

An Optimization Approach to Improving Collections of Shape Maps

Andy Nguyen¹ Mirela Ben-Chen¹ Katarzyna Welnicka² Yinyu Ye¹ Leonidas Guibas¹

¹Stanford University

²Technical University of Denmark

Abstract

Finding an informative, structure-preserving map between two shapes has been a long-standing problem in geometry processing, involving a variety of solution approaches and applications. However, in many cases, we are given not only two related shapes, but a collection of them, and considering each pairwise map independently does not take full advantage of all existing information. For example, a notorious problem with computing shape maps is the ambiguity introduced by the symmetry problem — for two similar shapes which have reflectional symmetry there exist two maps which are equally favorable, and no intrinsic mapping algorithm can distinguish between them based on these two shapes alone. Another prominent issue with shape mapping algorithms is their relative sensitivity to how “similar” two shapes are — good maps are much easier to obtain when shapes are very similar. Given the context of additional shape maps connecting our collection, we propose to add the constraint of global map consistency, requiring that any composition of maps between two shapes should be independent of the path chosen in the network. This requirement can help us choose among the equally good symmetric alternatives, or help us replace a “bad” pairwise map with the composition of a few “good” maps between shapes that in some sense interpolate the original ones. We show how, given a collection of pairwise shape maps, to define an optimization problem whose output is a set of alternative maps, compositions of those given, which are consistent, and individually at times much better than the original. Our method is general, and can work on any collection of shapes, as long as a seed set of good pairwise maps is provided. We demonstrate the effectiveness of our method for improving maps generated by state-of-the-art mapping methods on various shape databases.

Categories and Subject Descriptors (according to ACM CCS):

1. Introduction

Shapes often do not appear in isolation, but rather in collections or families of related shapes, whether it is humans, animals, fish, furniture, vehicles, etc. Such collections provide a context for each individual shape, and thus can potentially allow for higher quality analysis and processing of each object than would have been possible if the particular shape was treated in isolation. One application which can benefit from the context provided by a collection is the important task of generating an informative and structure-preserving map between a pair of shapes. This problem, for a pair of shapes in isolation, has been well studied for a variety of classes of shapes and of criteria for map quality. A general technique involves defining some measure of distortion of a map, and then computing the map that minimizes this distortion for a given pair of shapes. For example, for 3D shapes,

a common measure of distortion is the difference between the pairwise geodesic distances between points on the source shape, and the distance of their images under the map, aggregated over all pairs of points using a standard norm.

Such optimal “as isometric as possible” mappings relate to the classical problem of the Gromov-Hausdorff distance [Gro07] of shapes viewed as metric spaces and computationally are special instances of the quadratic assignment problem [Çel98], which in general is known to be NP-hard. Besides the computational difficulty of the problem, however, there are deeper reasons in the shape context for which such optimizations might not produce the maps that carry the semantic structure we want. We argue below that the context provided by other related shapes in a collection can often help overcome these difficulties.

One common difficulty is caused by symmetric shapes.

Consider, for example, two poses of a human shape. Two equally good maps exist between these shapes, which preserve geodesic distances equally well. One is the “straight” map, mapping a right hand to a right hand and so on, and the other is the symmetric map, taking each point to its symmetric point under the left-right reflection symmetry of the human body — mapping the left hand to the right hand.

By considering only two shapes and the “as isometric as possible” criterion, it is usually not possible to distinguish between these two maps and determine which one is “correct”, regardless of the choice of algorithm used for computing the map. For the specific case of reflectional symmetry, extrinsic properties such as normals may be used to distinguish symmetric maps from “straight” maps; however, many more types and degrees of symmetries exist (e.g., a starfish) which can foil such methods.

In some cases, the notion of “correctness” or “naturalness” of a map is actually a property of the underlying collection of shapes. That is, the “correctness” of a specific map between a given pair of shapes can vary depending on what other shapes exist in the collection (Fig. 1).

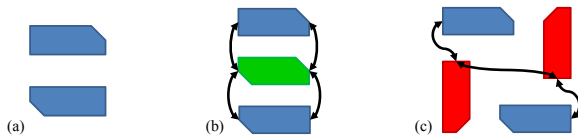


Figure 1: (a) Two shapes that admit multiple “correct” maps. (b) The third green shape indicates the “correct” map is translation. (c) The two new red shapes indicate the “correct” map is rotation.

Another common problem with pairwise mapping algorithms, is that they tend to perform well on shapes for which a low-distortion map exists, e.g. shapes which are almost-isometric deformations of each other. However, they are less successful in cases where the best map has a larger distortion. Consider for example various poses of a character, some with bigger distortions than others. Maps between poses which are “closer”, in the sense that the distortion of the geodesic distances is small, are well behaved, whereas a large distortion in geodesic distances might cause undesirable mismatches in the map.

In many cases, for a given collection of shapes, a number of maps computed using an optimization criterion may be “good”, whereas a number of others may be “bad”. For example, in a collection of male shapes and female shapes in various poses, we may have good maps within each subgroup, and perhaps good maps between a male and a female in a neutral position, but otherwise bad maps between males and females. Or, in a morph sequence between two shapes, we may get a good map between every two consecutive frames, but bad maps between more distant frames.

One reason bad maps can arise is that no simple geometric optimization criterion truly captures the semantic transfer of information that we want our maps to attain. The semantics may depend on hidden attributes of the objects beyond their surface geometry, not directly available to us — such as internal structure, functional use, etc. Such secondary hidden attributes, like missing coordinates, might affect correspondence-finding only a little when we have data sets that are very close to each other, but may make a significant difference when the sets are far removed (in terms of some shape space distance measure).

In all the cases described above, one can leverage the context provided by the collection to improve the “bad” maps. As mentioned previously, by considering only one map in isolation, we often cannot make an informed decision on whether the map is “good”, since we do not have the ground truth available. However, given the collection we can compute a few possible maps between every two shapes: the direct maps provided to us, and others that are compositions through other shapes. For example, given three shapes S_1, S_2, S_3 , we have the original map $m_{1,3} : S_1 \rightarrow S_3$, but we also have the composed map $m_{1,2,3} = m_{2,3} \circ m_{1,2}$ taking S_1 to S_2 and S_2 to S_3 . If the collection of maps is all good we would expect global consistency, namely that $m_{1,3}$ and $m_{1,2,3}$ should be the same, or at least similar. Hence, we can propose to define a criterion for the “goodness” of a collection of maps, as the *consistency* of the collection, namely how similar are the maps achieved by composing maps between two shapes along different paths. An equivalent way of measuring the same criterion, assuming the maps $m_{i,j}$ and $m_{j,i}$ are 1-1 and onto (map invertibility), is to check how close a map is along each “cycle” (e.g. $m_{1,2,3,1}$) to the identity map.

This criterion has a few advantages. First, it is very general, and does not require knowledge of the specific shapes, or the maps in any specific form. All that is needed is a way to compose maps, and a way to evaluate how close a map is to the identity — thus this criterion can be applied to a variety of shapes, and mapping algorithms. Second, it fits nicely with the scenarios described above. Under some conditions, it will correctly identify a collection of maps including (for example) a symmetric map, as inconsistent and thus “bad”. Finally, and most importantly, it provides a constructive way of *correcting* the maps, as we can replace a map $m_{i,j} : S_i \rightarrow S_j$ with the map $m_{i,\dots,j}$ if it improves the consistency of the collection.

