

# Consistent Shape Maps via Semidefinite Programming\*

Qi-Xing Huang and Leonidas Guibas  
Computer Science Department, Stanford University, Stanford, CA

October 15, 2013

## Abstract

Recent advances in shape matching have shown that jointly optimizing the maps among the shapes in a collection can lead to significant improvements when compared to estimating maps between pairs of shapes in isolation. These methods typically invoke a cycle-consistency criterion — the fact that compositions of maps along a cycle of shapes should approximate the identity map. This condition regularizes the network and allows for the correction of errors and imperfections in individual maps. In particular, it encourages the estimation of maps between dissimilar shapes by compositions of maps along a path of more similar shapes.

In this paper, we introduce a novel approach for obtaining consistent shape maps in a collection that formulates the cycle-consistency constraint as the solution to a semidefinite program (SDP). The proposed approach is based on the observation that, if the ground truth maps between the shapes are cycle-consistent, then the matrix that stores all pair-wise maps in blocks is low-rank and positive semidefinite. Motivated by recent advances in techniques for low-rank matrix recovery via semidefinite programming, we formulate the problem of estimating cycle-consistent maps as finding the closest positive semidefinite matrix to an input matrix that stores all the initial maps. By analyzing the Karush-Kuhn-Tucker (KKT) optimality condition of this program, we derive theoretical guarantees for the proposed algorithm, ensuring the correctness of the recovery when the errors in the inputs maps do not exceed certain thresholds. Besides this theoretical guarantee, experimental results on benchmark datasets show that the proposed approach outperforms state-of-the-art multiple shape matching methods.

## 1 Introduction

Recently, there has been growing interest in jointly matching many shapes [NBCW\*11, KLM\*12, HZG\*12], via approaches which aim at aggregating information from multiple shapes to improve the maps computed between pairs of shapes in isolation — for brevity we will refer to this class of problems as “multiple shape matching.” Existing approaches typically view the shape collection as a graph whose nodes are the shapes and whose edges are decorated with maps between the shapes. If the maps express a consistent understanding of “what is the same” across the collection, then a cycle-consistency criterion must be met, i.e., compositions of maps along cycles of shapes in the graph should approximate the identity map. This allows us, for example, to replace incorrect maps between pairs of dissimilar shapes by compositions of correct maps along paths of more similar shapes, or to fix partial map errors. Although these methods have shown great potential in practice, they typically lack theoretical guarantees — it is not known under what conditions the ground truth maps can be recovered.

In this paper, we introduce a novel approach with both practical applicability and theoretical justification. We consider the problem setting where the input consists of a collection of shapes, represented as discrete metric spaces (i.e., points with pair-wise distances), and a collection of maps computed between (a connected subset of the) pairs of shapes. The output consists of new maps between all pairs of shapes that are (i)

---

\*This manuscript is the author version of the paper: *Qixing Huang and Leonidas Guibas, Consistent Shape Maps via Semidefinite Programming, Proc. Eurographics Symposium on Geometry Processing (SGP), 2013.*

cycle-consistent and (ii) close to the original input maps. The key idea of the proposed approach is to first formulate the multiple shape matching problem in a constrained optimization framework, and then to find an effective convex optimization relaxation for its solution. Computationally, we solve this convex optimization relaxation to obtain an approximate solution, which is then rounded into a feasible solution to the original problem. Theoretically, we analyze the optimality condition of the relaxed convex problem to obtain exact recovery conditions. In a broader picture, the proposed approach follows the general methodologies of compressive sensing [CT05] and low-rank matrix recovery [CR09,CLMW11], which analyze and solve difficult optimization problems using their convex relaxations.

It is quite challenging to convert the multiple shape matching problem into a constrained optimization program, which also admits a tight convex relaxation. The proposed approach is based on establishing the equivalence between cycle-consistent maps and the semidefiniteness of a binary matrix that stores all pairs of maps in blocks. This allows us to convert the multiple shape matching problem into a binary semidefinite program, which can be effectively solved via its convex relaxation (i.e., binary variables are relaxed into real variables between 0 and 1). We analyze the Karush-Kuhn-Tucker (KKT) optimality condition of this program, and derive upper bounds on the percentage of incorrect correspondences in the input maps so that the ground truth maps are recovered by solving the convex relaxation.

