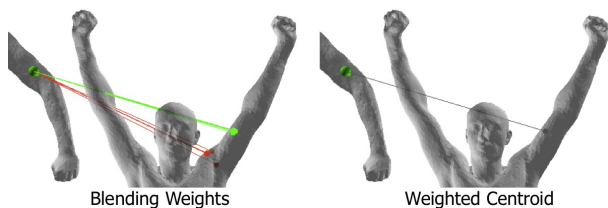

Figure 5: *Correspondences due to conformal maps with non-zero blending weight are depicted on the left image, where the color is set according to confidence $c_i(p)$ ranging from red $= 0$ to green $= 1$. The resulting approximate geodesic centroid is on the right image.*

8 Results

We test our approach on a benchmark constructed from TOSCA, SCAPE and Watertight data sets. We quantitatively analyze performance of our algorithm in various settings and compare results to several state of art methods for finding inter-surface maps and correspondences.

8.1 Data Sets

We selected three data sets that have a large variety of objects with ground-truth correspondences:

SCAPE 71 meshes representing a human body in different poses [Anguelov et al. 2004]. All the meshes were fit to scanner data with a common template, and thus they share the same mesh topology, providing a ground truth map for every vertex for any pair of surfaces (corresponding colors in Figure 6a).

TOSCA 80 meshes representing people and animals in a variety of poses [Bronstein et al. 2008]. The meshes appear in 8 groups with common topology, providing a per vertex ground truth map for any pair within a class (corresponding colors in Figure 6b).

Watertight: 400 meshes arranged evenly in 20 object categories, many of which are articulated figures (humans, octopus, four-legged animals, ants, etc.). The meshes were originally created for the SHREC 2007 Watertight Shape Retrieval Contest [Giorgi et al. 2007]. We selected 11 classes for our experiments that have well defined correspondences and genus zero (Human, Glasses, Airplane, Ant, Teddy, Hand, Plier, Fish, Bird, Armadillo, Four-legged Animal). In addition, we excluded two human models with non-zero genus.

In cases where no ground truth map was provided with a data set (e.g., Watertight), we established a sparse set of "ground truth" correspondences manually. Specifically, we recruited a volunteer to use an interactive program to select 10-35 semantically meaningful feature points in a manner that is consistent across all meshes within the same object class. For example, our volunteer selected 35 feature points for each human and 20 feature points for each four-legged animal (Figure 6c). These feature points form the basis for establishing symmetric correspondences and for evaluating maps between surfaces in the same object class.
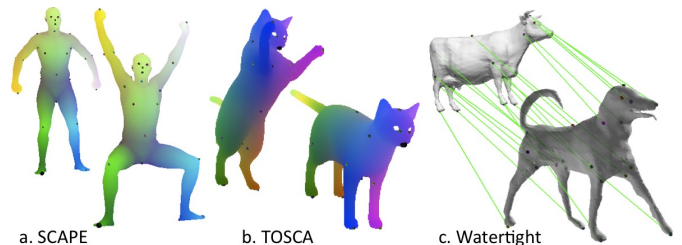

a. SCAPE   b. TOSCA   c. Watertight

Figure 6: *Ground truth examples. SCAPE and TOSCA models are colored according to ground truth per-vertex correspondences.*

8.2 Evaluation Methods

To evaluate the accuracy of a predicted map, $f : \mathcal{M}_1 \rightarrow \mathcal{M}_2$ with respect to a "ground truth" map, $f_{\text{true}} : \mathcal{M}_1 \rightarrow \mathcal{M}_2$, we compute for every point, p, on $\mathcal{M}_1$ in the ground truth correspondence the geodesic distance, $d_{\mathcal{M}_2}(f(p), f_{\text{true}}(p))$, between its image in the predicted map, $f(p)$, and its true correspondence, $f_{\text{true}}(p)$.

We aggregate these geodesic distances into an error measure:

$$Err(f, f_{\text{true}}) = \sum_{p \in \mathcal{M}_1} d_{\mathcal{M}_2}(f(p), f_{\text{true}}(p))$$

where $d_{\mathcal{M}_2}(f(p), f_{\text{true}}(p))$ is normalized by $\sqrt{Area(\mathcal{M}_2)}$, as all distances are throughout this paper.

We also generate plots to examine the distributions of errors, where the x-axis represents a varying geodesic distance threshold, D, and the y-axis shows the average percentage of points for which $d_{\mathcal{M}_2}(f(p), f_{\text{true}}(p)) < D$ (Figure 7 provides a scale bar for D).

