Consistent Partial Matching of Shape Collections via Sparse Modeling

L. Cosmo[1], E. Rodolà[2], A. Albarelli[1], F. Mémoli[3], D. Cremers[2]

[1]University of Venice, Italy [2]TU Munich, Germany [3]Ohio State University, U.S.

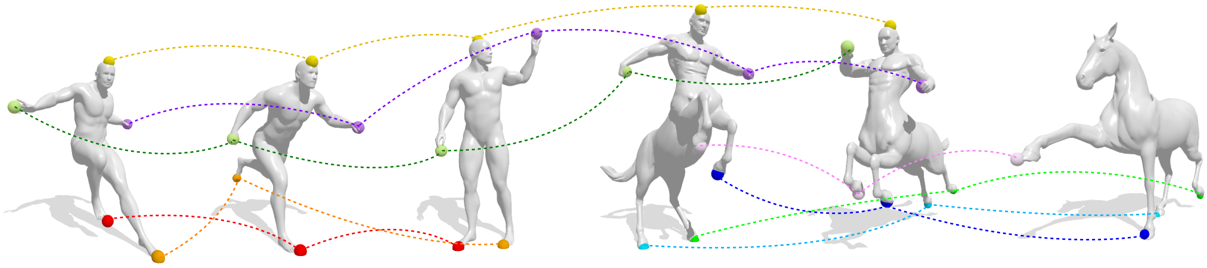


Figure 1: *A partial multi-way correspondence obtained with our approach on a heterogeneous collection of shapes. Our method does not require initial pairwise maps as input, as it actively seeks a reliable correspondence by operating directly over the space of joint, cycle-consistent matches. Partially-similar as well as outlier shapes are automatically detected and accounted for by adopting a sparse model for the joint correspondence. A subset of all matches is shown for visualization purposes.*

Abstract

Recent efforts in the area of joint object matching approach the problem by taking as input a set of pairwise maps, which are then jointly optimized across the whole collection so that certain accuracy and consistency criteria are satisfied. One natural requirement is cycle-consistency – namely the fact that map composition should give the same result regardless of the path taken in the shape collection. In this paper, we introduce a novel approach to obtain consistent matches without requiring initial pairwise solutions to be given as input. We do so by optimizing a joint measure of metric distortion directly over the space of cycle-consistent maps; in order to allow for partially-similar and extra-class shapes, we formulate the problem as a series of quadratic programs with sparsity-inducing constraints, making our technique a natural candidate for analyzing collections with a large presence of outliers. The particular form of the problem allows us to leverage results and tools from the field of evolutionary game theory. This enables a highly efficient optimization procedure which assures accurate and provably consistent solutions in a matter of minutes in collections with hundreds of shapes.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Shape Analysis

1. Introduction

Finding matches among multiple objects is a research topic that has received a good deal of attention in recent years. In its most common formulation, it translates to the problem of determining point-to-point maps between all shapes in a collection, subject to the requirement that the extracted correspondence be in some way "consistent". To this end, a natural and widely accepted criterion is *cycle-consistency* [ZKP10], namely that composition of maps along loops in the collection should approximate the identity. So far, the problem has been approached by independently computing pairwise maps [LH05, KLF11] between the objects in the collection; the set of maps is then given as input to a global optimizer which updates them so as to improve their quality and produce a final, consistent correspondence. Although most of these approaches work well provided that the input maps are sufficiently accurate, they suffer in the presence of noise (incorrect matches) in the maps themselves, or outlier (extra-class) shapes in the collection. Further, due to the combinatorial difficulty of imposing the consistency requirement, many of the existing schemes provide no guarantee that cycle-consistency is satisfied exactly.

In this paper, we introduce a new method for the joint matching of multiple deformable shapes in a collection. Unlike the common approach outlined above, we do *not* require any pairwise correspondence to be given as input, and instead formulate the problem as an optimization directly over the space of cycle-consistent (multi-way) matches.

1.1. Related work

Probably the earliest attempt to tackle multiple shape matching in a principled way is the synchronistic matching approach of Schmidt *et al.* [STCB07]. Given a collection of planar shapes, the authors model the joint matching problem as the search of a shortest path in their product space. Due to the resulting intractability, the problem is relaxed to a series of pairwise sub-problems, and the cycle-consistency criterion introduced as a regularizer. The method allows to improve initial pairwise solutions, but consistency is not satisfied exactly and the method operates under the assumption that all shapes in the collection are similar. Pachauri *et al.* [PKS13] took a similar perspective by formulating the matching problem using the language of combinatorial optimization; due to the spectral relaxation they perform, the method tends to be sensitive to noise and outliers. Recently, Yan *et al.* [YLL*14] formulated the problem as one of simultaneous multi-graph matching [SRS13], but similarly to [STCB07,PKS13], cycle-consistency is relaxed and gradually infused in a pairwise matching process as a regularizer.

Zach *et al.* [ZKP10] were probably the first to make an explicit attempt at finding solutions meeting the cycle-consistency requirement. Starting from an initial graph of pairwise associations among the objects in the collection, they detect and remove erroneous edges as the ones giving rise to inconsistent loops in the graph. As an extension to this approach, Nguyen *et al.* [NBCW*11] apply global optimization to select cycle-consistent maps while at the same time allowing edges to be replaced by better map compositions. The method performs well when the *full* point-to-point correspondence is known and accurate for all pairs of objects. Huang *et al.* [HZG*12] improved upon [NBCW*11] by allowing sparse correspondences, and later rephrased the problem by replacing the pairwise maps with a spectral counterpart [HWG14]; however, the approaches do not apply when the shapes being matched are only *partially* similar [RCB*16]. Finally, Sahillioğlu and Yemez [SY14] proposed a greedy algorithm that seeks nearly-isometric consistent solutions across all shapes in the collection. The approach only works well when matching shape extremities, and it is susceptible to outlier shapes and partiality. In particular, its accuracy depends on the specific ordering of the shapes in the collection.

All the methods outlined above demonstrate good practical performance in controlled settings, however there has been a general lack of theoretical guarantees that ensure correctness of the final correspondence under unfavorable conditions. First steps in this direction are taken by Huang and Guibas [HG13], who formulate a convex relaxation to the joint matching problem using the language of semi-definite programming. The authors derive theoretical guarantees on the recovery of the correct joint correspondence from noisy input maps [KLF11]. Very recent works in the field of information theory explore this direction more abstractly [CG14], giving conditions for perfect recovery under large outlier ratios. Chen *et al.* [CGH14] and Kezurer *et al.* [KKBL15] recently applied this analysis to consistently match partially similar objects from a small fraction of densely corrupted pairwise maps. To our knowledge, their algorithms currently represent the state of the art within this family of approaches.

1.2. Contribution

In this paper, we introduce a novel technique to construct accurate, consistent correspondences within shape collections. Our formulation has the following key properties:

- The method operates by optimizing directly over the space of cycle-consistent correspondences, without requiring pairwise maps to be given as input. As a result, cycle-consistency is satisfied *exactly* by construction.
- We employ sparsity techniques in order to cope with *partially similar* as well as *outlier* shapes in the collection – an aspect that has received limited interest so far, but that can frequently occur in practical scenarios.
- Our proposed method is easy to implement. Further, it compares favorably with the state of the art on challenging datasets while being orders of magnitude faster.

2. Preliminaries

We model shapes as compact two-dimensional Riemannian manifolds S_i (possibly with boundary) embedded in $\mathbb{R}^3$, equipped with the intrinsic distance function d_i.

Let us be given a collection $\mathcal{C} = \{S_1, S_2, \dots, S_n\}$ of n shapes. The product space $S_1 \times \cdots \times S_n$ consists of all possible n-way (*i.e.*, joint) matches between the shapes in $\mathcal{C}$. However, in practical settings it is often the case that outliers (*e.g.*, shapes belonging to different classes) as well as partially similar shapes (*e.g.*, man and centaur) are present in the collection (see Fig. 2). In order to deal with such cases, we extend the set of possible joint matches as follows. Formally, we consider the set constructed as the union

$$\check{\Gamma} = \bigcup_{k \in I} \prod_{j \in k} S_j \,, \qquad (1)$$

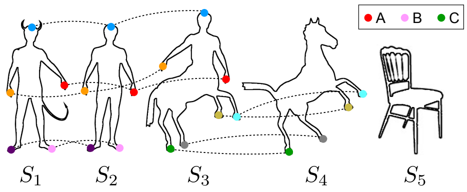


Figure 2: *A collection of shapes may carry partiality at different levels. Our method allows to extract consistent correspondences reliably under partial similarity* (e.g., $S_1/S_2/S_3$) *and at the same time detect and avoid outlier shapes* (S_5)*.*

where I is a collection of index sets k defined by the power set (denoted by $\mathcal{P}$) relation

$$I = \{k : k \in \mathcal{P}(\{1,2,\ldots,n\}) \wedge |k| > 1\}\,. \qquad (2)$$

In other words, $\check{\Gamma}$ is the set of all possible m-fold Cartesian products between the shapes in $\mathcal{C}$, with $1 < m \leq n$. Clearly, this set also includes $S_1 \times \cdots \times S_n$ and in particular $|\check{\Gamma}|$ grows exponentially with the number of shapes. Each element $\gamma \in \check{\Gamma}$ with $|\gamma| = d \leq n$ now represents a joint match between a subset of d shapes from the collection.