We propose an optimization method for estimating the consistency of a given collection of maps, for identifying the “bad” maps, and for improving the consistency by replacing certain maps by compositions of other maps. The method requires that there is a sufficient set of “good” maps in the collection to at least connect all or most of the shapes. Our algorithm is based on trying to estimate the “badness” of individual maps by the amount of inconsistency they introduce in the collection, and then improving them by replac-

ing them with a composition of other maps that introduces less inconsistency. We demonstrate how our method can improve on the results generated by state-of-the-art mapping methods such as the heat kernel map [OMMG10], Möbius voting [LF09], and blended conformal maps [KLF11] on a variety of shape collections. Furthermore, we show that the paths between shapes that result in the optimal maps can in some cases provide some information on the underlying semantic structure of the collection based on phenotypic information alone.

2. Related Work

There have been numerous prior papers that build similarity graphs between shapes as tools for clustering and aggregating shapes or understanding geodesics in a shape space. For example, dimension reduction of a shape space can be accomplished by non-linear dimensionality reduction approaches such as isomap [Tdl00], which depend on connecting each shape to its “near-neighbors” [YK06]. We also start from a network of shapes connected by edges whenever we have a map available between the two shapes. The important novelty in our setting is that the graph edges have maps associated with them. These maps can be composed and possess an algebraic structure that we deeply exploit. There has been little prior work related to improving pairwise maps based on the context provided by a collection of related shapes. Some recent papers address variations of this problem in the context of image analysis.

Our approach is most similar to the work of Zach et al. [ZKP10], whose aim is to identify false correspondences in a collection of images. They define a graph whose vertices are images, or portions of images, and whose edges imply some relationship between the regions, such as an affine correspondence between feature sets. If there are repetitions in the image (e.g., a building with a regular window pattern), some of these correspondences may be wrongly identified. They propose to prune the false correspondences using an optimization approach based on the consistency of cycles in this graph, assigning each edge a binary weight which will indicate whether or not it is a false correspondence and then using loopy belief propagation. Our approach is similar to theirs in the sense that we also use cycles for identifying inconsistencies. We, however, have a more general setup, which allows us not only to remove erroneous edges, but also to choose among better map compositions. In addition, we use the weights found in this process to identify some underlying structure of the collection of shapes.

Roberts et al. [RSS11] extend in some directions the work of Zach et al. by showing how to overcome the case where the number of erroneous correspondences is much larger than the number of correct correspondences due to appearance of large repeating structures in the image. Although such a case can also appear in our scenario, e.g. if

we have many non-similar shapes with no good maps, their proposed solution is not easily transferred to 3D shapes.

Charpiat [Cha09] uses the idea of composing maps between close shapes to improve the maps between non-similar shapes. Similarly to our approach, he builds a graph where each shape is a vertex, and edges between shapes are weighted by the cost of the best map between them. He then replaces a pairwise map between S_i and S_j by the composition of the maps on the shortest (best) path between i and j . However, the resulting compositions are solely based on the weights of the pairwise maps, which do not include knowledge about the collection (e.g., the quality of map compositions). We, on the other hand, learn the weights of the pairwise maps from the inconsistency the maps introduce to the collection and thus our method is more resilient to cases where equally good maps exist but are inconsistent.

The work on 3D shapes that is most closely related to our approach is on methods for joint segmentation of almost-isometric deformations of shapes, such as [dGGV08, Reu10], to mention just two recent works. These methods, however, use the structure of some isometry invariant operator on the surface, such as the Laplace-Beltrami operator, to achieve simultaneous correspondences, and do not take into account the consistency of the collection of maps.

The concept of consistency along cycles has been also addressed in the area of vector field processing (see e.g. [LJX*10]), where it is more commonly known as *holonomy*. In that scenario, it is well-known that given a vector field on a surface, in most cases consistency along cycles cannot be achieved everywhere, and singularities must arise. It is possible that the two approaches are related, if one considers vector fields on a full shape space manifold. In that case, it might also be the case that global consistency cannot be achieved. However, since we only have a very sparse and in general localized sampling of the shape space, we believe this problem will not arise in practice. We leave the investigation of the relationship between our approach and the holonomy of vector fields on the shape space manifold for future work.

Many pairwise correspondence methods have been proposed in recent years, and it is beyond the scope of our paper to review them all. Our method builds on top of these and is quite general — any such correspondence method, whether it is applied to 2D contours or 3D shapes, can be used as a building block for constructing the collection of pairwise maps. In our experiments we chose to use dynamic time warping [RRL78] for matching 2D contours, and Möbius voting [LF09], the heat kernel map [OMMG10], and blended conformal maps [KLF11] for matching 3D shapes, although any other choice of algorithms would be equally valid.

At a more abstract level, our approach to shapes through mappings relates to the key mathematical idea of *functoriality*, which has a long tradition associated with it. Functoriality in its broadest form is the notion that, in dealing with any

kind of mathematical object, it is at least as important to understand the transformations or symmetries possessed by the object or the family of objects to which it belongs, as it is to study the object itself. Indeed, the study of the transformations is frequently the most powerful tool for analyzing an object family. Instances of this program include Galois Theory, Felix Klein's Erlanger program for geometry, and the development of homological algebra and category theory.

3. Problem Statement

We are given as input a collection of n related shapes $\mathcal{S} = \{S_1, S_2, \dots, S_n\}$, along with a collection of maps between pairs of shapes $\mathcal{M} = \{m_{i,j}, m_{j,i} : S_i \rightarrow S_j\}$. For simplicity we will assume that the maps are invertible $m_{i,j} = m_{j,i}^{-1}$, that for each pair of shapes there is a corresponding map, and that the only self maps $m_{i,i}$ are the identity maps. Our goal is to produce an alternative set of maps between every pair $\hat{m}_{i,j} : S_i \rightarrow S_j$, which are "better" than the original maps. To be able to evaluate the quality of the maps, we assume we are given a metric $d_S : S \times S \rightarrow \mathbb{R}$ for any shape S , which is commensurable between the different shapes. For example, for a 3D shape, $d_S(p, q)$ can be the geodesic distance between p and q normalized by the diameter of S . To construct the new maps, we consider composition of maps as paths on the graph $G_{\mathcal{M}} = (\mathcal{S}, E)$, such that $e_{i,j} = (S_i, S_j) \in E$ if $m_{i,j} \in \mathcal{M}$. Any path $\gamma = \{i_1, i_2, \dots, i_k\}$ in G induces a map $m_\gamma : S_{i_1} \rightarrow S_{i_k}$ given by the composition of the pairwise maps on the edges of the path: $m_\gamma = m_{i_{k-1}, i_k} \circ \dots \circ m_{i_2, i_3} \circ m_{i_1, i_2}$.

We want maps $\hat{m}_{i,j}$ which are better than the original maps, in the sense that they are closer to some (possibly unknown) ground-truth maps. Assuming that for every pair of shapes S_i, S_j there exists a single ground-truth map $\tilde{m}_{i,j}$, we can define the accuracy of the map collection \mathcal{M} as follows.