To separate errors due to poor alignments from ones due to symmetric flips, we produce two such plots. The first is as already described. The second is similar, but factors out the effects of confusion in the predicted map due to global intrinsic reflective symmetries (e.g., bilateral symmetries that map the left side of a human to the right side). It plots the fraction of predicted correspondence points that either are closer than the geodesic distance threshold D to the correct correspondence point *OR* are closer than the threshold to the symmetric image of the correct correspondence point – i.e., it provides no penalty for predicting a map that is a symmetric flip with respect to the correct one. These plots favor methods that do not preserve orientations in predicted maps (other methods, not ours, as conformal maps are orientation-preserving).

Figure 7: *Reference for normalized geodesic distances on surfaces (measured to the nearest seed point). Colors are labeled by distances as shown in the legend on the right hand side.*

8.3 Comparison to Other Methods

We compare our work to several state of the art methods for finding inter-surface correspondences:

- **Blended Map** - the method proposed in this paper

- **Best Conformal** - the least-distortive conformal map roughly describes what is the best performance achieved by a single conformal map without blending.

- **Möbius Voting*** - the method proposed by [Lipman and Funkhouser 2009] explores millions of conformal maps generated by random triplets on a surface, and votes for correspondences generated by area-preserving maps. The output of this method is 50-100 coarse correspondences.

- **Heat Kernel Matching (HKM) with 1 correspondence** - This method is based on matching features in a space of a heat kernel for a given source point as described in [Ovsjanikov et al. 2010]. A full map is constructed from a single correspondence, which is obtained by searching a correspondence that gives the most similar heat kernel maps. We use code provided by authors for this experiment.

- **HKM with 2 correspondences** - in a non-isometric case the previous method might obtain better results by using a second correspondence. The matching is then performed in the augmented feature space of two heat kernel maps. With minimal changes to the original author's code we follow a procedure outlined in [Ovsjanikov et al. 2010] exhaustively searching for the second correspondence that minimizes the geodesic distortion.

- **GMDS*** - we use the method of [Bronstein et al. 2006] for surface matching. Authors hierarchically find correspondences between points by searching for assignment that best preserve geodesic features. We use authors implementation of this method with default parameter settings to find 50 coarse correspondences.

Note that some of these methods (marked with a "*" in the list above) only produce a sparse set of point correspondences, rather than a full surface map as required for comparison with our evaluation metrics. In those cases, we produce a full map from surface $\mathcal{M}_1$ to $\mathcal{M}_2$ by interpolating the sparse correspondence to using a method based on GMDS – we compute for each vertex on $\mathcal{M}_1$ the geodesic distances to all sparse correspondence points on $\mathcal{M}_1$, and then establish a correspondence to the vertex on $\mathcal{M}_2$ with the most similar distances to sparse correspondence points on $\mathcal{M}_2$ [Bronstein et al. 2008]. This method was chosen because it is simple to implement and because the accuracy is sufficient when a large number of sparse correspondences is provided, as is the case for all methods considered in this study.

Note also that the code available for some of these other methods crashed on meshes in our test data sets (8 in all). To keep comparisons fair, we eliminated those meshes from our evaluation in all experiments.

Near isometric pairs in TOSCA In our first experiment, we studied how methods perform for nearly isometric pairs of surfaces from the TOSCA data set. Specifically, for each model, we picked a random model within the same class, and then computed a map from one to the other use each algorithm. Results of this experiment are shown in Figure 8, where each curve depicts the percentage of correspondences with error below some normalized geodesic distance for a different method. Please note that our method (blue curve) detects over 75% correct correspondences for a small threshold of 0.05 and converges to finding almost all correct correspondences within geodesic error 0.2. We visually examine maps produced with our method and observe several small misalignments in some faces and limbs of animals mostly due to badly selected feature points.

In the bottom image of Figure 8, we show the fraction of correspondences within the distance threshold, if we also allow maps that invert surface orientation. This plot reveals that methods based on geodesic distances: GMDS and HKM (green, magenta and black curves) are commonly confused by bilateral reflective symmetry and intrinsic 180° rotation present in humans and animals. Note that although methods based on conformal geometry cannot produce a reflected solution, they still can intrinsically rotate a surface by 180° mapping front of a human to the back. In this case, errors also become smaller, since distances between front and back of a limb are smaller than distances between left and right limbs. Still the ranking of methods is largely the same.

Near isometric pairs in SCAPE The next experiment compared maps found between nearly isometric models in the SCAPE data set. For each SCAPE model, we picked another one at random, totaling to 71 pairs, and computed the mapping. Results are shown in Figure 9. These meshes are less smooth than TOSCA meshes, which explains the decrease in performance for algorithms based on conformal geometry, since mid-edge uniformization suffers from non-delaunay triangles, but does not affect GMDS (green) and heat kernel methods (magenta and black). The main source of error in this data set is confusion due to intrinsic symmetry in humans. Note that due to the aforementioned problem with individual conformal maps the resulting blended map in some cases does not have enough accuracy to distinguish between front and back of a human. Also note that although Möbius Voting (red) always provides better coverage than the best conformal map it suffers from picking inconsistent correspondences, which explains similar performance of these two methods in this experiment.