Definition 1 We define a *multi-way match* among $d \leq n$ shapes to be any element $\gamma \in \check{\Gamma}$. A multi-way match is represented as the ordered d-tuple

$$\gamma = (p_i)_{i \in k} \quad \text{with } k \in I \text{ and } p_i \in S_i\,,$$

where k is a sequence of *shape indices*, denoting the shapes matched by γ. We will write $p_i \in \gamma$ to say that the vertex p_i is matched via γ.

Note that two multi-way matches $\gamma, \gamma' \in \check{\Gamma}$ may in general have different lengths $|\gamma|$ and $|\gamma'|$. In particular, they may or may not have shapes in common. We will therefore define the *overlap* $\gamma \cap \gamma'$ as the longest common subsequence of their shape indices. For example, in Fig. 2 we show the multi-way matches $A, B, C \in \check{\Gamma}$. For A and B we have the overlap $A \cap B = (1,2)$, whereas $A \cap C = (3)$ and $B \cap C = \emptyset$.

For our purposes, we are interested in subsets of $\check{\Gamma}$ that satisfy certain properties, as described in the following:

Definition 2 A *multi-way correspondence* between the n shapes in $\mathcal{C}$ is a subset $\Gamma \subset \check{\Gamma}$ satisfying: for every $S_i \in \mathcal{C}$ and for every $p_i \in S_i$, there exists at least one $\gamma \in \Gamma$ such that $p_i \in \gamma$.

The above definition ensures that in a multi-way correspondence each vertex of each shape is matched to corresponding vertices on (a subset of) the other shapes. We now define what is the meaning of cycle-consistency in our setting.

Definition 3 We say that a multi-way correspondence Γ between shapes in the collection $\mathcal{C} = \{S_i\}_{i=1}^n$ is *cycle-consistent* if, for any $j, k, \ell \in \{1,\ldots,n\}$, whenever Γ matches $p_j \in S_j$ to $p_k \in S_k$ and matches p_k to $p_\ell \in S_\ell$, then Γ also matches p_j to p_ℓ.

Remark 1 A multi-way match is always cycle-consistent by construction, since it is an element of a product set. This applies to any cycle, with length possibly longer than 3.

Note that while individual multi-way matches are always cycle-consistent, a fixed point $p_i \in S_i$ might be mapped to multiple points on the other shapes by a multi-way correspondence. We therefore introduce the following notion:

Definition 4 Two distinct multi-way matches $\gamma, \gamma' \in \Gamma$ are said to be *incompatible* whenever $p_i \in \gamma$ and $p_i \in \gamma'$ for some $p_i \in S_i$ and $i \in \{1,\ldots,n\}$.

An illustration of incompatible matches is given in Fig. 3.

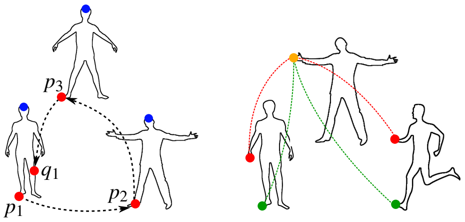


Figure 3: Left*: The red matches violate cycle-consistency, since* $p_1 \neq q_1$. Right*: Example of two incompatible multi-way matches (red and green): the hand in the middle shape is assigned to multiple distinct points on the other shapes.*

3. Problem statement

The goal of joint object matching is to determine a (possibly dense) correspondence among multiple shapes in a collection, with the requirement that the correspondence be consistent along cycles of any length. In this Section we formulate the joint matching problem as one of *minimum-distortion correspondence* [Mém11]. Differently from most other approaches [STCB07, ZKP10, NBCW*11, HZG*12, PKS13, SY14, YLL*14] our formulation comes with the *theoretical guarantee* of cycle-consistency, and additionally deals with partially similar as well as outlier shapes in a natural way.

Metric distortion. Suppose we are given two multi-way matches $\gamma, \gamma' \in \Gamma$ respectively putting $|\gamma|$ and $|\gamma'|$ points into correspondence, where in general $|\gamma| \neq |\gamma'|$. Then, we can quantify the quality of the correspondence by the cost function $\varepsilon : \Gamma \times \Gamma \to \mathbb{R}_+ \cup \{\infty\}$ defined as:

$$\varepsilon(\gamma, \gamma') = \max_{\substack{p_k, p_\ell \in \gamma \\ p'_k, p'_\ell \in \gamma'}} |d_k(p_k, p'_k) - d_\ell(p_\ell, p'_\ell)|\,. \qquad (3)$$

Here we tacitly assume that the multi-way matches are compared only on their overlap, *i.e.*, over the shapes in common. In (3) we put $\varepsilon(\gamma, \gamma') = \infty$ whenever γ and γ' are incompatible (see Fig. 3) or non-overlapping. This definition of cost encodes the maximum metric distortion attained by the two multi-way matches across the shape collection.

Multi-way L^p distortion. A multi-way correspondence

$\Gamma \subset \check{\Gamma}$ can be alternatively modeled as a binary function $g : \check{\Gamma} \to \{0,1\}$ such that for every $S_i \in \mathcal{C}$ and for every $q \in S_i$,

$$\sum_{\substack{\gamma \in \check{\Gamma} \\ \text{s.t. } q \in \gamma}} g(\gamma) \geq 1\,. \tag{4}$$

Function g can be seen as an indicator function over the space of all possible multi-way matches. Then, the condition above simply asks that each point in each shape is contained in at least one $\gamma \in \check{\Gamma}$ for which $g(\gamma) = 1$, thus being a strict requirement to match all points in all shapes.

The overall metric distortion caused by a correspondence Γ can be measured by the L^p distortion:

$$\|\varepsilon\|^p_{L^p(g\times g)} = \sum_{\gamma,\gamma' \in \check{\Gamma}} \varepsilon^p(\gamma,\gamma') g(\gamma) g(\gamma')\,, \tag{5}$$

with $p \geq 1$. Now, determining a correspondence of minimum distortion amounts to seeking a minimizer (not unique in general) to:

$$\min_{g:\check{\Gamma}\to\{0,1\}} \|\varepsilon\|^p_{L^p(g\times g)} \tag{6}$$

where g ranges over all correspondences $\Gamma \subset \check{\Gamma}$.

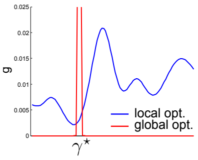
Example. In the specific case where $n = 2$, a multi-way match $\gamma = (p_1, p_2)$ reduces to a pair of points and the error criterion of Eq. (3) simplifies to the absolute metric distortion $\varepsilon((p_1,p_2),(q_1,q_2)) = |d_1(p_1,q_1) - d_2(p_2,q_2)|$. Then, by taking the limit for $p \to \infty$ the expression (6) yields the classical Gromov-Hausdorff distance between metric spaces (S_1, d_1) and (S_2, d_2) [Mém11].

Dealing with partiality. The combinatorial complexity of optimizing over all possible multi-way correspondences $\Gamma \subset \check{\Gamma}$ makes the problem intractable even for small collections. Partial remedy to this issue is provided by relaxing the binary map to take continuous values, *i.e.*, $g : \check{\Gamma} \to [0,1]$.

We further note that, although Eq. (1) enables us to better deal with partially similar shapes, the constraint defined in Eq. (4) requires us to match all points in all shapes. However, we would like outlier shapes to not partake to the final correspondence. Furthermore, we want to allow *individual* shape points to be left unmatched if they do not find suitable matches throughout the collection.

We model this requirement by introducing a *sparse model* for the correspondence. To this end, we relax condition (4) by demanding $\sum_\gamma g(\gamma) = 1$ over $\check{\Gamma}$. This requirement gives us an interpretation of g as a discrete probability distribution over the space of all multi-way matches. Importantly, the L^1-like constraint on g has a *sparsity-promoting* effect on the solution, hence modeling partiality.