Definition 1 The *accuracy error* of the collection of maps \mathcal{M} over the collection of shapes \mathcal{S} is given by:

$$E_{acc}(\mathcal{M}) = \frac{1}{|\mathcal{S}|^2} \sum_{A,B \in \mathcal{S}} E_{acc}(m_{A,B}),$$

$$E_{acc}(m_{A,B}) = \frac{1}{|A|} \sum_{p \in A} d_B(m_{A,B}(p), \tilde{m}_{A,B}(p)),$$

where we assume that every shape $A \in \mathcal{S}$ is given as a discrete set of points.

While this definition of accuracy error of a map is not symmetric in general, if the ground-truth maps preserve the distortion measure d_S up to some $\varepsilon \geq 0$, so that $\forall A, B \in \mathcal{S}$

$$|d_A(p, q) - d_B(\tilde{m}_{A,B}(p), \tilde{m}_{A,B}(q))| \leq \varepsilon,$$

then the accuracy error on a map is nearly symmetric:

$$|E_{acc}(m_{A,B}) - E_{acc}(m_{B,A})| \leq \varepsilon.$$

Using these definitions we can specify our problem as finding paths which will yield the most accurate maps.

Problem 1 Given a collection of shapes \mathcal{S} and a collection of maps \mathcal{M} , we seek paths γ_{ij} from S_i to S_j for all $i, j \in \{1, \dots, n\}$, such that for the collection of maps $\hat{\mathcal{M}} = \{\hat{m}_{i,j} = m_{\gamma_{ij}}\}$ we have that $E_{acc}(\hat{\mathcal{M}})$ is minimal.

Of course, since the ground-truth maps are unknown, this problem cannot be solved in this form. Instead, we devise an algorithm to improve the maps by improving the *consistency* of the collection. Our main claim is that maps which degrade the consistency of the collection are in many cases also far from the ground-truth. Our main focus is, therefore, identifying such maps and replacing them, thus improving both the collection's consistency and the maps' accuracy.

3.1. Consistency

For our maps to be consistent we require that different paths between two shapes yield similar maps, or equivalently (under the assumption that the maps are invertible) that cycles yield maps similar to the identity. Another way to express this is to say that we want our map diagrams to be as commutative as possible, if we think of them as morphisms in the sense of category theory. Hence, we define the consistency error as follows.

Definition 2 The *consistency error* of the collection of maps \mathcal{M} over the collection of shapes \mathcal{S} is given by:

$$E_{cons}(\mathcal{M}) = \frac{1}{|\mathcal{S}|} \sum_{A \in \mathcal{S}} \frac{1}{|C_G(A)|} \sum_{\gamma \in C_G(A)} E_{cons}(\gamma),$$

$$E_{cons}(\gamma) = \frac{1}{|A|} \sum_{p \in A} d_A(p, m_\gamma(p)),$$

where $C_G(A)$ is the collection of cycles in G which start at A . Note that in general a cycle $\gamma_1 = \{i_1, i_2, \dots, i_n, i_1\}$ induces a map $m_{\gamma_1} : S_{i_1} \rightarrow S_{i_1}$, whereas the cycle $\gamma_2 = \{i_2, \dots, i_n, i_1, i_2\}$ induces a map $m_{\gamma_2} : S_{i_2} \rightarrow S_{i_2}$. Therefore, when discussing cycles in G we will always mention their starting shape.

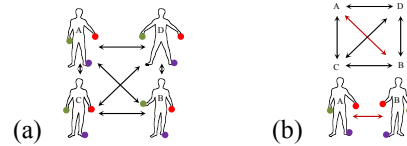


Figure 2: A collection of shapes with (a) consistent and (b) inconsistent maps. In (b), the collection includes the same maps as (a) except for $m_{A,B}$, which introduces a symmetry.

Figure 2 demonstrates the idea of consistency on a simple collection of shapes. In (a) we show a consistent collection, visualized by showing a small number of points which are in correspondence across all pairs of shapes. In (b) we have replaced a single map $m_{A,B}$ with the left-right symmetric map, resulting in an inconsistent collection, since, for example, the composed map over the cycle (A,B,C,A) will not yield the identity map.

The advantage of the consistency criterion is that we can measure it directly, without requiring the (usually unknown) ground-truth maps. The important question is how consistency can inform us about accuracy. Consider, for example, the collection in Figure 2(b). It is easy to check that the cycles (A, B, C, A) and (A, B, D, A) are not consistent, whereas the cycles (B, C, D, B) and (A, C, D, A) are consistent. Furthermore, the inaccurate map $m_{A,B}$ appears only in the inconsistent cycles. In this case then, the consistency of the collection allows us to identify the inaccurate map as the only map which appears only in inconsistent cycles. We therefore want to derive a connection between the consistency of a path and the accuracy of the maps along its edges. To this end, we will require the following lemma, which relates the accuracy of a path and the accuracy of its edges.

Lemma 1 If the ground-truth maps preserve the distortion measure d_S up to some $\varepsilon \geq 0$, then the accuracy of a path $\gamma = \{i_1, \dots, i_n\}$ is bounded by the accuracy of its maps:

$$E_{acc}(\gamma) \leq \sum_{j=1}^{n-1} E_{acc}(m_{i_j, i_{j+1}}) + (n-2)\varepsilon,$$

where the accuracy of a path is defined as:

$$E_{acc}(\gamma) = \frac{1}{|A|} \sum_{p \in A} d_B(m_\gamma(p), \tilde{m}_\gamma(p)), \quad A = S_{i_1}, B = S_{i_n}.$$

The proof is quite straightforward, and is based on the fact that since the maps are invertible, we can compute all the accuracy errors in the sum in terms of points on the first shape S_{i_1} . Furthermore, since d_S is preserved under the ground-truth maps, we can measure all these errors on the last shape S_{i_n} . The result follows from the fact that d_S is a metric and thus fulfills the triangle inequality. Note that there are $n-1$ maps and the first map does not contribute an ε to the accuracy error, hence the additional error of at most $(n-2)\varepsilon$.

If we further assume that the ground-truth maps are exactly consistent, namely $\tilde{m}_\gamma(p) = p$ for any cycle γ , then $E_{acc}(\gamma) = E_{cons}(\gamma)$, and we get the following connection:

Corollary 1 If $E_{cons}(\tilde{\mathcal{M}}) = 0$, then for any n -cycle $\gamma = \{i_1, \dots, i_n, i_1\}$, we have that

$$E_{cons}(\gamma) \leq \sum_{(i,j) \in \gamma} E_{acc}(m_{i,j}) + (n-1)\varepsilon. \quad (1)$$

In particular, if all the edges in a cycle are accurate, then the cycle must be consistent. Similarly, if all the edges in a cycle are exactly accurate except one, we may link the inaccuracy of that edge to the inconsistency of the cycle:

Corollary 2 If $E_{cons}(\tilde{\mathcal{M}}) = 0$, then for any cycle $\gamma = \{i_1, \dots, i_n, i_1\}$ for which $E_{acc}(m_{i_j, i_{j+1}}) = 0, \forall j$ except $j = k$, we have that

$$E_{cons}(\gamma) - E_{acc}(m_{i_k, i_{k+1}}) \leq (n-1)\varepsilon. \quad (2)$$

This allows us to “flush out” inaccurate maps by investigating the consistency of the cycles, as in Figure 2(b).