Non-isometric humans In a realistic settings, it is desirable to obtain correspondences between surfaces with different resolutions, tessellations, and even semantic variations. In this experiment, we used all 71 human models from SCAPE data set, 43 from TOSCA (including gorilla), and 18 from SHREC (excluding two humans
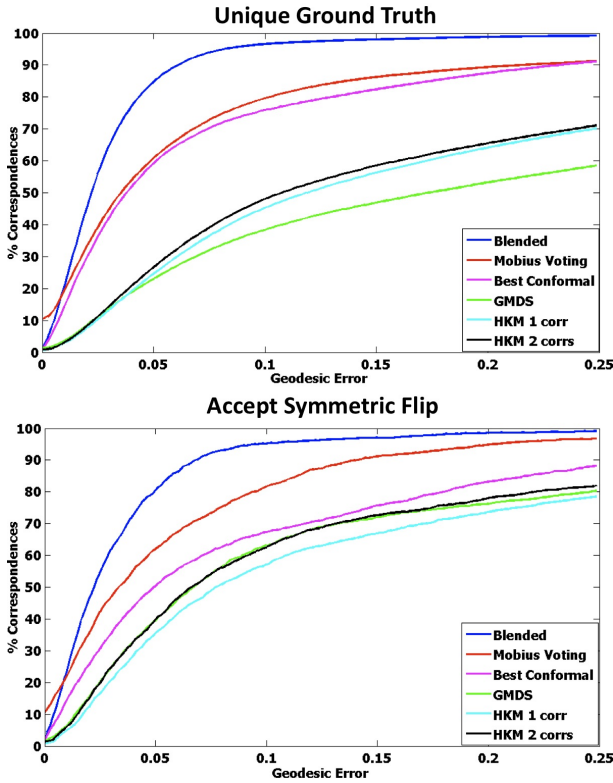
Figure 8: *TOSCA: performance of various methods on nearly-isometric human and animal models. We depict a geodesic distance on the x-axis, and a percentage of correspondences within the prescribed distance of the ground truth on y-axis.*

Figure 9: *SCAPE: humans. Most of our errors on this data set are due to confusion by $180°$ rotations that map the front of a body to the back.*

with non-zero genus), totaling 132 meshes. We used each method to find a map from each model to another selected at random from this set (excluding near-isometric examples from previous experiments) and plotted results in Figure 10. As in the experiment with SCAPE models, results mainly suffer from symmetric flips. Note that while our method performed just as well as in the isometric case, Möbius Voting is suffering from inconsistent correspondences, because for large deformations locally accumulated votes have more noise. Observe also that if one is allowed to invert surface orientation, HKM with 2 correspondences and GMDS (black and green curves) outperform Möbius Voting because they produce more consistent results.

Non-isometric animals In another comparison, we found a map from every animal model in TOSCA and Watertight data sets to a random animal model (excluding near-isometric pairs). Thus, we used 31 TOSCA models and 20 Watertight models to produce maps between 51 pairs using each inter-surface mapping method. Note in Figure 11, that Möbius Voting performed significantly better than for non-isometric humans since there is no intrinsic near-symmetry in animals (the left-to-right symmetry requires inverting orientation of the surface, which is impossible with conformal maps). However, our method still out-performs Möbius Voting on average, while providing a consistent map (avoiding outlier correspondences, which can appear with Möbius Voting). Since limbs and outliers usually cover only a small part of the surface area, these errors do not contribute largely to the fraction of correspondences. However the difference becomes more obvious if we look at maximal per mesh errors, whose average is presented in Table 1. Our method performs more uniformly across different experiments and produces smaller maximal per map errors.
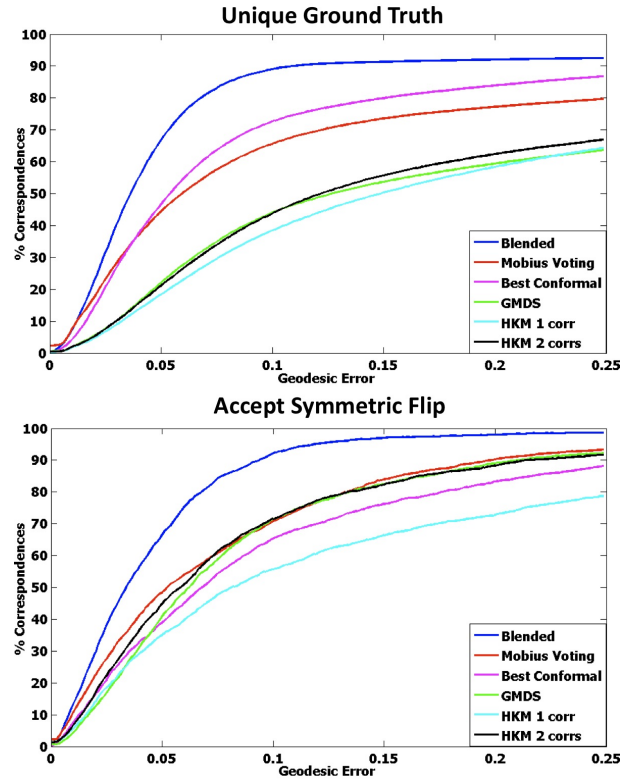
Visualization We also present visualizations of maps produced by each method for two examples in Figure 13. Please refer to the caption of that figure for more details.