We also evaluate the proposed approach on benchmark datasets TOSCA [BBK08], SCAPE [ASK\*05] and SHREC07 [GBP07,KLF11]. Experimental results show that the new method outperforms state-of-art multiple shape matching approaches [NBCW\*11,KLM\*12,HZG\*12] on these datasets.

## 1.1 Related work

The problem of multiple shape matching appears as a crucial step in many scientific problems including fusing partially overlapping range scans [Hub02], assembling fractured surfaces [HFG\*06] and structure from motion [ZKP10]. As it is beyond the scope of this paper to review all existing works, we only discuss approaches that can be applied to 3D shapes. To facilitate the discussion, we employ the terminology of a *model graph* [Hub02], whose vertices and edges represent shapes and maps between pairs of shapes, respectively.

Almost all existing approaches follow the general methodology of applying the cycle-consistency criterion to improve maps computed between pairs of shapes in isolation. Depending on how the cycle-consistency criterion is applied, existing approaches fall into three categories. The first category of methods [Hub02, HFG\*06] utilizes the fact that a collection of cycle-consistent maps can be generated from maps associated with a spanning tree in the model graph. However, as the number of spanning trees in a graph is usually exponential in the number of vertices, these approaches only optimize the spanning trees greedily and/or locally, often resulting in sub-optimal solutions.

The second category of approaches [ZKP10,NBCW\*11] applies constrained optimization to select cycle-consistent maps. These approaches are typically formulated so that the objective functions encode the score of selected maps, and the constraints enforce the consistency of selected maps along cycles, i.e., if the composite map along a cycle deviates from the identity map, then at least one map along this cycle is incorrect. Once bad maps are identified, they can be replaced by compositions of other maps. The major advantage of these methods is that the correct maps are determined globally, leading to better performance than the first category of approaches. However, their success relies on the assumption that correct maps are dominant in the model graph so that the small number of bad maps can be identified through their participation in many bad cycles. In addition, as there are exponential number of cycles, how to effectively sample cycles remains an open question.

Besides the limitations described above, a shared limitation of the first two categories of approaches is that the resulting maps are generated by composing input maps. In other words, these approaches require the full correctness of (a subset of) input maps, and would certainly fail if all input maps are partially incorrect but the majority correspondences in each map are correct. The third category of methods [KLM\*12,HZG\*12] addresses this limitation by composing correspondences to generate maps. These approaches encode maps between pairs of shapes using a big correspondence matrix, and apply spectral techniques to improve the

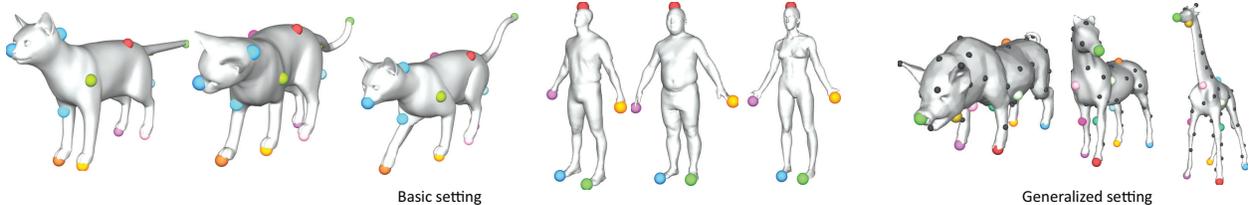


Figure 1: Basic structure of the output maps. (Left) When the sample distributions are consistent, e.g., the input shapes are similar, or samples are given by common shape extremals, the proposed approach outputs one-to-one maps between all the samples. (Right) Otherwise, the approach selects samples and outputs one-to-one maps between the selected samples.

map consistency. Although producing good performance in practice, they are limited in the fact there is no theoretical guarantee, i.e., it is not known under what conditions these approaches can recover the ground truth maps. In contrast, the proposed approach admits precise theoretical guarantees. Experimentally, it also modestly outperforms [KLM\*12] and [HZG\*12] on benchmark datasets.