Unfortunately, directly minimizing a problem of the form given in Eq. (6) subject to $\sum_\gamma g(\gamma) = 1$ would yield trivial solutions. Specifically, we can characterize the global minimizers by: $g(\gamma) = 1$ for $\gamma = \gamma^\star$ and $g(\gamma) = 0$ otherwise, where $\gamma^\star$ is taken to be any $\gamma \in \check{\Gamma}$. This amounts to concentrating the whole mass of g into one single multi-way match, as illustrated in the inset figure.



We sidestep this issue by passing to a maximization problem. Suppose we are given, as opposed to the cost ε, a *similarity* function $s : \check{\Gamma} \times \check{\Gamma} \to \mathbb{R}_+$ measuring the extent to which two given multi-way matches preserve pairwise distances. A possible choice is given by the Gaussian score:

$$s(\gamma,\gamma') = e^{-\frac{1}{\mu^2}\varepsilon^2(\gamma,\gamma')}\,, \tag{7}$$

where $\mu^2 \in \mathbb{R}_+$ is the variance of s. Note that $s(\gamma,\gamma') = 0$ whenever $\varepsilon(\gamma,\gamma') = \infty$; that is, incompatible matches are assigned zero similarity. We get to the following optimization problem, which we consider throughout this paper:

Problem 1 (Partial multi-way correspondence). Given a collection of shapes $\mathcal{C}$, we seek a *partial* multi-way correspondence among them as a maximizer to:

$$\max_{g:\check{\Gamma}\to[0,1]} \sum_{\gamma,\gamma' \in \check{\Gamma}} s(\gamma,\gamma') g(\gamma) g(\gamma') \tag{8}$$

$$\text{s.t.} \quad \sum_{\gamma \in \check{\Gamma}} g(\gamma) = 1 \tag{9}$$

$$\bar{c}(\gamma,\gamma') g(\gamma) g(\gamma') = 0 \quad \forall\, \gamma,\gamma' \in \check{\Gamma}, \tag{10}$$

where we set $\bar{c}(\gamma,\gamma') = 1$ if the two matches are incompatible, and $\bar{c}(\gamma,\gamma') = 0$ otherwise. Eq. (10) ensures that incompatible matches will not appear in any local optimum.

The transition to a maximization problem has a regularizing effect on its optima, as there are no trivial maximizers meeting the constraints in this case.

Remark 2 Any local solution to Problem 1 satisfies the key requirements of a multi-way correspondence: 1) it is always cycle-consistent (by construction of $\check{\Gamma}$); 2) shape points are activated at most once by the correspondence (by Eq. (10)); and 3) partial matches are allowed (by $\check{\Gamma}$ and Eq. (9)).

A note on symmetries. In case the shapes in the collection carry bilateral symmetries, mapping either side would in principle yield the same optimum for Problem 1. In this paper we deem correct such symmetric solutions as long as they remain consistent across all pairs of shapes (see Fig. 4).

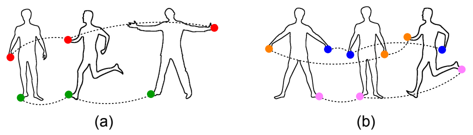


Figure 4: (a) *Incorrect correspondence due to inconsistent handling of the symmetry.* (b) *Even if the solution is not orientation-preserving, symmetries are treated consistently.*

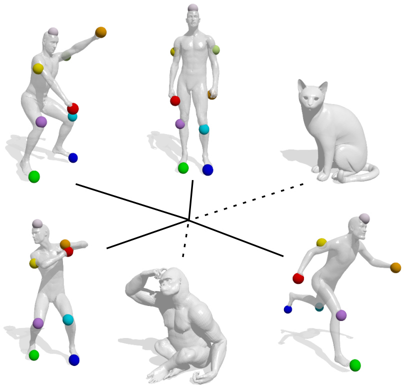


Figure 5: *Outlier shapes are automatically excluded by our approach, as they do not find support from the other shapes in the collection. Note that the human shapes appearing in this example come from different datasets (TOSCA, SCAPE, SHREC'14). A subset of all matches is shown.*

4. Optimization

Problem 1 is a non-convex quadratic program with $O(|\check{\Gamma}|)$ variables; as such, it is in general very difficult to solve and to give guarantees on the optimality of the solution. In this Section we develop an efficient strategy to get good local solutions to this problem. The general strategy is to decompose it into two sub-problems: a robust process to get good match candidates (Section 4.1), and a restriction of the original problem to the reduced feasible set (Section 4.2).

4.1. First sub-problem (reducing the feasible set)

The first sub-problem is aimed at reducing the size of the feasible set $\check{\Gamma}$ to a smaller subset of "stable" candidates $\Lambda \subset \check{\Gamma}$. Then, we will directly optimize Problem 1 over the reduced feasible set Λ.

Outline. The general insight behind our formulation is that, given a collection of shapes, it is relatively easy and inexpensive to solve for one *single* multi-way match between them. Specifically, the idea is to seek for a multi-way match $\gamma \in \check{\Gamma}$ that maximizes a measure of point-wise similarity across several shapes, hence taking advantage of the stability induced by the whole shape collection. The final goal is to keep in the feasible set only multi-way matches maximizing this measure of similarity, since they are expected to be accurate and stable against outliers, as shown in Fig. 5.

This problem can be formulated as a series of quadratic programs with sparsity constraints (Eq. (11)), each yielding a multi-way match $\gamma \in \Lambda$. Note that mapping constraints are imposed such that *only* cycle-consistent matches are allowed to be local optima.

Solving for a single multi-way match. Assume for simplicity that $|S_i| = N$ for all $i = 1, \ldots, n$. Further, let us be given a point-wise similarity function $\tau : S_k \times S_\ell \to \mathbb{R}_+$, measuring the similarity of some descriptor defined at shape points (an example is given in Section 5.1). Note that this function is not the same as the one defined in Eq. (7), which instead measures the similarity between multi-way matches.

We introduce the vector $\mathbf{x} \in [0,1]^{nN}$, representing a probability distribution over all points in $\bigcup_i S_i$. Then, consider the L^1-regularized non-convex quadratic program:

$$\max_{\mathbf{x} \geq 0} \mathbf{x}^\top \mathbf{A} \mathbf{x} \quad \text{s.t. } \mathbf{x}^\top \mathbf{1} = 1 \,. \tag{11}$$

Here, matrix $\mathbf{A}$ is a symmetric similarity matrix:

$$\mathbf{A} = \begin{pmatrix} \mathbf{0} & \mathbf{S}^{1,2} & \cdots & \mathbf{S}^{1,n} \\ \mathbf{S}^{1,2} & \mathbf{0} & \cdots & \cdots \\ \vdots & \vdots & \mathbf{0} & \mathbf{S}^{n-1,n} \\ \mathbf{S}^{1,n} & \vdots & \mathbf{S}^{n-1,n} & \mathbf{0} \end{pmatrix}, \tag{12}$$

where each symmetric block $\mathbf{S}^{k,\ell} \in \mathbb{R}^{N \times N}$ contains the similarity values between the points in S_k and S_ℓ, according to function τ. The reason for the zero blocks along the diagonal will become clear with Theorem 1. Note that the matrix above is not related to the block matrix appearing in [HG13, CGH14], which instead represents a collection of pairwise maps (ideally permutations).

The key result of this Section is that the support of any local maximizer to the problem above (*i.e.*, the set of points for which $\mathbf{x}_i \neq 0$) is guaranteed to be a *single* partial multi-way match $\gamma \in \check{\Gamma}$ between the shapes in the collection, as we state in the following theorem.

Theorem 1 Let $\mathbf{x}$ be a strict local maximizer of problem (11), where $\mathbf{A} = \mathbf{A}^\top$ and $\mathbf{A}_{ii} = 0$ for all $i = 1, \ldots, nN$. Then, $\mathbf{A}_{ij} > 0$ for all i, j such that $\mathbf{x}_i \neq 0$, $\mathbf{x}_j \neq 0$.

Proof. See the Appendix.

According to Thm. 1, local solutions to (11) cannot simultaneously activate points with zero similarity. This gives us a powerful means to restrict feasibility to solutions that activate *at most* one point per shape: It is sufficient to set $\mathbf{A}_{ij} = 0$ whenever indices i and j correspond to points on the same shape, *i.e.*, matrix $\mathbf{A}$ must have zero blocks on the diagonal.

Remark 3 Since local solutions to problem (11) are guaranteed to be multi-way matches, they are always cycle-consistent by definition.

A series of quadratic problems. Clearly, in order to construct the reduced set $\Lambda \subset \check{\Gamma}$ we need a way to enumerate the local optima of problem (11). We do so by solving a sequence of problems of this form, each with a different data matrix (12). Specifically, in each problem we compute similarities from a reference descriptor (or "query") to all shape points, and we discard all dissimilar points.