4. From Consistency to Accuracy

We now show how to use Corollary 1 to devise an optimization problem to identify “isolated” inaccurate maps. Then, using this optimization as a building block, we construct an algorithm which iteratively improves the collection’s accuracy by repeatedly replacing maps suspected to be inaccurate with compositions of more accurate maps.

4.1. Almost Accurate Collections

In our discussion so far, we have defined consistency in terms of *all* cycles in G . However, considering all cycles is algorithmically intractable, and so we limit ourselves to a subset of all cycles. The 2-cycles are consistent by definition (as the maps are invertible), thus we focus on the 3-cycles. We discuss later how to overcome this limitation.

Definition 3 Given a collection of maps \mathcal{M} , let $\mathcal{B}(\mathcal{M}) = \{m_{i,j} \in \mathcal{M} \mid E_{acc}(m_{i,j}) > 0\}$ — the collection of inaccurate maps. Then we say that \mathcal{M} is *almost accurate*, if there do not exist two maps $m_1, m_2 \in \mathcal{B}(\mathcal{M})$, which both belong to the same 3-cycle in $G_{\mathcal{M}}$. We call such maps *isolated*.

We now define an optimization problem whose aim is to identify the inaccurate maps by assigning a positive cost to every edge $e = (S_i, S_j)$. On the one hand, we want to minimize the total inaccuracy in the collection, and so we use a weighted sum of the costs as the energy function. Since the costs are positive, this is equivalent to a scaled l_1 norm of the vector of costs, which has the additional advantage of favoring sparser assignments, as shown by all the recent work in compressive sensing [Bar07] and sparse reconstruction [CT05]. On the other hand, we want these costs to explain the measured inaccuracy of the cycles, so we use Equation (1) as the constraints in the problem. To avoid the dependency on the starting shape, we define the cost of the cycle as the maximal consistency error for *any* starting shape.

Definition 4 The cost of a cycle $\gamma = \{i_1, \dots, i_n\}$ is given by:

$$C_\gamma = \max_j (E_{cons}(\{i_j, i_{j+1}, \dots, i_n, i_1, \dots, i_{j-1}, i_j\})),$$

and since Equation (1) holds for any starting shape, we have

$$\sum_{(i,j) \in \gamma} E_{acc}(m_{i,j}) \geq C_\gamma - (n-1)\varepsilon, \quad (3)$$

which leads to the following optimization problem:

Problem 2 Given a collection of shapes \mathcal{S} , a collection of maps \mathcal{M} , and the resulting graph $G_{\mathcal{M}} = (\mathcal{S}, E)$, we seek an assignment of costs c_e for all $e \in E$:

$$\begin{aligned} & \underset{c_e}{\text{minimize}} && \sum_{e \in E} w_e c_e \\ & \text{subject to} && \sum_{e \in \gamma} c_e \geq C_\gamma, \quad \forall \gamma \in L, \\ & && c_e \geq 0, \quad \forall e \in E, \end{aligned}$$

where L denotes the set of all 3-cycles in $G_{\mathcal{M}}$ and the objective weights w_e are given by $w_e = 1/\sum_{\gamma:e\in\gamma}C_{\gamma}$.

This is a linear program with a polynomial number of variables and constraints, so it is tractable. Note that the value of ϵ is not known and therefore cannot be incorporated into the constraints. We therefore choose slightly stronger constraints, so that equation (3) may be satisfied. The objective weights we choose make it more expensive for the LP to assign costs to edges that participate in few bad cycles, which improves its ability to identify a sparse set of bad edges. As a simple example, consider a collection of 5 shapes with all input maps being correct except for a single 4-cycle of bad maps, such that $C_{\gamma} = 1$ if γ includes at least one of these 4 maps. It is easy to check that if we choose $w_e = 1$ for all e , the resulting LP has multiple integral optima with different supports, whereas if we use w_e as defined above, the resulting LP has a unique optimum that matches the true solution.

Before discussing the properties of our optimization problem and its extensions, we would like to show how it performs on real data. We chose a small collection of shapes from the SCAPE database [ASK*05] and computed straight and symmetric pairwise maps between them using the heat kernel map [OMMG10], by giving it two initial correspondences which yield either a straight or a symmetric map. We use Euclidean distances as an inexpensive estimator of geodesic distances; the fact that the relevant measurement is the average distance over the entire shape alleviates the problems associated with this choice.

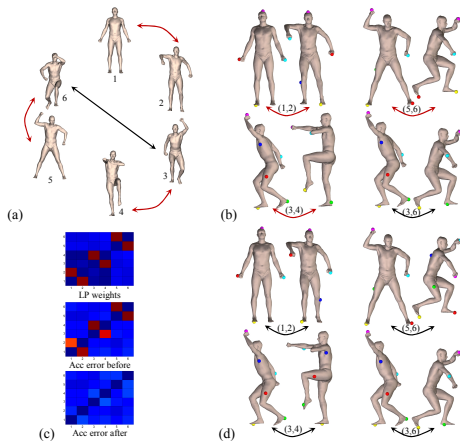


Figure 3: (a) An “almost-accurate” collection — all maps are straight except red ones. (b) 3 symmetric maps and one straight map. (c) The costs computed by the LP, accuracy errors of the original maps, and accuracy errors of the final maps. (d) Some of the final maps (compare with (b)).

We chose three of the maps to be symmetric, and the rest to be straight, so that the collection is almost accurate, in the sense that the “bad” maps are isolated. Fig. 3(a) shows

the shapes, where the symmetric maps are marked with red arrows. Fig. 3(b) shows the symmetric maps themselves, and one example of a straight map. Note, that for real data the “good” maps are never exactly accurate. The SCAPE database comes with correspondences, which we treat as the ground-truth maps to measure the accuracy of the source maps. Fig. 3(c)(center) shows the color coded accuracy errors of the original maps, and it is obvious that the symmetric maps are the inaccurate ones. In addition, we show in Fig. 3(c)(top) the color coding of the costs computed by solving Problem 2. It is evident that the optimization indeed identified correctly the inaccurate maps.

Using the computed costs we can find better maps on the shape collection. We compute the shortest paths on γ_{ij} for every two shapes S_i, S_j using the costs computed by the linear program as edge weights, and take the new maps to be $\hat{m}_{i,j} = m_{\gamma_{ij}}$. Fig. 3(c)(bottom) shows the color coded accuracies of the new maps $\hat{m}_{i,j}$, and (d) shows the output maps, for the same pairs as in (b) — all the symmetries were resolved.

In some very simple cases, the objective weights w_e alone can be sufficient to identify the inaccurate maps; however, it is easy to see that simply using these weights as a measure of quality of the maps frequently produces poor results, since too much information is lost by summing the weights of the cycles. Consider the synthetic example in Fig. 4. As we will soon show, the LP can correctly identify the three inaccurate maps (a); however, summing the weights of the cycle distortions results in the incorrect conclusion that one of the 3 inaccurate maps is the best map (b). Such an approach tends to fail because it averages out the “blame” for the inconsistency of cycles among all edges, so “good” edges become indistinguishable from “bad” edges.