In independent work [WS12], Wang and Singer analyzed a semidefinite programming formulation to the problem of recovering consistent rotational transforms between rigid shapes. Their approach also recognizes the fact that the matrix that stores pair-wise rotational transforms in blocks is semidefinite. However, we consider a different problem in this paper, i.e., that of computing point-based maps, where the both the formulation of the problem and the analysis of the proposed algorithm are drastically different.

We remark that although the matrix we try to recover is low-rank, it is also sparse, reflecting the 1–1 nature of shape correspondences. This means standard low-rank matrix recovery techniques [CR09, CLMW11] do not apply well in our setting because they only work for dense matrices. The proposed approach is made possible because we utilize special structures present in the matrix — that the diagonal blocks are identity matrices and that off-diagonal blocks are doubly stochastic matrices.

## 2 Problem Statement and Overview

### 2.1 Terminology

**Shape.** Without loss of generality, we assume shapes are represented as discrete metric spaces, i.e., a shape  $S$  is given by a set of  $m$  samples and a distance matrix  $d_S(\cdot, \cdot)$  that describes all pair-wise distances between these samples. Given a triangular mesh  $M$ , we generate such a representation using the procedure described in [LF09], which first detects extremals of Gaussian curvatures, and then applies furthest point sampling to add the remaining samples. The distance matrix  $d_S(\cdot, \cdot)$  stores geodesic distances on  $M$ .

**Map.** A point-to-point map  $\phi : S \rightarrow S'$  from a source shape  $S$  to a target shape  $S'$  is equivalent to a set of correspondences of the form  $\{s, \phi(s) \mid s \in S_i\}$ , where each point on the source shape appears in exactly one correspondence. This map is one-to-one if in addition each point from the target shape appears in exactly one correspondence.

**Cycle-consistency.** Cycle-consistency refers to the fact that composite maps along cycles in the model graph are identity maps. As we compute maps between all pairs of shapes (complete graph), it is sufficient to only consider 1-cycles, 2-cycles and 3-cycles [NBCW\*11]:

**Definition 1.** Given a shape collection  $\mathcal{S} = \{S_1, \dots, S_n\}$  of  $n$  shapes where each shape consists of the same number of samples, we say a map collection  $\Phi = \{\phi_{ij} : S_i \rightarrow S_j \mid 1 \leq i, j \leq n\}$  of maps between all pairs of

shapes is cycle consistent if and only if the following equalities are satisfied:

$$\begin{aligned}
 \phi_{ii} &= id_{S_i}, & 1 \leq i \leq n, & & \text{(1-cycle)} \\
 \phi_{ji} \circ \phi_{ij} &= id_{S_i}, & 1 \leq i < j \leq n, & & \text{(2-cycle)} \\
 \phi_{ki} \circ \phi_{jk} \circ \phi_{ij} &= id_{S_i}, & 1 \leq i < j < k \leq n, & & \text{(3-cycle)}
 \end{aligned} \tag{1}$$

where  $id_{S_i}$  denotes the identity self-map on  $S_i$ .

## 2.2 Input and output

The input is given by a model graph  $\mathcal{G} = (\mathcal{S}, \mathcal{E})$ , where  $\mathcal{S} = \{S_1, \dots, S_n\}$  encodes the input shapes, and  $\mathcal{E}$  specifies a collection of point-to-point input maps  $\Phi^{\text{in}} = \{\phi_{ij}^{\text{in}} : S_i \rightarrow S_j \mid (i, j) \in \mathcal{E}\}$ . Without loss of generality, we assume  $\mathcal{E}$  is symmetric, i.e., if  $(i, j) \in \mathcal{E}$ , then  $(j, i) \in \mathcal{E}$ . As above, each shape is represented by a discrete metric space of  $m$  samples.