Suppose we are given a collection Q of queries to compare against. A family of problems of the form (11) can

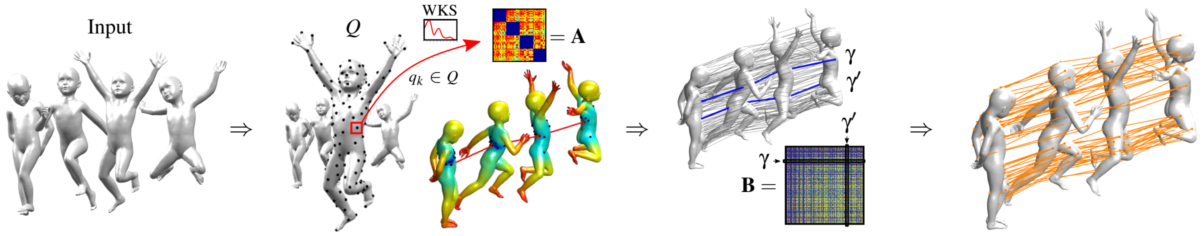


Figure 6: *Our matching pipeline. First sub-problem (from left): Given a collection of shapes as input, a set Q of queries are generated (*e.g.*, by farthest point sampling in the joint WKS space); we then compute distance maps (shown here as heat maps over the shapes) in descriptor space from each shape point to each query $q_k \in Q$, and keep the vertices having distance smaller than a threshold; finally, a single multi-way match is extracted by solving problem (11). Second sub-problem: The multi-way matches extracted by iterating the previous step are compared using a measure of metric distortion; the final solution (in orange) is obtained by solving problem (13) over the reduced feasible set.*

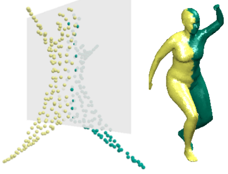
then be generated as follows. Given a query $q_k \in Q$, for each shape $S_i \in \mathcal{C}$ we only consider the vertices $p \in S_i$ such that $\tau(p, q_k) > \xi$ for some threshold $\xi > 0$. In other words, each query selects a different subset of vertices from each shape; since we can generate and solve as many problems (11) as there are queries q_k, we can proceed constructively and store each solution in our reduced feasible set Λ, which will have size $|\Lambda| = |Q|$.

Note that each of these problems will be quite small, since the number N' of shape points that are similar to each query is significantly smaller than the total number of points N. We refer to Sec. 5.5 for an empirical evaluation. We also note that this approach is different from previous approaches which require pairwise maps as input or which require geometrically consistent samples to be pre-selected across the shapes [NBCW*11, HZG*12, HG13, SY14, CGH14].

Example. Suppose we are given a point descriptor function $f : \bigcup_i S_i \to \mathbb{R}^m$, providing an embedding of all shapes in $\mathbb{R}^m$. The query set Q can be defined implicitly by a k-means clustering or by farthest point sampling directly in $\mathrm{Im} f$.

Numerical solution. It is worthwhile to note that problems of this form have a natural interpretation from the point of view of evolutionary game theory [ARBTP09, RBA*12]. We leverage this connection by adopting the infection-immunization dynamics algorithm [RBB11], an efficient local optimizer with convergence guarantees that exploits the specific structure of problem (11).

Symmetries. In order to favor symmetry-consistent solutions (Fig. 4), we assume to be given left-right maps for the shapes in the collection, *i.e.*, labelings $f : S \to \{\mathsf{L}, \mathsf{R}\}$ associating each shape point to either side. The maps are then used to augment the shape descriptors. While there are robust approaches to perform this task [LKF12], for our purposes it is enough to have a rough estimate so as to avoid obviously inconsistent solutions. We do so by looking for approximate rigid symmetries [PSG*06] on a multidimensional scaling in $\mathbb{R}^3$ of a few farthest samples per shape. An example of such procedure is shown in the inset figure. Note that, compared to existing methods, this is a simpler requirement than having input pairwise maps.



4.2. Second sub-problem (correspondence)

We are now ready to solve a smaller version of Problem 1 by replacing $\tilde{\Gamma}$ with the reduced set Λ. We proceed by directly rewriting the problem in matrix notation.

Suppose we solved $|\Lambda| = M$ instances (one per query) of problem (11), hence we have partial multi-way matches γ_i for $i = 1, \dots, M$ at our disposal. We can now compose the similarity terms $s(\gamma_i, \gamma_j)$ into a similarity matrix $\mathbf{B} \in \mathbb{R}_+^{M \times M}$ such that $\mathbf{B}_{ij} = \mathbf{B}_{ji} = s(\gamma_i, \gamma_j)$, and we set $\mathbf{B}_{ii} = 0$ for all $i = 1, \dots, M$ by Theorem 1. The correspondence function g can simply be represented by a vector $\mathbf{g} \in [0,1]^M$. Similarly to the previous case, we arrive at the quadratic program:

$$\max_{\mathbf{g} \geq 0} \mathbf{g}^\top \mathbf{B} \mathbf{g} \quad \text{s.t.} \quad \mathbf{g}^\top \mathbf{1} = 1\,. \qquad (13)$$

Note that the mapping constraints (10), which impose that incompatible matches cannot be part of the final solution, are already incorporated in the data matrix $\mathbf{B}$. This is because we set $\mathbf{B}_{ij} = 0$ whenever γ_i and γ_j are incompatible (by Eq. (7)).

A problem of this form for the simple case of two shapes was previously considered in [RBA*12]. Local solutions to (13) (obtained again with [RBB11]) will be accurate, although sparse. However, since the candidate set Λ is likely to contain good match hypotheses due to the previous optimization, there is hope to elicit a larger correspondence from it. To this end, we consider three simple approaches:

Grouped sparse. Following [RBTP09, ART12], we proceed by iteratively solving updated versions of problem (13). Whenever a local optimum is reached, the matches resulting

```
Data: Shape collection $\mathcal{C}$ of $n$ shapes
Result: Partial multi-way correspondence $\Gamma$ among the
        shapes in $\mathcal{C}$
pre-processing (Sec. 5);
$Q \leftarrow$ generated as in the Example of Sec. 4.1;
$\Lambda \leftarrow \emptyset$;
forall the $q \in Q$ do
    find points $p \in S_i$ s.t. $\tau(p,q) > \xi$ for $i = 1,\ldots,n$;
    construct similarity matrix $\mathbf{A}$ as in Eq. (12);
    $\mathbf{x} \leftarrow$ solve problem (11) using [RBB11];
    $\gamma \leftarrow$ support of $\mathbf{x}$;
    update $\Lambda \leftarrow \Lambda \cup \{\gamma\}$;
end
construct similarity matrix $\mathbf{B}$ as in Sec. 4.2;
$\mathbf{g} \leftarrow$ solve problem (13) "grouped" as in Sec. 4.2;
$\Gamma \leftarrow$ support of $\mathbf{g}$;
```

Algorithm 1: Full pipeline of our method for consistent partial matching of shape collections. Detailed parameter values are given in Section 5.

from the optimizer $\mathbf{g}$ are stored, and the data matrix is modified by setting $\mathbf{B}_{i\star} = \mathbf{B}_{\star i} = 0$ for all i such that $\mathbf{g}_i \neq 0$. By Theorem 1, this amounts to reducing the feasible set to the remaining candidates in Λ. The iterations stop when the objective $\mathbf{g}^\top \mathbf{B}\mathbf{g}$ falls below a certain threshold.

Spectral relaxation. A different way to approach the problem consists in replacing the L^1 constraint $\mathbf{g}^\top \mathbf{1} = 1$ by a L^2 counterpart $\mathbf{g}^\top \mathbf{g} = 1$. This type of constraint acts as a Tikhonov regularizer, which tends to yield denser solutions for this kind of problems. A *global* optimum can then be computed by Rayleigh's ratio as the principal eigenvector of $\mathbf{B}$. This comes at the price of sacrificing the mapping constraints guaranteed by Theorem 1, which must be imposed by a post-processing of the obtained solution [LH05].

Elastic net. Finally, one may introduce a form of controllable sparsity into the problem by elastic net regularization [RTH*13]. In this case, the L^1 constraint is replaced by the convex combination $(1-\alpha)\mathbf{g}^\top \mathbf{1} + \alpha \mathbf{g}^\top \mathbf{g} = 1$, where parameter $\alpha \in [0,1]$ allows to transition smoothly from a formulation equivalent to (13) (hence sparse) to a purely spectral solution (denser).

In Fig. 7 we show a full comparison of the three alternatives on the TOSCA dataset, using the cumulative error measure defined in Sec. 5. Finally, the main steps describing our matching pipeline are given in Algorithm 1 and Fig. 6.

4.3. Complexity and scalability

We conclude the theoretical part with a complexity analysis of our method. Suppose our collection $\mathcal{C}$ is made of n shapes, each shape has N points, and M is the number of queries.