Figure 4: Summing cycle weights is insufficient to distinguish accurate and inaccurate maps. (a) Accuracy errors; the LP also assigns these costs. (b) Sum of the consistency errors of the 3-cycles each edge belongs to; the “best” map is using these values is one of the 3 inaccurate maps.

In the special case of an almost accurate collection we can show under which conditions on G the solution to Problem 2 will always identify the inaccurate maps. For simplicity we will present only the proof in the case where all the ground-truth maps are isometries (that is, they are all distance-preserving, so $\epsilon = 0$), though we will discuss the bounds obtained in the more general case as well.

Theorem 1 If \mathcal{M} is almost accurate, all ground-truth maps are isometries, and $\max_{m\in\mathcal{B}(\mathcal{M})}E_{acc}(m) < ((|\mathcal{S}| - 2)^2 - 3)\min_{m\in\mathcal{B}(\mathcal{M})}E_{acc}(m)$, then the unique optimal solution to Problem 2 is the assignment $c_{(S_i,S_j)} = E_{acc}(m_{ij})$.

Proof In general, due to equation (3), the assignment $c_{(i,j)} = E_{acc}(m_{ij})$ always fulfills the constraints. The challenge is to show that any other assignment must be suboptimal. First, we note that $w_{ij} = 1/((|\mathcal{S}| - 2)E_{acc}(m_{ij}))$ for all $m_{ij} \in \mathcal{B}(\mathcal{M})$, and so the assignment $c_{(i,j)} = E_{acc}(m_{ij})$ achieves an objective value of $|\mathcal{B}(\mathcal{M})|/(|\mathcal{S}| - 2)$. Now assume there exists an alternative solution to the optimization problem \hat{c}_e . Let us define some additional subsets of \mathcal{M} : Let $\mathcal{G}_0(\mathcal{M})$, $\mathcal{G}_1(\mathcal{M})$, and $\mathcal{G}_2(\mathcal{M})$ denote the set of maps that participate in exactly 0, 1, and 2 3-cycles with positive distortion, respectively. Since \mathcal{M} is almost-accurate, the four sets $\mathcal{B}(\mathcal{M})$, $\mathcal{G}_0(\mathcal{M})$, $\mathcal{G}_1(\mathcal{M})$, and $\mathcal{G}_2(\mathcal{M})$ form a partition of \mathcal{M} . With a slight abuse of notation, let \mathcal{B} , \mathcal{G}_0 , \mathcal{G}_1 , and \mathcal{G}_2 denote the corresponding sets of edges. Note that since $w_e = \infty$ for all $e \in \mathcal{G}_0$, we may assume $\hat{c}_e = 0$ for all $e \in \mathcal{G}_0$.

Now any 3-cycle γ such that $C_\gamma > 0$ must include a single edge $b \in \mathcal{B}$ and two edges $g_1, g_2 \in \mathcal{G}_1 \cup \mathcal{G}_2$. We then rewrite the corresponding constraint $\hat{c}_b + \hat{c}_{g_1} + \hat{c}_{g_2} \geq C_\gamma$ as $w_b \hat{c}_b + w_b \hat{c}_{g_1} + w_b \hat{c}_{g_2} \geq 1/(n-2)$. If we sum this over all 3-cycles γ such that $C_\gamma > 0$, we obtain the following:

$$(|\mathcal{S}| - 2) \sum_{b \in \mathcal{B}} w_b \hat{c}_b + \sum_{g \in \mathcal{G}_1} w_{b_{g_1}} \hat{c}_g + \sum_{g \in \mathcal{G}_2} (w_{b_{g_1}} + w_{b_{g_2}}) \hat{c}_g \geq |\mathcal{B}| \quad (4)$$

where b_{g_1} and b_{g_2} are the edges in \mathcal{B} that share some 3-cycle with g . But for all $g \in \mathcal{G}_1$, $(|\mathcal{S}| - 2)w_g = (|\mathcal{S}| - 2)/E_{acc}(m_{b_{g_1}}) > 1/((|\mathcal{S}| - 2)E_{acc}(m_{b_{g_1}})) = w_{b_{g_1}}$. Also, for all $g \in \mathcal{G}_2$, $(|\mathcal{S}| - 2)w_g > w_{b_{g_1}} + w_{b_{g_2}}$ using the inequality constraining the values of E_{acc} over $\mathcal{B}(\mathcal{M})$. (The proof is simple algebraic manipulation and omitted for brevity.) Substituting these values in and dividing by $|\mathcal{S}| - 2$, we obtain $\sum_e w_e \hat{c}_e \geq |\mathcal{B}|/(|\mathcal{S}| - 2)$, with equality only if $\hat{c}_g = 0$ for all $g \notin \mathcal{B}$. Therefore \hat{c}_e is optimal if and only if $\hat{c}_{(i,j)} = E_{acc}(m_{ij})$. \square

If E_{acc} is normalized so that $0 \leq E_{acc}(m) \leq 1$ for all maps m , then we may satisfy the constraint on the accuracy of bad maps by satisfying $E_{acc}(m) > 1/((|\mathcal{S}| - 2)^2 - 3)$ for all $m \in \mathcal{B}(\mathcal{M})$. (Note that neither this new constraint nor the original constraint can be satisfied when $|\mathcal{S}| < 5$.) Intuitively, this requirement means that maps we consider to be bad must be significantly bad; it is easy to construct an example where a single “slightly” bad map among several “extremely” bad maps will be assigned slightly *lower* costs by the LP than the perfectly good maps that share a shape with it. Also note that this constraint is quadratic in $|\mathcal{S}|$, and so it rapidly becomes irrelevant as the size of the collection increases.

When the ground-truth maps are only ϵ -distance-preserving, the constraints on the values of E_{acc} are more stringent. Using a similar normalization, we satisfy these constraints when $E_{acc}(m) > \max\{1/(|\mathcal{S}| - 7), 4\epsilon\}$. Since this constraint is linear as opposed to the quadratic constraint obtained earlier, it requires larger collections in order to be useful. Intuitively, increasing the size of the collection allows the algorithm to handle a wider range of quality of maps, up to a limit imposed by the quality of the ground-truth maps.

4.2. The Algorithm

So far we have shown how to infer accuracy from consistency in special circumstances, under strict conditions on the ground truth maps, the distance metric, and the existence of inaccurate maps. In general, however, maps are never exactly accurate, and inaccurate maps are much more prevalent than allowed by Theorem 1. Nevertheless, we can use the optimization problem devised for the idealized setting to design an algorithm for improving collections of maps.

Our main motivation stems from the fact that even if not all inaccurate edges are isolated, some 3-cycles might still contain a single inaccurate edge, which should be identified by the optimization described in the previous section. If the costs c_e computed by the optimization indeed approximate to some extent the accuracy of the maps, we can find better maps \mathcal{M}^1 by composing the original ones along shortest paths on G , weighted by c_e , as we did in Figure 3. If we now consider 3-cycles on \mathcal{M}^1 , there might be additional 3-cycles which contain a single inaccurate map. Note that these 3-cycles correspond to larger cycles on \mathcal{M} . Therefore, by repeating these two steps — finding approximate accuracies and composing maps using shortest paths — we can iteratively improve the maps using the information from progressively larger cycles. This replaces the need in the method by Zach et al. [ZKP10] to randomly sample larger cycles.