8.4 Per-class performance

Finally, we investigated performance of our algorithm for a variety of object classes in the SHREC Watertight 2007 data set, where models vary in semantic content, resolution, and tessellation. As in other experiments, we mapped every model in each considered class to a random model within the same class, yielding 218 blended maps. In Figure 12, we observe that our method gives the best performance for articulated figures, but it also successfully handles many cases from other classes (such as planes, fish or birds). Some examples, as well as discussion of results can be found in Figure 14.

8.5 Timing

We assume that our method is used in a non-interactive manner for a large collection of surfaces that require pairwise maps. Thus, we want our method to find the best maps as quickly as possible with no user intervention. The running time of our method depends on two factors: the number of vertices in the mesh $|\mathcal{M}|$ and the number of extracted feature points $|\mathcal{P}|$. In a pre-processing stage, we find feature points and precompute geodesic distances for them which requires about 30s for a Scape model with 12500 vertices, 60s for a TOSCA model with 27894 vertices (cat), and 170s for a TOSCA model with 52565 vertices (David) on 2.2GHz Opteron 275 processor. The remaining part of the algorithm mainly depends on the number of selected feature points $|\mathcal{P}|$. For a pair of SCAPE models usually $|\mathcal{P}_{\mathcal{M}_1}| = |\mathcal{P}_{\mathcal{M}_2}| = 5$, and our algorithm takes 50s, of which 24s spent on calculating per map confidences, 22s spent on
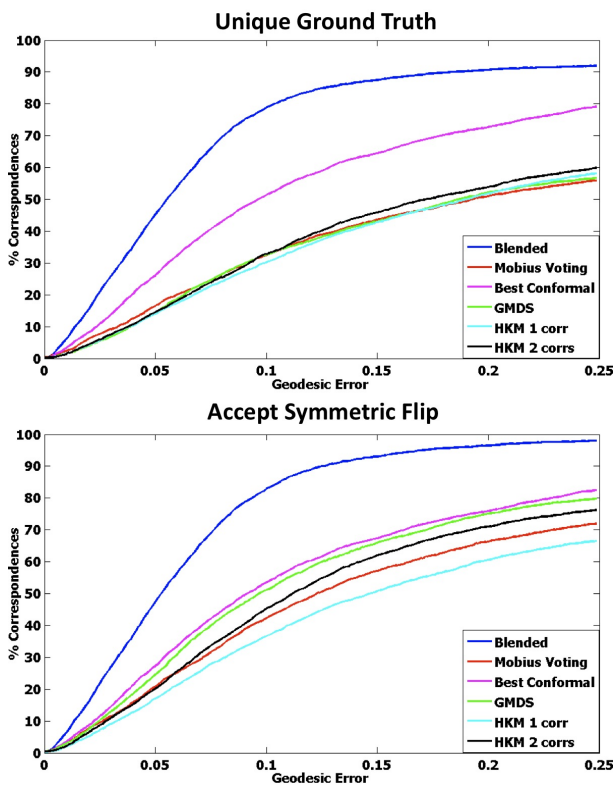
Figure 10: *Humans: performance of various methods on non-isometric pairs of humans. Our method outperforms other approaches because it extracts two consistent solutions (due to intrinsic symmetry) and then uses local cues (integrated confidence) to choose the best map.*

Figure 11: *Animals: performance of various methods on non-isometric pairs from TOSCA and SHREC Watertight'07 data sets. Although the performance of Möbius Voting is more similar to ours in this test, it suffers from inconsistent outliers as seen in Table 1, which shows average of maximal per mesh errors.*

filling the weight matrix, and 4s is spent on processing eigenvectors and finding correspondences for every vertex on a surface. For a pair of cats from TOSCA data set with $|\mathcal{P}_{\mathcal{M}_1}| = |\mathcal{P}_{\mathcal{M}_2}| = 6$, confidence calculations take 86s, and the matrix is filled in 212s. In the hardest cases (e.g., ants, centaur), we randomly remove maps until we are left with 10,000, thus in the slowest case confidences are calculated in 364s, and the matrix is filled in 1411s. A more intelligent pruning of the blending matrix is required to obtain better results for more complex shapes with larger number of relevant feature points.