For the output, we would like to obtain cycle consistent one-to-one maps between all pairs of shapes. Based on how samples are distributed on different shapes, we consider a basic setting and a generalized setting (See Figure 1).

**Basic setting.** The basic setting addresses the case where sample distributions are consistent across shapes, e.g., when input shapes are quite similar to each other (e.g., TOSCA and SCAPE). Alternatively, this setting is also relevant when we focus on maps between extremal points which can be obtained consistently across the shapes (c.f., [SOG09]). In this case, the output consists of cycle-consistent maps between the input shapes  $\Phi^{\text{out}} = \{\phi_{ij} : S_i \rightarrow S_j \mid 1 \leq i < j \leq n\}$ .

**Generalized setting.** The generalized setting is used in the case, where input shapes exhibit larger variability (e.g., SHREC07), i.e., it is unlikely to obtain consistent sample distributions across all the shapes. In this case, the output consists of (i) a reduced shape  $S'_i \subset S_i$  of  $m_0 \leq m$  samples from each input shape and (ii) a set of cycle-consistent one-to-one maps  $\Phi^{\text{out}} = \{\phi_{ij} : S'_i \rightarrow S'_j \mid 1 \leq i < j \leq n\}$  between these reduced shapes. For all the experiments, we set  $m_0 = m/4$ . A theoretical analysis that supports this choice is provided in the supplemental material. To avoid selecting samples that unduly aggregate on certain areas of the the input shapes, we assume the selected samples on  $S_1$  are given by its first  $m_0$  samples, which are distributed uniformly on  $S_1$  due to the natural of furthest point sampling.

Note that when analyzing the algorithm, we mainly focus on the basic setting. This helps us to separate the effect of having incorrect input maps, the major focus of this paper, from the uncertainty introduced by having inconsistent sample distributions on the shapes.

## 2.3 Paper organization

The remainder of this paper is organized as follows. In Section 3, we establish the important relation between cycle-consistency and semidefiniteness. In Section 4, we present the constrained optimization formulation in the basic setting, and describe an optimization strategy using convex relaxation. In Section 5, we analyze this convex relaxation and provide conditions on the exact recovery of the ground truth maps. We extend the basic setting to the generalized setting in Section 6. In Section 7, we evaluate the performance of the proposed approach on benchmark datasets. Finally, we conclude the paper and discuss future directions in Section 8.

# 3 Cycle-Consistency and Semidefiniteness

The formulation of the proposed approach is rooted in the connection between cycle-consistent maps and the semidefiniteness of the binary matrix that stores these maps in blocks. In the following, we first introduce the matrix representation of maps. Then we describe the connection.

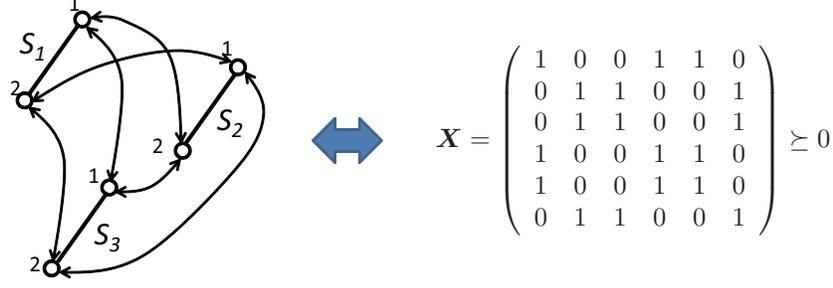


Figure 2: The equivalence between cycle-consistent maps and positive semidefiniteness of the big map collection matrix, storing these maps in blocks.