For a single query, computing the similarity matrix $\mathbf{A}$ takes $O((nN)^2)$ operations. In practice, since for each query we have $N' \ll N$, this is a fast operation of the order $O((nN')^2)$. Optimization of problem (11) using evolutionary dynamics [RBB11] is a $O(nN')$ step. The complexity of the first sub-problem (generation of Λ) is thus $O(M \cdot (nN')^2)$.

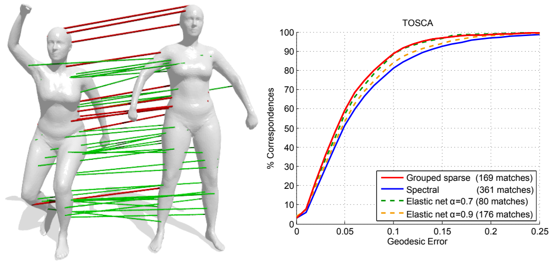


Figure 7: *Comparison between different numerical approaches to solve problem* (13). Left*: Matches visualized on two scanned shapes from the SHREC'14 dataset, extracted from a multi-way correspondence of length 7; the spectral relaxation yields 43 matches (in green), while the elastic net with $\alpha = 0.7$ only 11 matches (in red), although less noisy.* Right*: Quantitative comparison on the entire TOSCA dataset; the iterated L^1 approach provides the best combination of size and accuracy.*

Next, constructing matrix $\mathbf{B}$ is a $O(M^2)$ operation; this also involves computing geodesic distances among all points in Λ, which can be done efficiently via fast marching [WDB*08]. Since optimizing problem (13) is a $O(M)$ process, the overall complexity of the second sub-problem is $O(M^2)$. Note that in all our experiments we have once again $M \ll nN$, hence this step of the pipeline is typically very fast. We refer to Sec. 5.5 for an experimental evaluation.

5. Experimental results and applications

We performed a wide range of experiments on several benchmarks, namely: TOSCA [BBK08], SCAPE [ASK*05], KIDS [RRBW*14], and SHREC'14 [PSR*14]. These datasets consist of multiple classes of nearly-isometric shapes, with some intra-class variation in the case of KIDS and SHREC'14. All datasets with the exception of SHREC'14 come with ground-truth correspondences within each category. In all the experiments, we ran our matching algorithm using $M = 500$ queries in descriptor space. Parameter ξ was chosen as the 10th percentile of the descriptor distances to each query; the iterative process for solving problem (13) was stopped when the energy fell below 0.5. [†]

Pre-processing. WKS descriptors [ASC11] are precomputed for all the meshes. We rescale each shape by the square

[†] Code will be made available at: `http://vision.in.tum.de/members/rodola/code`

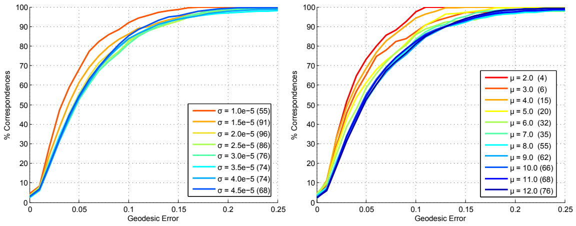


Figure 8: *Sensitivity experiments on a subset of TOSCA. Here we plot the error curves under different choices of point-wise similarity parameter* σ *(left), and metric distortion parameter* μ *(right). In both graphs, the resulting number of multi-way matches is reported in parentheses.*

root of the k-th Laplacian eigenvalue, with $k = 100$ for all shapes; this has the effect of normalizing the meshes to have similar surface area (by Weyl's law), and at the same time to yield comparable WKS. Where not specified otherwise, in our experiments we run the matching pipeline on $N = 300$ farthest samples per shape (using the Euclidean metric). This is done in order to avoid solutions that unduly aggregate in small regions. Note that we do *not* assume samples to be compatible across shapes as in [HG13], hence some local error in the computed matches is to be expected.

Error measure. We quantify the quality of the correspondence by using the same measure of error defined in [HG13]. Specifically, in our plots we show the percent of matches p_e which have geodesic error (*i.e.*, distance from the ground-truth) smaller than a threshold e. This cumulative distribution is computed and aggregated over all the pairwise matches induced by the obtained multi-way correspondences. The geodesic error is normalized by the square root of the area of each shape. As in [HG13], we also report values for $p_{0.16}$ and $p_{0.02}$, which respectively capture the global and local accuracy of the matching method.

5.1. Sensitivity analysis

The first set of experiments is aimed at analyzing the sensitivity of our method to different parametrizations. In order to reduce overfitting, these experiments are performed on a representative subset of the TOSCA dataset, consisting of the *victoria* (12 shapes) and *cat* (11 shapes) classes.

Point-wise similarity. We measure the similarity between points on different shapes by the similarity between their associated WKS descriptors. Each signature is computed on the shape samples using 100 eigenpairs, 100 energy levels and variance equal to 6.0 (default parameters as provided by the authors). Given two points $p \in S_k$ and $q \in S_\ell$, we define their similarity by the Gaussian weight

$$\tau(p,q) = e^{-\frac{1}{\sigma^2}\|WKS(p)-WKS(q)\|_2^2}\,. \tag{14}$$

In Fig. 8 (left) we plot the error curves under different choices of $\sigma \in \mathbb{R}$. Note that smaller values of σ tend to yield more accurate solutions. The choice of σ also has an effect, although not very pronounced, on the final number of multi-way matches (reported in parentheses).

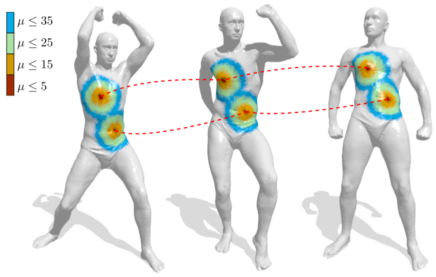


Figure 9: *Effect of parameter* μ *on the metric distortion term of Eq.* (7)*. Increasing the value of* μ *makes geometric validation more tolerant to distorted matches. In this real example on the SCAPE dataset, the colored regions show the admitted metric distortion for a pair of multi-way matches at different values of* μ*. An optimal value for this parameter can be chosen such that a prescribed metric distortion is not exceeded (*e.g.*, constrained to the orange area).*

Metric distortion. As described in Eq. (7), penalizing the metric distortion of a pair of multi-way matches is done by means of a control parameter μ. As shown in Fig. 8 (right), changing the value of μ allows to control the size/accuracy trade-off of the final correspondence: as μ is increased, distorted matches are tolerated and included in the solution. Further illustration of this behavior is given in Fig. 9, where we show how the *worst-case* metric distortion over the shape collection can be bounded by an appropriate choice of μ. The choice of this upper bound is ultimately driven by the application; *e.g.*, it makes sense in shape exploration applications (see Sec. 5.4) to require more accurate, although sparser matches in order to obtain a better clustering.

In a separate set of experiments, we investigate the effect of different similarity functions s. Namely, we consider both the Gaussian function of Eq. (7) as well as a modified version of it, given by replacing the worst-case cost of Eq. (3) by:

$$\varepsilon^2(\gamma,\gamma') = \sum_{\substack{p_k,p_\ell \in \gamma \\ p'_k,p'_\ell \in \gamma'}} |d_k(p_k,p'_k) - d_\ell(p_\ell,p'_\ell)|^2\,. \tag{15}$$

Following [RBA*12], we also include a relative (Lipschitz) notion of similarity in the comparison, defined as:

$$s(\gamma,\gamma') = \frac{\min_k d_k(p_k,p'_k)^\mu}{\max_k d_k(p_k,p'_k)^\mu}\,. \tag{16}$$

Since a fixed value of μ will in general scale differently in the three cases, each variant is parametrized so as to yield 30 multi-way matches on average. The results of this experiment are summarized in Table 1.

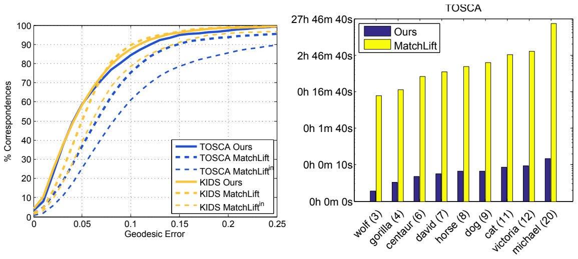


Figure 10: Left*: Comparison between our method and the state-of-the-art method of [CGH14] on the TOSCA and KIDS datasets; the quality of the input maps of MatchLift is also reported.* Right*: Comparison of execution times. In all the comparisons, the two methods generated a comparable amount of matches.*

5.2. Comparisons

We compare our method with MatchLift, the convex relaxation approach of Chen *et al.* [CGH14]. This method represents, to the best of our knowledge, the state of the art for this class of problems. In Fig. 10 we report the results on the TOSCA and KIDS datasets. Note that MatchLift did not previously appear in these benchmarks. For a fair comparison, the input pairwise maps were computed using the method described in [CGH14] with WKS as a descriptor. In the same figure we also report a runtime comparison of the two methods on collections of increasing size. In Table 2 we show additional comparisons with [HG13] and [HZG*12] on the TOSCA and SCAPE datasets. As a baseline for standard pairwise matching, we also include the method of [RBA*12] in the comparison. The results show that our method performs in line with the state of the art in most cases, with the additional theoretical guarantee of cycle-consistency and at a fraction of the computational time.