9 Conclusion and Future Work

We have developed a method for finding a map between surfaces by blending a collection of low dimensional maps. Our approach is fully automatic, it outperforms several state of the art methods, and it provides mapping between non-isometric surfaces.

In the future, we hope to investigate blending of non-conformal maps and how the performance of our approach is affected by this choice. We believe that combining multiple types of low-dimensional maps might improve results, since different types of maps may be better for different parts of a surface. A limitation of the current approach is that it is able to find only global mappings, and hence is not guaranteed to work in cases of partial near-isometric matching. Furthermore, we have currently implemented uniformization only for genus zero surfaces, leaving generalization to higher genus to future work. Finally, observing that blended maps define a low-dimensional space of maps between surfaces, we think it is interesting to investigate other ways to create flexible maps with few degrees of freedom with the goal of understanding
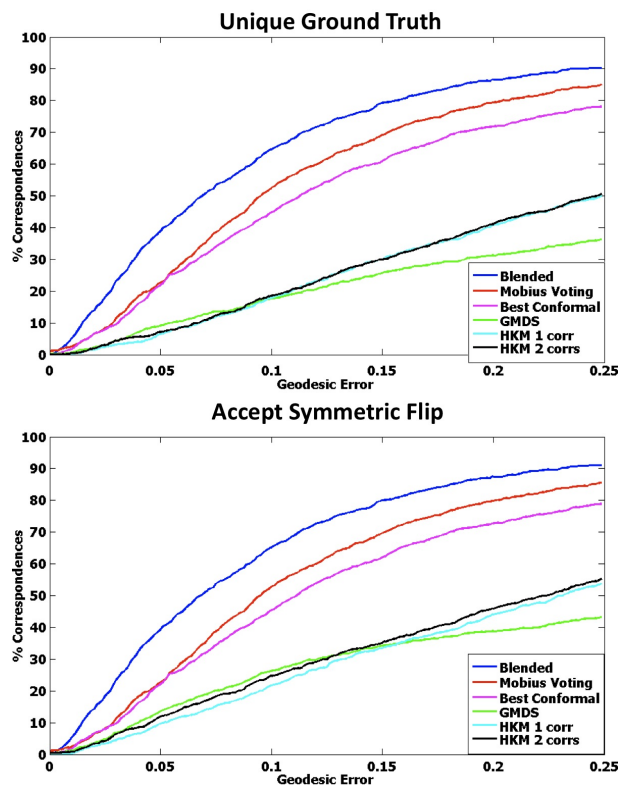
the limits of polynomial time algorithms for mapping non-isometric surfaces.

10 Acknowledgments

References

ALEXA, M. 2001. Recent advances in mesh morphing. *Computer Graphics Forum 21*, 2, 173–198.

ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. 2003. The space of all body shapes: reconstruction and parameterization from range scans. *ACM Transactions on Graphics (proc. SIGGRAPH)*.

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., PANG, H., AND DAVIS, J. 2004. The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces. *Proc. of the Neural Information Processing Systems*.

	TOSCA	SCAPE	Humans	Animals
Blended	0.31	0.25	0.32	0.33
Möbius Voting	0.45	0.64	0.97	0.57
Best Conformal	0.54	0.60	0.69	0.56
GMDS	0.97	0.86	1.11	1.02
HKM 1 corr	1.27	1.29	1.39	1.08
HKM 2 corrs	1.21	1.07	1.33	1.06

Table 1: *Table of averaged maximal per map geodesic errors. Note that our method has smaller average maximal error compared to all other methods. Even in the experiment with non-isometric animals, where Möbius Voting gives a comparable performance in terms of fraction of accepted correspondences, we give substantially smaller maximal per mesh error: first, because we better map limbs, and second, because we generate consistent map that does not have outlier correspondences.*
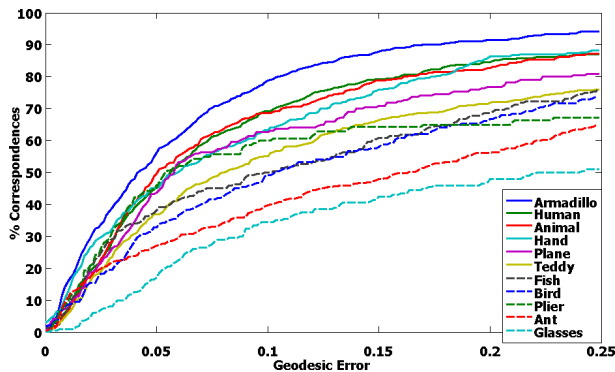

Figure 12: *Per class performance of our method on models from SHREC Watertight'07 data sets. Our method currently performs best for articulated figures with small numbers of feature points.*

BESL, P., AND MCKAY, N. 1992. A method for registration of 3-d shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*.

BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2006. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. National Academy of Sciences (PNAS)*.

BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2008. *Numerical geometry of non-rigid shapes*. Springer.

BROWN, B. J., AND RUSINKIEWICZ, S. 2007. Global non-rigid alignment of 3-d scans. *ACM Transactions on Graphics (Proc. SIGGRAPH)*.

CHANG, W., LI, H., MITRA, N., PAULY, M., AND WAND, M. 2010. Geometric registration for deformable shapes. *Eurographics 2010 course*.

ELDAR, Y., LINDENBAUM, M., PORAT, M., AND ZEEVI, Y. 1997. The farthest point strategy for progressive image sampling. mover's distance as a metric for image retrieval. *Int. J. Comput. Vision 40*, 2, 99–121.

FUNKHOUSER, T., AND SHILANE, P. 2006. Partial matching of 3d shapes with priority-driven search. *Symp. on Geom. Processing*.

GELFAND, N., MITRA, N. J., GUIBAS, L., AND POTTMANN, H. 2005. Robust global registration. *Symp. on Geom. Processing*.

GHOSH, D., SHARF, A., AND AMENTA, N. 2009. Feature-driven deformation for dense correspondence. *Proc. SPIE 7261*.

GIORGI, D., BIASOTTI, S., AND PARABOSCHI, L. 2007. Shrec:shape retrieval contest: Watertight models track. *http://watertight.ge.imati.cnr.it/*.

GOLOVINSKIY, A., AND FUNKHOUSER, T. 2009. Consistent segmentation of 3D models. *Computers and Graphics (Shape Modeling International 09) 33*, 3 (June), 262–269.

HUANG, Q., ADAMS, B., WICKE, M., , AND GUIBAS, L. J. 2008. Non-rigid registration under isometric deformations. *Computer Graphics Forum (Proc. SGP 2008)*.

KIM, V. G., LIPMAN, Y., CHEN, X., AND FUNKHOUSER, T. 2010. Mobius transformations for global intrinsic symmetry analysis. *Computer Graphics Forum (Proc. of SGP)*.

KRAEVOY, V., AND SHEFFER, A. 2004. Cross-parameterization and compatible remeshing of 3d models. *ACM Transactions on Graphics (Proc. SIGGRAPH 2004)*.

LI, H., SUMNER, R. W., AND PAULY, M. 2008. Global correspondence optimization for non-rigid registration of depth scans. *Computer Graphics Forum (Proc. SGP'08)*.

LIPMAN, Y., AND FUNKHOUSER, T. 2009. Mobius voting for surface correspondence. *ACM Transactions on Graphics (Proc. SIGGRAPH) 28*, 3 (Aug.).

MÉMOLI, F., AND SAPIRO, G. 2004. Comparing point clouds. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*.

OVSJANIKOV, M., MÉRIGOT, Q., MÉMOLI, F., AND GUIBAS, L. 2010. One point isometric matching with the heat kernel. In *Computer Graphics Forum (Proc. of SGP)*.

PAULY, M., MITRA, N. J., GIESEN, J., GROSS, M., AND GUIBAS, L. 2005. Example-based 3d scan completion. In *Symposium on Geometry Processing*, 23–32.

PINKALL, U., AND POLTHIER, K. 1993. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2*, 15–36.

PRAUN, E., SWELDENS, W., AND SCHRÖDER, P. 2001. Consistent mesh parameterizations. *Proc. of SIGGRAPH 2001*.

SCHREINER, J., ASIRVATHAM, A., PRAUN, E., AND HOPPE, H. 2004. Inter-surface mapping. *ACM Transactions on Graphics (Proc. SIGGRAPH)*.

TEVS, A., BOKELOH, M., M.WAND, SCHILLING, A., AND SEIDEL, H.-P. 2009. Isometric registration of ambiguous and partial data. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition*.

VAN KAICK, O., ZHANG, H., HAMARNEH, G., AND COHEN-OR, D. 2010. A survey on shape correspondence. *Eurographics State-of-the-Art report*.