**Matrix representation of maps.** We present a point-to-point map  $\phi_{ij} : S_i \rightarrow S_j$  from shape  $S_i$  to  $S_j$  as a binary matrix  $\mathbf{X}_{ij} \in \{0, 1\}^{m \times m}$ , where  $X_{ij}(s, s') = 1$  if and only if  $(s, s') \in \phi_{ij}$ . It is clear that each row of  $\mathbf{X}_{ij}$  has exactly one entry equal to 1. In particular, if  $\phi_{ij}$  is a one-to-one map, then  $\mathbf{X}_{ij}$  is a permutation matrix. It is well known that a permutation matrix  $\mathbf{X}_{ij}$  is characterized by

$$\mathbf{X}_{ij}\mathbf{1} = \mathbf{1}, \quad \mathbf{X}_{ij}^T\mathbf{1} = \mathbf{1},$$

where  $\mathbf{1}$  represents the vector whose elements are 1. Note that in the matrix representation of maps, map composition becomes matrix multiplication:

$$\phi_{jk} \circ \phi_{ij} = \phi_{ik} \Leftrightarrow \mathbf{X}_{jk}\mathbf{X}_{ij} = \mathbf{X}_{ik}.$$

Consider a shape collection  $\mathcal{S} = \{S_1, \dots, S_n\}$ , where each shape consists of  $m$  samples. Given a complete collection of one-to-one maps  $\Phi = \{\phi_{ij}, 1 \leq i < j \leq n\}$ , we use a big binary matrix  $\mathbf{X} \in \{0, 1\}^{nm \times nm}$  to store these maps in blocks:

$$\mathbf{X} = \begin{pmatrix} \mathbf{I}_m & \mathbf{X}_{12} & \cdots & \mathbf{X}_{1n} \\ \mathbf{X}_{12}^T & \mathbf{I}_m & \cdots & \cdots \\ \vdots & \vdots & \mathbf{I}_m & \mathbf{X}_{(n-1),n} \\ \mathbf{X}_{1n}^T & \vdots & \mathbf{X}_{(n-1),n}^T & \mathbf{I}_m \end{pmatrix}.$$

Note that the diagonal blocks of  $\mathbf{X}$  are identity matrices, representing identity self-maps. Each block  $\mathbf{X}_{ji}, j > i$  is given by the transpose of  $\mathbf{X}_{ij}$ , representing the inverse map of  $\phi_{ij}$ . In other words,  $\mathbf{X}$  is symmetric. In the following, we will call  $\mathbf{X}$  the *map collection matrix* of map collection  $\Phi$ .

**Connection.** We now describe the connection between the cycle-consistency of  $\Phi$  and the low-rank and semidefinite property of  $\mathbf{X}$  (See Figure 2). Following a standard convention, we write  $\mathbf{X} \succeq 0$  to denote that  $\mathbf{X}$  is positive semidefinite.

**Proposition 1.** *The following three statements are equivalent:*

1.  $\Phi$  is cycle-consistent, i.e., it satisfies (1).
2. The rank of  $\mathbf{X}$  is  $m$ , and it can be factorized as  $\mathbf{X} = \mathbf{Y}_i^T \mathbf{Y}_i$  for one  $i \in \{1, \dots, n\}$ , where  $\mathbf{Y}_i$  is the matrix that stores all the maps from  $S_i$  in a row:  $\mathbf{Y}_i = (\mathbf{X}_{i1}, \dots, \mathbf{X}_{in})$ .
3.  $\mathbf{X}$  is positive semidefinite:  $\mathbf{X} \succeq 0$ .

**Proof:**  $1 \Leftrightarrow 2 : 1 \Rightarrow 2$  is straightforward due to the the matrix expression of cycle consistency. Suppose 2 is true. Without losing generality, we assume  $\mathbf{X} = \mathbf{Y}_1^T \mathbf{Y}_1$ . Then we have

$$\begin{aligned} \phi_{i1}\phi_{1i} &= id_{S_1}, \phi_{1i}\phi_{i1} = id_{S_i}, \quad 1 \leq i \leq n, \\ \phi_{ij} &= \phi_{1j} \circ \phi_{i1}, \quad 1 \leq i < j \leq n. \end{aligned}$$

It follows that  $\forall 1 \leq i < j \leq n$ ,

$$\phi_{ij} \circ \phi_{ji} = \phi_{1j} \circ \phi_{i1} \circ \phi_{1i} \circ \phi_{j1} = id_{S_j},$$

and  $\forall 1 \leq i < j < k \leq n$ ,

$$\phi_{ki} \circ \phi_{jk} \circ \phi_{ij} = \phi_{1i} \phi_{k1} \phi_{1k} \phi_{j1} \phi_{1j} \phi_{i1} = id_{S_i}.$$