This leads to an iterative algorithm, described in Algorithm 1. Convergence is reached when either all the shortest paths are of length 1 (i.e. a single edge), or when the total consistency error is below some threshold.

<p>Input: A collection of shapes \mathcal{S} and maps \mathcal{M} Output: A new collection of maps \mathcal{M} $\mathcal{M}^0 \leftarrow \mathcal{M}; k \leftarrow 0$ repeat $W^k = \{c_{(i,j)}\} \leftarrow \text{Solve Problem 2 on } \mathcal{M}^k$ forall the $(i, j) \in E$ do $\gamma_{ij} \leftarrow \text{ShortestPath}(G_{\mathcal{M}^k}, i, j, W^k)$ $m_{i,j}^{k+1} \leftarrow m_{i,j}^k$ end $\mathcal{M}^{k+1} \leftarrow \{m_{i,j}^{k+1}\}$ $k \leftarrow k+1$ until converged $\hat{\mathcal{M}} \leftarrow \mathcal{M}^{k-1}$</p>
--

Algorithm 1: Improving a collection of maps

Using the same setup as described in the previous section, we applied Algorithm 1 to a subset of 10 shapes from the SCAPE database, this time using more symmetric maps, such that the map collection has some 3-cycles with more than one symmetric map. The resulting behavior depends on the number of symmetric maps and on whether every shape has enough “good” maps — as we are composing maps, our output can only be good if we are given enough good maps.

Figure 5 shows the results for varying proportions of straight and symmetric maps. In (a), we removed all the symmetric maps and remained with an accurate collection. In

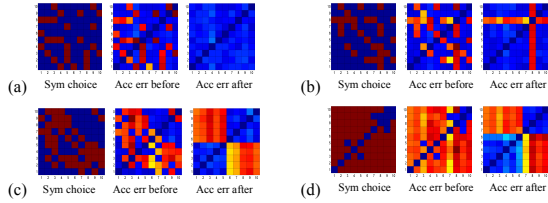


Figure 5: Results on a collection of human shapes, where some maps — 28 (a), 30 (b), 46 (c), and 82 (d) out of 90 — were chosen to be symmetric. (left) Symmetric choice, accuracy error of (center) original maps, and (right) final maps.

(b), we have a configuration where one shape (shape 8) does not have enough “good” maps, and therefore not all the symmetries can be resolved. In (c), we have almost 50 percent bad maps. The process converges to a fully consistent state by partitioning the shapes into two groups S_1, S_2 , such that the maps $m_{i,j}$ are straight if S_i and S_j belong to the same group, and symmetric otherwise. Finally, in (d), although we have only four pairs of shapes with straight maps, the method still manages to reduce the accuracy error, and converge to a consistent state. This experiment suggests that if enough “good” maps are provided, the process will find the correct composition of maps to improve both accuracy and consistency. In all the experiments we did (including those in the next section) the process always converged, and the LP energy always decreased after each iteration; however, we do not currently have a proof that this is indeed guaranteed to occur. We have observed that path replacements tend to improve consistency, as all shapes within a path used as a replacement become consistent with each other afterwards; we have also observed that sparse assignments of costs result in more path replacements, as sparse assignments are more likely to violate the triangle inequality. These observations suggest that the process should converge in many cases.

5. Experimental Results

In this section we show a few more examples of the application of Algorithm 1 to various collections of shapes. Since the ground truth data is not always available, in some cases we use different methods to evaluate our performance.

5.1. Limitations

Before discussing the results we would like to address the limitations of our method. First and foremost, we are very much dependent on the quality of the input maps. If we are given a collection where not enough maps are good, in many cases the accuracy of the output maps will not improve (although the consistency usually will still improve, see Fig. 5). We conjecture that if the majority of maps for each shape are good, then the output maps of our method will all be good.

Second, the complexity of our method is quite high, as

we need to not only obtain but also compose dense maps many times, and then evaluate all of those maps over the entire shapes. Composing and evaluating all the maps takes tens of seconds per iteration when we have 20 shapes with 10000 points each, and it grows cubically with the number of shapes. However, the composition and evaluation of maps may be done massively in parallel, so the primary serial step is the solving of the LP. Tests with random weights show that solving the LP once for 100 shapes takes around 20 minutes on a modern laptop. Unfortunately, we lack a bound on the number of iterations (and therefore number of LP instances) required; however, there is the alternative to cap the number of iterations performed. Furthermore, in all the experiments presented, the process converged in 8 or fewer iterations.

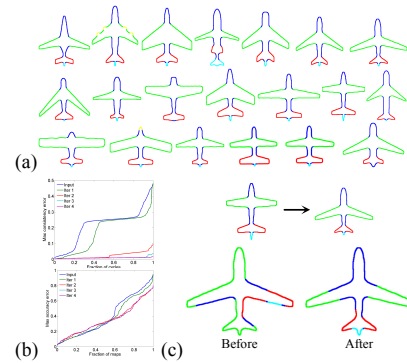


Figure 6: (a) shapes, (b) accuracy and consistency errors, given by the fraction of shapes/cycles whose error is smaller than a threshold. (c) Example of initial and final maps.

5.2. 2D Maps

Our method is quite general, and can be easily applied both to 2D and 3D shapes. To demonstrate our method in the 2D setting, we used as our input shapes the outline curves of 20 airplane meshes from the Princeton Segmentation Benchmark data set [CGF09] projected onto the xy -plane. To generate the maps \mathcal{M} we used dynamic time warping [RRL78] on the curvature function of the outline curves. We took the metric d_S to be the curve length between two points divided by the total length of the curve. Since the ground truth maps are not available in this case, we used the segmentation labels generated by [KHS10] to evaluate the maps. The accuracy error $E_{acc}(m_{i,j})$ is measured as the fraction of points $p \in S_j$ which are labeled incorrectly: $label_i(p) \neq label_j(m_{i,j}(p))$. Note that in this case the distance metrics d_S we used for the accuracy and consistency errors are not the same. However, our algorithm does not require the accuracy errors; they are used solely for evaluation.

Fig. 6 shows our results on this dataset. To show the change in accuracy and consistency errors, we measure the fraction of shapes x whose error is smaller than e and plot e as a function of x . For example, 60% of the input maps had

accuracy error below 0.6, whereas 60% of the output maps had an error below 0.4. In addition we show one map, and how it improved. We visualize the map by transferring the segmentation from the source to the target. It can be seen that our method improved both the accuracy and the consistency of the collection and resulted in better maps.

5.3. 3D Morph

Symmetry is not the only obstruction to obtaining good maps, and our algorithm can be applied in other cases as well. For example, for many shapes the quality of the map depends heavily on how “close” the shapes are, with closer shapes yielding better maps. To check the performance of our method in that situation, we took \mathcal{S} to be 20 uniformly spaced frames from the a morph sequence of an elephant to a horse. On these shapes we computed all the pairwise maps using Möbius voting [LF09], and interpolated those to a full map using a variant of the GMDS technique [BBK06]. This data set is in correspondence, so we used that as our ground truth map, taking d_S to be the Euclidean distance normalized by the diagonal of the bounding box as an inexpensive approximator of geodesic distance, as in our earlier example.