ZHANG, H., SHEFFER, A., COHEN-OR, D., ZHOU, Q., VAN KAICK, O., AND TAGLIASACCHI, A. 2008. Deformation-driven shape correspondence. *Computer Graphics Forum (Proc. of SGP)*.
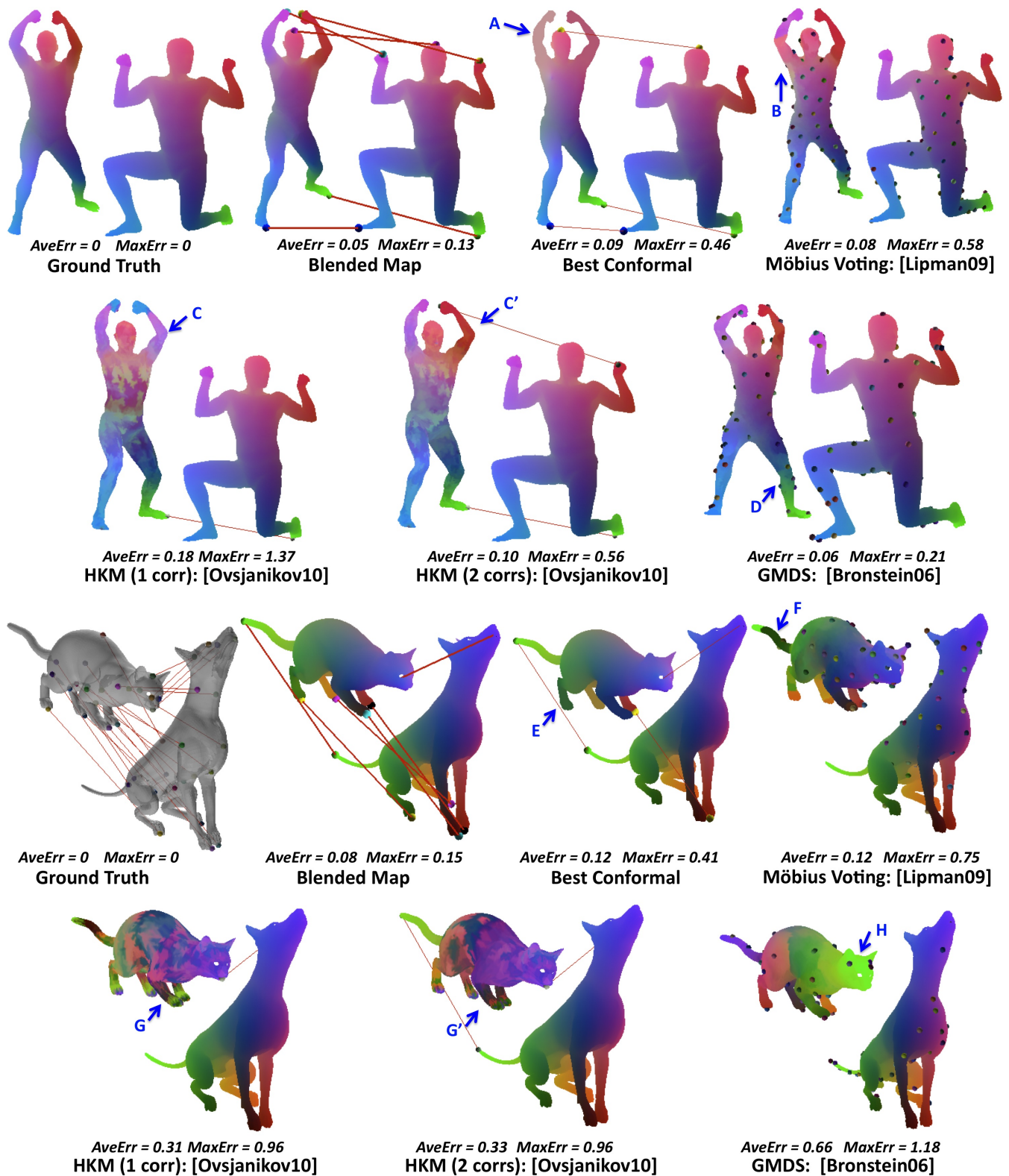
Figure 13: *This figure shows a comparison of our method (Blended Map) to others. In the top row we see a map between two humans. A common problem a single conformal map is that it collapses one of extremities to a small area, as seen in areas (A) and (E). Image area (B) reveals an inconsistent correspondence assigned by Möbius Voting on a shoulder. Areas (C) and (C') show that adding a second correspondence improves results for some examples, in this case a left hand was correctly mapped with two heat kernel maps. Note however, that maps produced by searching for nearest neighbors in some space usually do not produce continuous maps (HKM and GMDS (D) methods). For stronger deformations, for example mapping a cat to a dog, it is challenging to even find a general structure of a map. For example, in cases labeled (F), (G), (H) coarse correspondences map at least some limbs (or a whole body) incorrectly. Note that for such strong deformations a single heat kernel map is not sufficient to map the whole body, augmenting a second correspondence improves the result for tail, but other limbs are still mapped incorrectly (G), (G').*
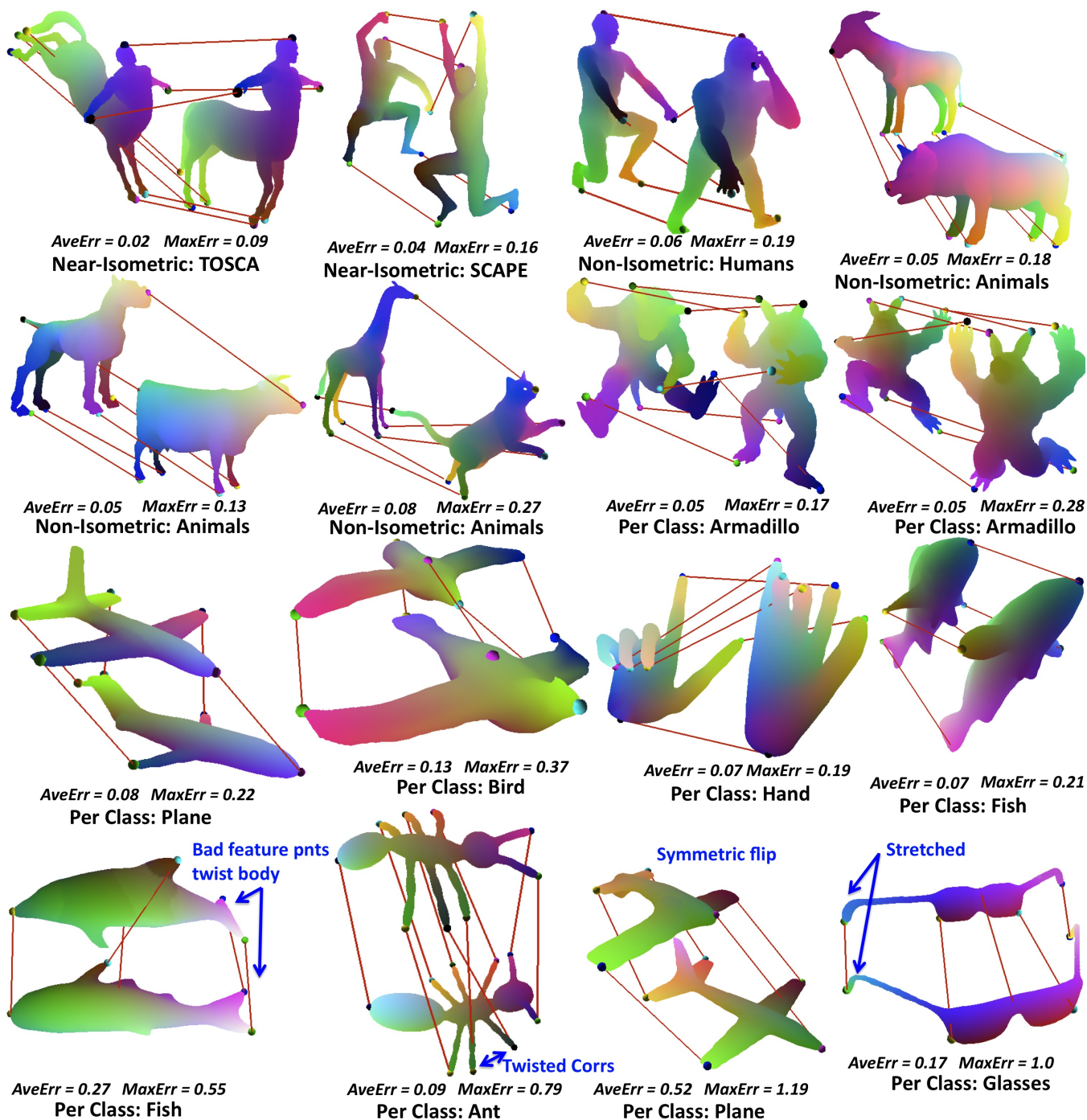
Figure 14: *Performance of our method on various classes. The first three rows of this plot include success cases for our method, and the bottom row depicts four typical failure cases. Note that our method successfully handles highly non-isometric pairs. Observe that our method does not rely on local geometry cues, and thus succeeds in a map even if local geometry of surfaces is very different, for example, there is no local resemblance of a shape of giraffe and a shape of a cat, or shape of a dog and a cow. The fact that conformal maps do not preserve geodesic distances can also serve as an advantage in case of partial scaling. For example, in the rightmost armadillo example one of the models is missing part of a hand, but associating a full hand with a stump does not propagate any errors to the rest of the shape. You can observe the most common failures of our methods in the bottom row. We believe the most common problem is misalignment in feature points. For example, fish and dolphin in the first image have different orientation of a tail, thus conformal maps generated by feature points on a tail twist map on the body. Another common issue is symmetric flip as depicted for ant and airplane examples. Note that the the only hint for our method on deciding which is the correct solution is a small asymmetry in location of ant's limbs and orientation of plane's tail. This in many cases is too subtle to be captured by our integrated confidence value. Note also that for creatures with more than 7 feature points we prune the blending matrix to bound processing time. This introduces an undesired bias, which can (as in the example with the ant) result in incorrect eigenvectors; for example, two limbs of the ant are twisted. Finally, individual conformal maps can also be low quality if objects have long, thin parts. Specifically, map for glasses introduces a lot of distortion of tips of a frame which is not distributed uniformly.*