$2 \Leftrightarrow 3 : 2 \Rightarrow 3$  is also straightforward as  $\forall \mathbf{z} \in \mathbb{R}^{nm}$ ,

$$\mathbf{z}^T \mathbf{X} \mathbf{z} = \mathbf{z}^T \mathbf{Y}_i^T \mathbf{Y}_i \mathbf{z} = \|\mathbf{Y}_i \mathbf{z}\|_{\mathcal{F}}^2 \geq 0.$$

To prove  $3 \Rightarrow 2$ , we require the following lemma

**Lemma 1.** *Suppose  $x \in \mathbb{R}$ , then*

$$A_{33}(x) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & x \\ 1 & x & 1 \end{pmatrix} \succeq 0 \quad \Leftrightarrow \quad x = 1.$$

*Proof:* The three eigenvalues of  $A_{33}(x)$  are given by  $\lambda_1 = 1 - x, \lambda_2, \lambda_3 = \frac{(x+2) \pm \sqrt{x^2+8}}{2}$ . Setting  $\lambda_i \geq 0$  yields  $x = 1$ .  $\square$

Let  $\mathbf{D} = \text{Diag}(\mathbf{I}_m, \mathbf{X}_{12}, \dots, \mathbf{X}_{1n}) \in \mathbb{R}^{nm \times nm}$  denote the block diagonal matrix that collects  $\mathbf{I}_m, \mathbf{X}_{12}, \dots, \mathbf{X}_{1n}$  in diagonal blocks. As  $\mathbf{X} \succeq 0$ , we have

$$\mathbf{X}' = \mathbf{D} \mathbf{X} \mathbf{D}^T = \begin{pmatrix} \mathbf{I}_m & \mathbf{I}_m & \cdots & \mathbf{I}_m \\ \mathbf{I}_m & \ddots & \mathbf{X}_{1i} \mathbf{X}_{ij} \mathbf{X}_{1j}^T & \vdots \\ \vdots & \mathbf{X}_{1j} \mathbf{X}_{ij}^T \mathbf{X}_{1i} & \ddots & \vdots \\ \mathbf{I}_m & \cdots & \cdots & \mathbf{I}_m \end{pmatrix} \succeq 0.$$

Note that any principal submatrix of a positive semidefinite matrix are positive semidefinite. It follows that the  $3 \times 3$  principal submatrix of both row and column indices  $(s, im + s, jm + s)$  is positive semidefinite:

$$A_{33}(X'_{ij}(s, s)) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & X'_{ij}(s, s) \\ 1 & X'_{ij}(s, s) & 1 \end{pmatrix} \succeq 0.$$

Using Lemma 1, we have  $X'_{ij}(s, s) = 1$ . Thus,  $\mathbf{X}'_{ij} = \mathbf{I}_m$ , which means  $\mathbf{X}_{ij} = \mathbf{X}_{1i}^T \mathbf{X}_{1j}$ . This completes the proof.  $\square$

Proposition 1 provides a simple representation of cycle-consistent maps, i.e., as a positive semidefinite matrix  $\mathbf{X} \in \mathbb{R}^{nm \times nm}$ , whose diagonal blocks are identity matrices, and whose off-diagonal blocks are permutation matrices. If we further relax the permutation matrix constraint so that each off-diagonal block is only a doubly stochastic matrix, i.e., each element of  $\mathbf{X}_{ij}$  may take real values between 0 and 1, and rows and columns of each block of  $\mathbf{X}$  sum to 1, then the cycle-consistency constraint becomes in fact convex. Note that this relaxation is considered optimal since the space of doubly stochastic matrices is the smallest convex set containing all permutation matrices.