We remind the reader that all methods included in the comparisons, except for ours, require pairwise maps to be given as input (also evaluated in the comparisons), hence acting more like global regularizers rather than "pure" multiple shape matching methods. Also note that our method was not tuned to perform well in the comparisons, as our sensitivity analysis was only executed on a small subset of TOSCA.

	L^∞	L^2	Lipschitz
Local ($p_{0.02}$)	**26.81**	16.01	20.04
Global ($p_{0.16}$)	**96.21**	95.49	91.05

Table 1: *Comparison between different metric distortion measures on a subset of TOSCA. The best results (in bold) are obtained when we penalize the worst-case absolute metric error. Interestingly, there is no clear advantage in using a relative error as opposed to its absolute counterpart.*

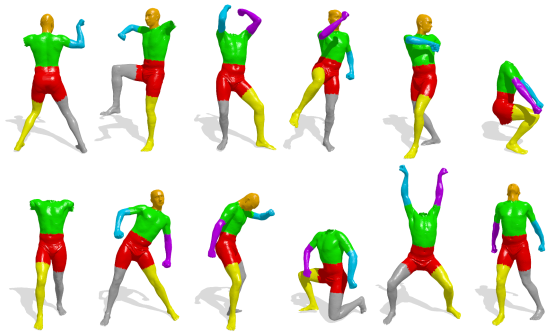


Figure 11: *Joint region matching on SCAPE (only a subset shown). The optimization process automatically excluded shape regions having incompatible segmentations with respect to the rest of the collection (*e.g.*, two segments per arm). Our pipeline took 2 sec. to produce these results.*

5.3. Region matching

Our method can be trivially modified to work with region-wise rather than point-wise correspondences, assuming a (possibly noisy) segmentation is provided for the input shapes. The modification boils down to define a proper similarity measure among regions. To this end we use the simple Gaussian score of Eq. (14), where the cost term is replaced by the L^2 distance between the area-weighted average WKS of each region. Regions are computed by consensus segmentation [RRBC14], using the code provided by the authors.

Note that since most shapes typically contain only 5 to 15 regions, a *full* similarity matrix $\mathbf{A}$ can be constructed which encodes the pairwise similarities among all regions in the collection (*i.e.*, we do not need to define queries). We can then solve the resulting problem (11) iteratively, each time reducing the feasible set by removing solutions from the past iterates (this is done by putting rows and columns of $\mathbf{A}$ to zero, as per Theorem 1). In Fig. 11 we show some qualitative results produced by this simple procedure when applied to a noisy version of SCAPE, in which 10 random outlier shapes from TOSCA were introduced.

	Ours	[HG13]	[HG13]$^{\text{in}}$	[HZG*12]	[RBA*12]
TOSCA ($p_{0.16}$)	97.7	100	84.1	97.2	94.81
SCAPE ($p_{0.16}$)	95.9	99.1	83.2	99.3	91.10
TOSCA ($p_{0.02}$)	21.9	35.7	-	38.4	14.87
SCAPE ($p_{0.02}$)	50.6	42.1	-	44.4	10.29

Table 2: *Comparisons with other recent methods in terms of global ($p_{0.16}$) and local ($p_{0.02}$) accuracy. The $^{\text{in}}$ column reports the quality of the input maps [KLF11].*

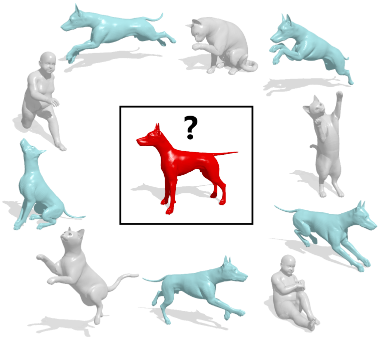


Figure 12: *An example of shape retrieval. The query shape is matched jointly to the shapes in the database, forming a cluster with the shapes from the same class.*

5.4. Other applications

Our approach is robust to the presence of outliers by design, and we can always extract an accurate solution as long as the outliers do not have a structure.

Consider the example in Fig. 1. As problem (11) is iteratively solved, the candidate set Λ is updated with matches that put the horse parts into correspondence, in addition to matches that only relate the human bodies. The subsequent optimization of (13) then extracts two intra-similar clusters of matches, one for each semantic group. In this case, it is clear that there is technically no reason to treat either of the two solutions as noise. Consider now a collection of shapes of a given class, which has been corrupted by introducing other shapes (Fig. 5). Since the extra-class objects fail to form stable matches with any other object in the collection, they will not appear in the final solution. This key feature of our framework suggests, among others, two applications:

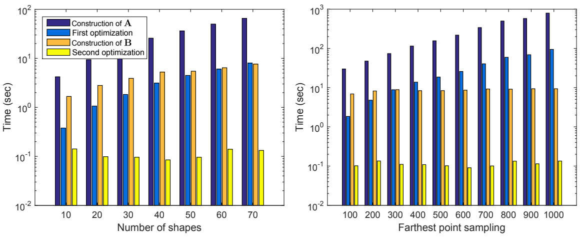


Figure 13: *Breakdown of our computational times over the SCAPE dataset.* Left*: Runtime as a function of a subset of shapes in the collection, with 300 samples per shape. The first and second optimizations refer to solving problems* (11) *and* (13) *respectively. The runtime for the first problem is accumulated over $M = 500$ queries.* Right*: Runtime as a function of farthest samples per shape, over the entire collection (71 shapes). Note the different scales among the two graphs.*

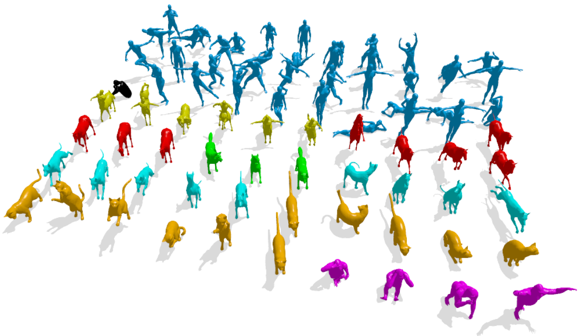


Figure 14: *An example of shape clustering of the TOSCA dataset, obtained by running the matching algorithm followed by extraction of connected components. Classes are encoded by color; note how all humans except for one* victoria *pose (in black) have been clustered together. Total running time is around 1 min. 30 sec.*

Shape exploration and clustering. Consider once again the example in Fig. 5, and suppose both outlier shapes actually belong to the same class. This scenario can be seen as an instance of structured noise – in fact, we now have *two* semantic classes forming intra-similar groups, and it would be desirable to separate them into disjoint clusters [KLM*12]. We do so by a simple iterative procedure: 1) run Algorithm 1 on the whole collection; 2) relabel the resulting multi-way correspondence into clusters, based *solely* on the shape indices; 3) remove the matched shapes from the collection and repeat. Note that the clustering step is especially efficient, as it boils down to detecting connected components in a graph where each node represents a shape, and an edge exists between two nodes whenever there exist (at least 3) matches connecting the respective shapes. Running this procedure on the TOSCA dataset gives the results reported in Fig. 14.

Shape retrieval. The approach described above can be directly applied to shape retrieval applications. Given a query shape $S_q \notin \mathcal{C}$, the task is to detect the subset $\mathcal{C}_q \subseteq \mathcal{C}$ containing shapes that belong to the same class as S_q. This can be done by seeking a multi-way correspondence on the augmented set $\mathcal{C} \cup \{S_q\}$, and by retaining the cluster of shapes that match to S_q in the final solution (see Fig. 12).

5.5. Runtime

One of the key advantages of our matching method lies in its computational efficiency. In Fig. 13 we show a breakdown of the runtimes across the whole pipeline. Observe that the first optimization can be easily parallelized (we used 7 cores in our tests), as it amounts to solving independent instances of problem (11), one problem per query.

Our method takes around 1 min. 30 sec. to match the *entire* SCAPE collection (71 shapes) when we use 300 samples per shape. We note that, while from the point of view of shape *retrieval* our method cannot compete with specialized approaches, the increased running time accounts for the correspondences we obtain across all the shapes in the collection. In this regard, shape retrieval can be seen more as a byproduct of our method than an application per se.