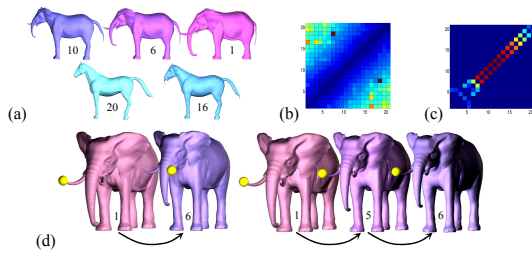


Figure 7: (a) Shapes, (b) LP costs after one iteration, (c) number of times original maps were used in the final maps, (d) The original map (1, 6), and the better map computed by the composition (1, 5, 6).

Fig. 7 shows the results. In (a) we show a few of the shapes from the morph sequence (frames 1,6,10,16,20). Since consecutive frames in the morph are more similar than other pairs, we expect the accuracy of the maps to degrade the further we are from the diagonal of the accuracy error matrix. In (b) we show the costs c_e computed in the first iteration. It is evident that we identified correctly the relative quality of the maps. It is interesting to check which original maps are used the most in the final maps, as this information discloses something about the structure of the collection. Fig. 7(c) shows the number of times the original maps were used in the final maps. As expected, the maps which are used the most are the ones between consecutive frames — thus in effect our algorithm recovers the morph sequence order just from the quality of the shape maps. In (d) we show an example of an original map between two remote frames, and its replacement by the composition of two better maps, selected using the shortest path on c_e .

5.4. 3D Shape Collections

We applied our algorithm to a few collections of 3D shapes: the deformed Armadillos used in the heat kernel map paper [OMMG10], as well as 10 shapes from the hand category in the Princeton Segmentation Database [CGF09]. For these collections we used the heat kernel map to generate a set of 100 sparse correspondences, which we then interpolated to a full map using a GMDS [BBK06] variant. As in previous examples we used the Euclidean distance normalized by the diagonal of the bounding box for d_S . In addition, we applied our method to a few collections used in the paper Blended Intrinsic Maps [KLF11].

Armadillos We computed all the pairwise maps between the Armadillos using the fully automatic mode of the HK map, and applied our algorithm to the resulting maps. These models are in correspondence, which we used as the ground truth for evaluating the accuracy of the original and new maps. Fig. 8 shows (a) the shapes, and (b) a plot of the accuracy and (c) consistency errors. In this case, some shape pairs had larger geodesic distortion than others, leading to different map qualities. For example, the shape “1” in (a) had a single good map to it from the shape “2” (Fig. 8(d)). This caused the final maps of all the other shapes to include a path that goes through “2”. (e-f) show two such examples: the top row shows the source shape, and the bottom row shows the map before and after the compositions. In both cases, in the original map, some of the points were mapped correctly, and others were mapped to the symmetric part. After the composition, all the points shifted to better locations. After the maps to “1” were fixed, all maps had good accuracy and consistency, and the algorithm finished after one iteration.

Segmented hands We ran our algorithm on a collection of hand models from the Princeton segmentation database, which are not in correspondence. Since we do not have the ground truth we used the same evaluation method as in the 2D case, measuring the fraction of vertices which were mapped to the correct segment. Fig. 9 shows (a) the shapes, (b) the accuracy and (c) consistency error plots, and (d) the improvement of a set of maps. We show the source shape with a few points marked, and for the other shapes we show the corresponding points under the original maps (top) and composed maps (bottom). In all these cases, the composed maps fixed all the points on the first, fourth and fifth fingers. The second and third fingers are still flipped in a few cases.

Collections from Blended Conformal Maps We used the animals, hands, humans and teddies collections, each containing 20 shapes, for which we computed all pairwise maps using the Blended Maps method [KLF11]. These collections are accompanied by ground-truth sparse correspondences, which we used to evaluate the accuracy errors. We took $E_{acc}(m_{A,B})$ to be the average geodesic distance on B between the mapped correspondences and their correct locations, normalized by the square root of the surface area of B .

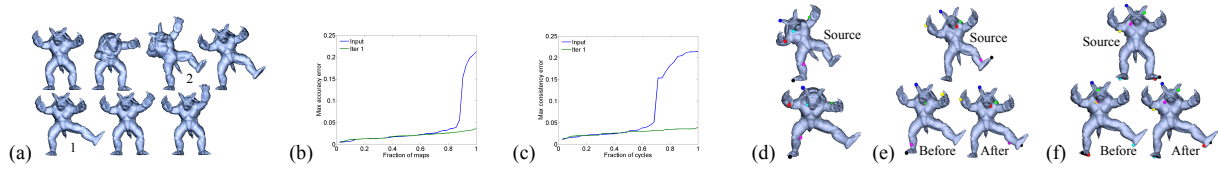


Figure 8: (a) Shapes, (b) accuracy and (c) consistency errors, (d-f) a few examples of maps. Source shapes (top row), and the maps (bottom row), before and after running the algorithm. See the text for details.

Fig. 10 shows some maps computed for the (a) animals, (b) hands, (c) teddies and (d,e) humans. For each group we show the source shape with its index, the original maps (top) and the composed maps (bottom). Fig. 11 shows the accuracy error matrices of the collections before (top) and after (bottom) applying our algorithm. Furthermore, the figure shows the accuracy error plots of these collections before (dashed lines) and after (solid lines).

Both figures indicate that our method improved on the maps generated by [KLF11] with varying success rates. For the animals collections, only 7 bad maps existed, and they were all isolated, so our method found them all and fixed them (Fig. 10(a)). Note how the legs are mapped correctly for all the shapes. For the hands collection we also significantly improved the maps, as can be seen both in the error matrix, and in the plot. Fig. 10(b) shows examples of a few of the improved maps from shape 1 in that collection.

The humans collection exhibits the interesting behavior we have also encountered in previous examples: shapes which do not have a majority of good maps associated with them (such as shapes 16 and 18) do not improve. In fact, as can be seen in Fig. 10(d), some of the good maps are replaced by bad maps, to improve the consistency. The maps from shape 18 to 9,11,12 were good, and they were replaced by bad maps consistent with 18 to 10, and with the other bad maps from shape 18. On the other hand, since most human shapes do have a majority of good maps, all the bad maps between these shapes were cleaned up. Fig. 10(e) shows the maps from shape 9 to a set of other shapes. Note how the hands, feet, knees and shoulders are mapped correctly on all shapes. It is also worth noting that the global accuracy of the collection has improved in this case, as shown in the accuracy plot. The teddies collection exhibits similar behavior.

It is interesting to take a closer look at some of the improved maps in the teddies collection (Fig. 10(c)). For example, the consistency condition allowed us to infer correctly the location of the hip (the light pink point). Consider the map from shape 18: although the geometry of the hip point is very different for the source and target shapes, causing the original mapping algorithm to confuse it with a more curved area near the nose, our algorithm has used the additional information from the collection to correctly identify this point.