Further runtime comparisons with the method of [CGH14] are given in Fig. 10. All experiments were coded in Matlab/C++ and run on an Intel Core i7 4900MQ with 32GB memory, using publicly available code for [CGH14] and for the optimization step [RBB11].

6. Discussion and conclusions

In this paper we tackled the problem of consistent joint matching of shape collections. Differently from the dominant approaches, we considered a situation in which the collection is not equipped with input maps between the shape pairs. To deal with this challenging scenario, we modeled the problem as one of minimum distortion correspondence across the whole shape collection, while at the same time allowing outlier or partially similar shapes. We showed how to retrieve good local solutions to the resulting optimization problem by solving a sequence of quadratic programs in an efficient way – which in turn enabled favorable results in region matching and shape exploration applications.

Our approach does have a few shortcomings. First, since our method relies on the computation of geodesics, we require the shapes to have no significant missing parts, *i.e.*, shapes with large holes are not allowed. Second, while the sparse model allows to successfully deal with partial similarity at different levels, this partiality is not easily controllable and it might well be that incomplete matches are extracted even within outlier-free collections. An example of this is shown in Fig. 14, where one human shape (in black) was left unmatched by our method. This is related to our necessity to establish a similarity criterion that acts globally on the whole collection, hence driving longer, but less globally-similar matches to be cut out from the solution even if correct. Enforcing specific shapes to partake in the final solution is a possible direction of future work.

Acknowledgments

The authors wish to thank Mohamed Souiai, Zorah Lähner, Samuel Rota Bulò, Thomas Frerix, Luca Balsanelli, and Valentina Gualtieri for valuable discussions. ER is supported by an Alexander von Humboldt Fellowship.

References

[ARBTP09] ALBARELLI A., ROTA BULÒ S., TORSELLO A., PELILLO M.: Matching as a non-cooperative game. In *Proc. ICCV* (2009), pp. 1319–1326. 6, 12

[ART12] ALBARELLI A., RODOLÀ E., TORSELLO A.: Imposing semi-local geometric constraints for accurate correspondences selection in structure from motion: A game-theoretic perspective. *Int. J. Comput. Vision 97*, 1 (2012), 36–53. 6

[ASC11] AUBRY M., SCHLICKEWEI U., CREMERS D.: The wave kernel signature: A quantum mechanical approach to shape analysis. In *Proc. ICCV Workshops* (2011), pp. 1626–1633. 7

[ASK*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: Scape: Shape completion and animation of people. *ACM Trans. Graph. 24*, 3 (2005), 408–416. 7

[BBK08] BRONSTEIN A., BRONSTEIN M., KIMMEL R.: *Numerical Geometry of Non-Rigid Shapes*, 1 ed. Springer Publishing Company, Incorporated, 2008. 7

[CG14] CHEN Y., GOLDSMITH A.: Information recovery from pairwise measurements. In *Proc. ISIT* (2014), pp. 2012–2016. 2

[CGH14] CHEN Y., GUIBAS L., HUANG Q.-X.: Near-optimal joint object matching via convex relaxation. In *Proc. ICML* (2014), pp. 100–108. 2, 5, 6, 9, 11

[HG13] HUANG Q.-X., GUIBAS L.: Consistent shape maps via semidefinite programming. *Computer Graphics Forum 32*, 5 (2013), 177–186. 2, 5, 6, 8, 9

[HWG14] HUANG Q., WANG F., GUIBAS L.: Functional map networks for analyzing and exploring large shape collections. *ACM Trans. Graph. 33*, 4 (July 2014), 36:1–36:11. 2

[HZG*12] HUANG Q.-X., ZHANG G.-X., GAO L., HU S.-M., BUTSCHER A., GUIBAS L.: An optimization approach for extracting and encoding consistent maps in a shape collection. *ACM Trans. Graph. 31*, 6 (2012), 167:1–167:11. 2, 3, 6, 9

[KKBL15] KEZURER I., KOVALSKY S. Z., BASRI R., LIPMAN Y.: Tight relaxation of quadratic matching. *Computer Graphics Forum 34*, 5 (2015), 115–128. 2

[KLF11] KIM V. G., LIPMAN Y., FUNKHOUSER T.: Blended intrinsic maps. *ACM Trans. Graph. 30*, 4 (2011), 79:1–79:12. 1, 2, 9

[KLM*12] KIM V. G., LI W., MITRA N. J., DIVERDI S., FUNKHOUSER T.: Exploring collections of 3d models using fuzzy correspondences. *ACM Trans. Graph. 31*, 4 (July 2012), 54:1–54:11. 10

[LH05] LEORDEANU M., HEBERT M.: A spectral technique for correspondence problems using pairwise constraints. In *Proc. ICCV* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 1482–1489. 1, 7

[LKF12] LIU T., KIM V. G., FUNKHOUSER T.: Finding surface correspondences using symmetry axis curves. *Computer Graphics Forum 31*, 5 (2012). 6

[Mém11] MÉMOLI F.: Gromov-Wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics* (2011), 1–71. 3, 4

[NBCW*11] NGUYEN A., BEN-CHEN M., WELNICKA K., YE Y., GUIBAS L. J.: An optimization approach to improving collections of shape maps. *Computer Graphics Forum* (2011), 1481–1491. 2, 3, 6

[PKS13] PACHAURI D., KONDOR R., SINGH V.: Solving the multi-way matching problem by permutation synchronization. In *Proc. NIPS* (2013), pp. 1860–1868. 2, 3

[PSG*06] PODOLAK J., SHILANE P., GOLOVINSKIY A., RUSINKIEWICZ S., FUNKHOUSER T.: A planar-reflective symmetry transform for 3d shapes. *ACM Trans. Graph. 25*, 3 (2006), 549–559. 6

[PSR*14] PICKUP D., SUN X., ROSIN P. L., ET AL.: SHREC'14 track: Shape retrieval of non-rigid 3d human models. In *Proceedings of the 7th Eurographics workshop on 3D Object Retrieval* (2014), EG 3DOR'14, Eurographics Association. 7

[RBA*12] Rodolà E., Bronstein A., Albarelli A., Bergamasco F., Torsello A.: A game-theoretic approach to deformable shape matching. In *Proc. CVPR* (2012), pp. 182–189. 6, 8, 9

[RBB11] Rota Bulò S., Bomze I. M.: Infection and immunization: A new class of evolutionary game dynamics. *Games and Economic Behavior 71*, 1 (2011), 193–211. 6, 7, 11, 12

[RBTP09] Rota Bulò S., Torsello A., Pelillo M.: A game-theoretic approach to partial clique enumeration. *Image Vision Comput. 27*, 7 (2009), 911–922. 6

[RCB*16] Rodolà E., Cosmo L., Bronstein M. M., Torsello A., Cremers D.: Partial functional correspondence. *Computer Graphics Forum* (2016). 2

[RRBC14] Rodolà E., Rota Bulò S., Cremers D.: Robust region detection via consensus segmentation of deformable shapes. *Computer Graphics Forum 33*, 5 (2014), 97–106. 9

[RRBW*14] Rodolà E., Rota Bulò S., Windheuser T., Vestner M., Cremers D.: Dense non-rigid shape correspondence using random forests. In *Proc. CVPR* (2014), pp. 4177–4184. 7

[RTH*13] Rodolà E., Torsello A., Harada T., Kuniyoshi Y., Cremers D.: Elastic net constraints for shape matching. In *Proc. ICCV* (2013), pp. 1169–1176. 7

[SRS13] Solé-Ribalta A., Serratosa F.: Graduated assignment algorithm for multiple graph matching based on a common labeling. *Int. J. Patt. Recog. Art. Intel. 27*, 01 (2013). 2

[STCB07] Schmidt F. R., Töppe E., Cremers D., Boykov Y.: Intrinsic mean for semi-metrical shape retrieval via graph cuts. In *Proc. DAGM* (Berlin, Heidelberg, 2007), Springer-Verlag, pp. 446–455. 2, 3

[SY14] Sahillioğlu Y., Yemez Y.: Multiple shape correspondence by dynamic programming. *Computer Graphics Forum 33*, 7 (2014), 121–130. 2, 3, 6

[WDB*08] Weber O., Devir Y. S., Bronstein A. M., Bronstein M. M., Kimmel R.: Parallel algorithms for approximation of distance maps on parametric surfaces. *ACM Trans. Graph. 27*, 4 (2008), 104:1–104:16. 7

[YLL*14] Yan J., Li Y., Liu W., Zha H., Yang X., Chu S. M.: Graduated consistency-regularized optimization for multi-graph matching. In *Proc. ECCV* (2014), vol. 8689, pp. 407–422. 2, 3

[ZKP10] Zach C., Klopschitz M., Pollefeys M.: Disambiguating visual relations using loop constraints. In *Proc. CVPR* (2010), pp. 1426–1433. 1, 2, 3

Appendix

The appendix contains a proof to Theorem 1. The proof follows along the lines of, *e.g.*, [ARBTP09, RBB11], although by taking a pure optimization perspective as opposed to a game-theoretical one.

Problem. We consider the constrained quadratic problem:

$$\max_{\mathbf{x}\geq 0} \mathbf{x}^\top \mathbf{A}\mathbf{x} \quad \text{s.t. } \mathbf{x}^\top \mathbf{1} = 1\,, \tag{17}$$

where $\mathbf{x} \in [0,1]^n$, $\mathbf{A} = \mathbf{A}^\top$ and $\mathbf{A}_{ii} = 0$ for all $i = 1,\dots,n$.

First-order optimality conditions. A point $\mathbf{x}$ satisfies the Karush-Kuhn-Tucker (KKT) conditions for problem (17) if there exist real constants λ and $\mu_1,\dots,\mu_n$ with $\mu_i \geq 0$ for all $i = 1,\dots,n$ such that:

$$(\mathbf{A}\mathbf{x})_i - \lambda + \mu_i = 0\,, \tag{18}$$

and $\sum_i \mathbf{x}_i \mu_i = 0$. This latter condition further implies that $\mu_i = 0$ whenever $\mathbf{x}_i \neq 0$, since both quantities are nonnegative for all $i = 1,\dots,n$. Thus, we can rewrite the KKT conditions (18) as

$$(\mathbf{A}\mathbf{x})_i \begin{cases} = \lambda & \text{if } \mathbf{x}_i \neq 0 \\ \leq \lambda & \text{otherwise}\,, \end{cases} \tag{19}$$

for some $\lambda > 0$. We can easily see that

$$\mathbf{x}^\top \mathbf{A}\mathbf{x} = \sum_{i,j} \mathbf{x}_i \mathbf{x}_j \mathbf{A}_{ij} = \sum_{i\,:\,\mathbf{x}_i \neq 0} \mathbf{x}_i (\mathbf{A}\mathbf{x})_i = \sum_{i\,:\,\mathbf{x}_i \neq 0} \mathbf{x}_i \lambda = \lambda\,, \tag{20}$$

where the last equality follows from the constraint $\mathbf{x}^\top \mathbf{1} = 1$. Hence, a point $\mathbf{x}$ satisfies the KKT conditions if

$$\mathbf{x}^\top \mathbf{A}\mathbf{x} \geq (\mathbf{A}\mathbf{x})_i\,, \tag{21}$$

for all $i = 1,\dots,n$. This, in turn, implies $\mathbf{x}^\top \mathbf{A}\mathbf{x} \geq \mathbf{y}^\top \mathbf{A}\mathbf{x}$ for all $\mathbf{y}$ satisfying $\mathbf{y}^\top \mathbf{1} = 1$.

Second-order optimality conditions. A point $\mathbf{x}$ satisfies the second-order sufficiency conditions for *strict* local optimality if $\mathbf{x}$ is a KKT point, and if the Hessian of (17) is negative definite on the subspace $M(\mathbf{x})$, that is $\mathbf{z}^\top \mathbf{A}\mathbf{z} < 0$ for all $\mathbf{z} \in M(\mathbf{x})$, where

$$M(\mathbf{x}) = \{\mathbf{z} \in \mathbb{R}^n : \mathbf{z}^\top \mathbf{1} = 0 \text{ and } \mathbf{z}_j = 0 \text{ for all } j \in J\} \setminus \{\mathbf{0}\}\,, \tag{22}$$

and

$$J = \{j = 1,\dots,n : \mathbf{x}_j = 0, \mu_j > 0\}\,. \tag{23}$$

This condition can be more compactly rephrased as follows. Consider the set

$$U = \{\mathbf{y} \in \mathbb{R}^n : \mathbf{y}^\top \mathbf{A}\mathbf{x} = \mathbf{x}^\top \mathbf{A}\mathbf{x}, \mathbf{y} \neq \mathbf{x}\}\,, \tag{24}$$

and let $\mathbf{z} = \mathbf{y} - \mathbf{x}$ with $\mathbf{y} \in U$. Then $(\mathbf{x} + \mathbf{z}) \in U$, which means $\mathbf{z}^\top \mathbf{A}\mathbf{x} = 0$, thus $\mathbf{z} \in M(\mathbf{x})$; the converse is also true. Now we can write

$$(\mathbf{x} - \mathbf{y})^\top \mathbf{A}\mathbf{y} = -\mathbf{z}^\top \mathbf{A}(\mathbf{x} + \mathbf{z}) = -\mathbf{z}^\top \mathbf{A}\mathbf{z} > 0\,, \tag{25}$$

by the negative definitess condition $\mathbf{z}^\top \mathbf{A}\mathbf{z} < 0$. Thus, the second-order optimality condition can be succinctly phrased as:

$$\mathbf{x}^\top \mathbf{A}\mathbf{y} > \mathbf{y}^\top \mathbf{A}\mathbf{y} \tag{26}$$

whenever $\mathbf{y}^\top \mathbf{A}\mathbf{x} = \mathbf{x}^\top \mathbf{A}\mathbf{x}$ and $\mathbf{x} \neq \mathbf{y}$. Note that the converse can also be easily proven, *i.e.*, Eq. (26) holds if and only if $\mathbf{x}$ is a KKT point and $\mathbf{z}^\top \mathbf{A}\mathbf{z} < 0$ for all $\mathbf{z} \in M(\mathbf{x})$.

We are now ready to prove the following theorem.

Theorem 1. Let $\mathbf{x}$ be a strict local maximizer of problem (17), where $\mathbf{A} = \mathbf{A}^\top$ and $\mathbf{A}_{ii} = 0$ for all $i = 1,\dots,n$. Then, $\mathbf{A}_{ij} > 0$ for all i,j such that $\mathbf{x}_i \neq 0$, $\mathbf{x}_j \neq 0$.

Proof Assume $\mathbf{A}_{ij} \leq 0$ for $i \neq j$ and $\mathbf{x}_i \neq 0$, $\mathbf{x}_j \neq 0$. Denote by $\mathbf{e}^i$ the i-th column of the identity matrix, and note that ${\mathbf{e}^i}^\top \mathbf{A}\mathbf{x} = (\mathbf{A}\mathbf{x})_i$ and ${\mathbf{e}^i}^\top \mathbf{A}\mathbf{e}^j = \mathbf{A}_{ij}$. Now let $\mathbf{y} = \delta(\mathbf{e}^i - \mathbf{e}^j) + \mathbf{x}$, where $0 < \delta \leq \mathbf{x}_j$. We have

$$\mathbf{y}^\top \mathbf{A}\mathbf{x} = \delta(\mathbf{e}^i - \mathbf{e}^j)^\top \mathbf{A}\mathbf{x} + \mathbf{x}^\top \mathbf{A}\mathbf{x} = \mathbf{x}^\top \mathbf{A}\mathbf{x}\,, \tag{27}$$

where we used the fact that $(\mathbf{e}^i - \mathbf{e}^j)^\top \mathbf{A}\mathbf{x} = (\mathbf{A}\mathbf{x})_i - (\mathbf{A}\mathbf{x})_j = \lambda - \lambda = 0$ by Eq. (19). However,

$$\begin{aligned}(\mathbf{x} - \mathbf{y})^\top \mathbf{A}\mathbf{y} &= -\delta(\mathbf{e}^i - \mathbf{e}^j)^\top \mathbf{A}[\mathbf{x} + \delta(\mathbf{e}^i - \mathbf{e}^j)] && (28)\\ &= -\delta^2(\mathbf{e}^i - \mathbf{e}^j)^\top \mathbf{A}(\mathbf{e}^i - \mathbf{e}^j) && (29)\\ &= -\delta^2(\mathbf{A}_{ii} + \mathbf{A}_{jj} - 2\mathbf{A}_{ij}) && (30)\\ &= 2\delta^2 \mathbf{A}_{ij} \leq 0\,, && (31)\end{aligned}$$

which implies $\mathbf{x}^\top \mathbf{A}\mathbf{y} \leq \mathbf{y}^\top \mathbf{A}\mathbf{y}$. In other words, we have constructed a $\mathbf{y}$ for which the second-order condition (26) does not hold when $\mathbf{y}^\top \mathbf{A}\mathbf{x} = \mathbf{x}^\top \mathbf{A}\mathbf{x}$, contradicting the assumption that $\mathbf{x}$ is a strict local maximizer. □