6. Conclusions and Future Work

Multiple pairwise maps between shapes create a context that allows an improved evaluation of individual maps. We have shown how to use an optimization-based framework to infer the accuracy of a map from the inconsistencies it generates in a collection of other maps. In the special case when only a relatively small number of the maps are not accurate, we can prove under idealized conditions that one step of our algorithm identifies correctly the inaccurate maps, and returns a better set of alternative maps. For more complicated collections where the number of inaccurate maps might be large, we show experimentally on a variety of shape collections that our iterative algorithm persistently generates maps which improve on the originals both in accuracy and consistency. Finally, we have shown that by considering which of the original maps are most used in the composition process, we can identify the important “links” within the collection, therefore extracting to some extent structural relationships within the collection of shapes which may reflect on the underlying process by which the shapes were generated.

The topic of investigating maps between shapes in the context of a collection of multiple such maps is at its infancy, and we believe that much more fruitful work can be done on this subject. First, an obvious task for future research is to prove our conjecture that the iterations converge, and to provide some theoretical guarantees on the consistency and accuracy of the resulting maps in the more general case. Second, our work currently considers only pairwise compositions of maps, and better ways of combining maps can be devised, such as the method proposed in [KLF11]. Finally, we have only touched upon the possibility of investigating the structure of the collection using the map composition information, and we hope it may be possible to correlate such phenotypic structure with other deeper information in the data based on the semantics of the dataset, such as morphological evolution in biology or stylistic variability in the case of human-designed shapes.

Acknowledgements

This work was supported by NSF grants FODAVA 0808515, CCF 1011228, ONR MURI grant N0001470710747 and the Weizmann Institute WiS Award. The authors would like to thank the anonymous reviewers for their helpful comments.

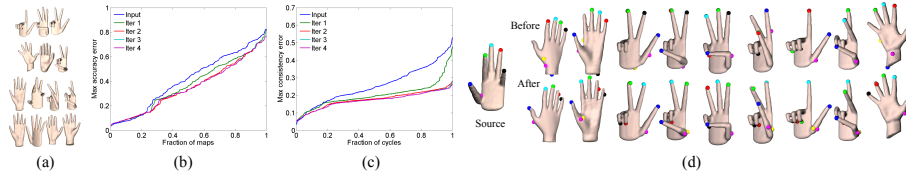


Figure 9: (a) Shapes, (b) accuracy and (c) consistency error plots, (d) original (top row) and composed (bottom row) maps.

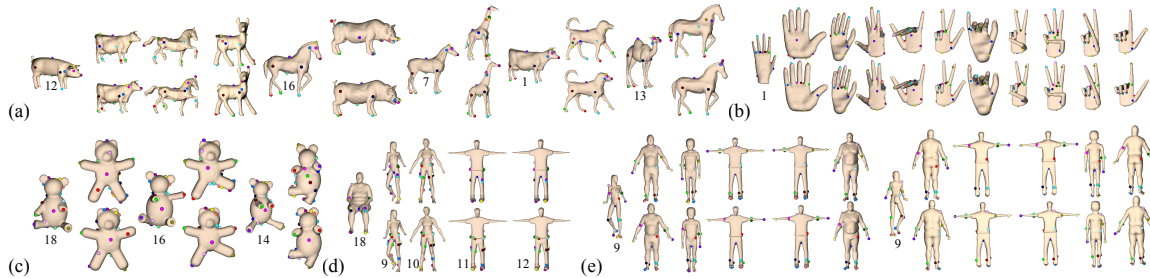


Figure 10: (a-c,e) Maps before (top) and (after) our algorithm for a few collections, (d) example of fail case See text for details.

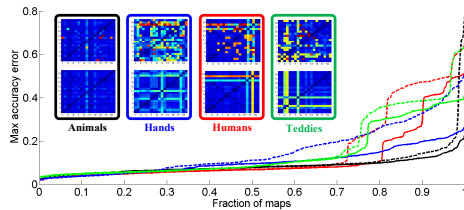


Figure 11: The accuracy error matrices and plots for a few shape collections. See text for details.

References

[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 (2005), 408–416. 6

[Bar07] BARANIUK R.: Compressive sensing. *IEEE Signal Processing Mag* (2007), 118–120. 5

[BBK06] BRONSTEIN A. M., BRONSTEIN M. M., KIMMEL R.: Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. of Nml. Academy of Sci.* 103, 5 (2006), 1168–1172. 9

[Çel98] ÇELA E.: *The Quadratic Assignment Problem: Theory and Algorithms*. Kluwer Academic Publishers, 1998. 1

[CGF09] CHEN X., GOLOVINSKIY A., FUNKHOUSER T.: A benchmark for 3D mesh segmentation. *In Proc. SIGGRAPH 2009* (2009). 8, 9

[Cha09] CHARPIAT G.: Learning shape metrics based on deformations and transport. *In Proc. NORDIA 2009* (2009). 3

[CT05] CANDÈS E. J., TAO T.: Decoding by Linear Programming. *IEEE Transactions on Information Theory* 51, 12 (Dec. 2005), 4203–4215. 5

[dGGV08] DE GOES F., GOLDENSTEIN S., VELHO L.: A hier-

archical segmentation of articulated bodies. *In Proc. SGP* (2008). 3

[Gro07] GROMOV M.: *Metric Structures for Riemannian and Non-Riemannian Spaces*. Modern Birkhäuser Classics, 2007. 1

[KHS10] KALOGERAKIS E., HERTZMANN A., SINGH K.: Learning 3d mesh segmentation and labeling. *In Proc. SIGGRAPH 2010*. 8

[KLF11] KIM V. G., LIPMAN Y., FUNKHOUSER T.: Blended intrinsic maps. *In Proc. SIGGRAPH 2011* (2011). 3, 9, 10

[LF09] LIPMAN Y., FUNKHOUSER T.: Mobius voting for surface correspondence. *In Proc. SIGGRAPH 2009* (2009). 3, 9

[LJX*10] LAI Y.-K., JIN M., XIE X., HE Y., PALACIOS J., ZHANG E., HU S.-M., GU X.: Metric-driven rosy field design and remeshing. *IEEE Trans. on Vis. and Comp. Graph.* 16 (2010), 95–108. 3

[OMMG10] OVSJANIKOV M., MÉRIGOT Q., MÉMOLI F., GUIBAS L.: One point isometric matching with the heat kernel. *In Proc. SGP 2010* (2010). 3, 6, 9

[Reu10] REUTER M.: Hierarchical Shape Segmentation and Registration via Topological Features of Laplace-Beltrami Eigenfunctions. *Int. J. of Comp. Vis.* 89, 2 (2010), 287–308. 3

[RRL78] RABINER L. R., ROSENBERG A. E., LEVINSON S. E.: Considerations in dynamic time warping algorithms for discrete word recognition. *The J. of the Acoustical Soc. of America* 63, S1 (1978), S79. 3, 8

[RSSS11] ROBERTS R., SINHA S. N., SZELISKI R., STEEDLY D.: Structure from motion for scenes with large duplicate structures. *In Proc. CVPR 2011* (2011). 3

[Tdl00] TENENBAUM J., DE SILVA V., LANGFORD J.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 5500 (2000), 2319–2323. 3

[YK06] YANKOV D., KEOGH E.: Manifold clustering of shapes. *In Proc. IEEE Int. Conf. Data Mining (ICDM)* (2006). 3

[ZKP10] ZACH C., KLOPSCHITZ M., POLLEFEYS M.: Disambiguating visual relations using loop constraints. *In Proc. CVPR 2010* (2010). 3, 7