Although Proposition 1 does not hold after this relaxation, we show in the following sections that when enforcing the relaxed constraints to optimize the L1 distance between  $\mathbf{X}$  and  $\mathbf{X}^{in}$  (i.e, the matrix that encodes the input maps), then there exist weak conditions under which we are able to recover the ground truth maps, which are cycle-consistent. This is partially due to the optimality of this relaxation.

## 4 Constrained Optimization and Convex Relaxation

In this section, we describe our constrained optimization formulation in the basic setting, i.e., the input is given by maps  $\Phi^{\text{in}} = \{\phi_{ij}^{\text{in}} : S_i \rightarrow S_j, (i, j) \in \mathcal{E}\}$ , and output consists of cycle-consistent maps  $\Phi^{\text{out}} = \{\phi_{ij} : S_i \rightarrow S_j, 1 \leq i < j \leq n\}$  between the input shapes. We first present the formulation in Section 4.1. Then we describe how to solve it via its convex relaxation in Section 4.2.

### 4.1 Formulation

**Variable parametrization and constraints.** We use a map collection matrix  $\mathbf{X} \in \{0, 1\}^{nm \times nm}$  to encode  $\Phi^{\text{out}}$ . Applying Proposition 1, we write down the following constraints that specify the cycle-consistency of  $\Phi^{\text{out}}$ :

$$\begin{aligned} \mathbf{X} &\in \{0, 1\}^{nm \times nm}, \mathbf{X} \succeq 0 \\ \mathbf{X}_{ii} &= I_m, & 1 \leq i \leq n \\ \mathbf{X}_{ij}\mathbf{1} &= \mathbf{1}, \mathbf{X}_{ij}^T\mathbf{1} = \mathbf{1}, & 1 \leq i < j \leq n \end{aligned} \quad (2)$$

**Objective function.** Following low-rank matrix recovery techniques [CR09, CLMW11], which evaluate the L1-norm between the input matrix and the recovered matrix, we define our objective function so that it sums the L1-norm between the output maps and input maps:

$$f_{\text{align}} = \sum_{(i,j) \in \mathcal{E}} \|\mathbf{X}_{ij}^{\text{in}} - \mathbf{X}_{ij}\|_1, \quad (3)$$

where  $\|\cdot\|_1$  denotes the elementwise L1-norm. As  $\mathbf{X}_{ij}^{\text{in}}$  are binary matrices, it turns out whenever the elements of  $\mathbf{X}_{ij}$  are between 0 and 1 (i.e., when they are relaxed), we can rewrite  $f_{\text{align}}$  as a linear function over  $\mathbf{X}$ :

$$\begin{aligned} f_{\text{align}} &= \sum_{(i,j) \in \mathcal{E}} \left( \sum_{X_{ij}^{\text{in}}(s,s')=0} X_{ij}(s,s') + \sum_{X_{ij}^{\text{in}}(s,s')=1} (1 - X_{ij}(s,s')) \right) \\ &= m|\mathcal{E}| + \langle \mathbf{e}\mathbf{e}^T - 2\mathbf{X}_{ij}^{\text{in}}, \mathbf{X}_{ij} \rangle, \end{aligned} \quad (4)$$

where  $m$  is the number of sample points per shape, and  $\langle A, B \rangle = \text{Tr}(A^T B)$  denotes matrix inner product.

**Constrained optimization.** Combining (2) and Equation (4), we formulate the following constrained optimization problem to recover cycle-consistent maps as:

$$\begin{aligned} \min_{\mathbf{X}} \quad & \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{e}\mathbf{e}^T - 2\mathbf{X}_{ij}^{\text{in}}, \mathbf{X}_{ij} \rangle \\ \text{s.t.} \quad & \\ \mathbf{X} &\in \{0, 1\}^{nm \times nm}, \mathbf{X} \succeq 0 \end{aligned} \quad (5)$$

$$\begin{aligned} \mathbf{X}_{ii} &= I_m, & 1 \leq i \leq n \\ \mathbf{X}_{ij}\mathbf{1} &= \mathbf{1}, \mathbf{X}_{ij}^T\mathbf{1} = \mathbf{1}, & 1 \leq i < j \leq n \end{aligned} \quad (6)$$

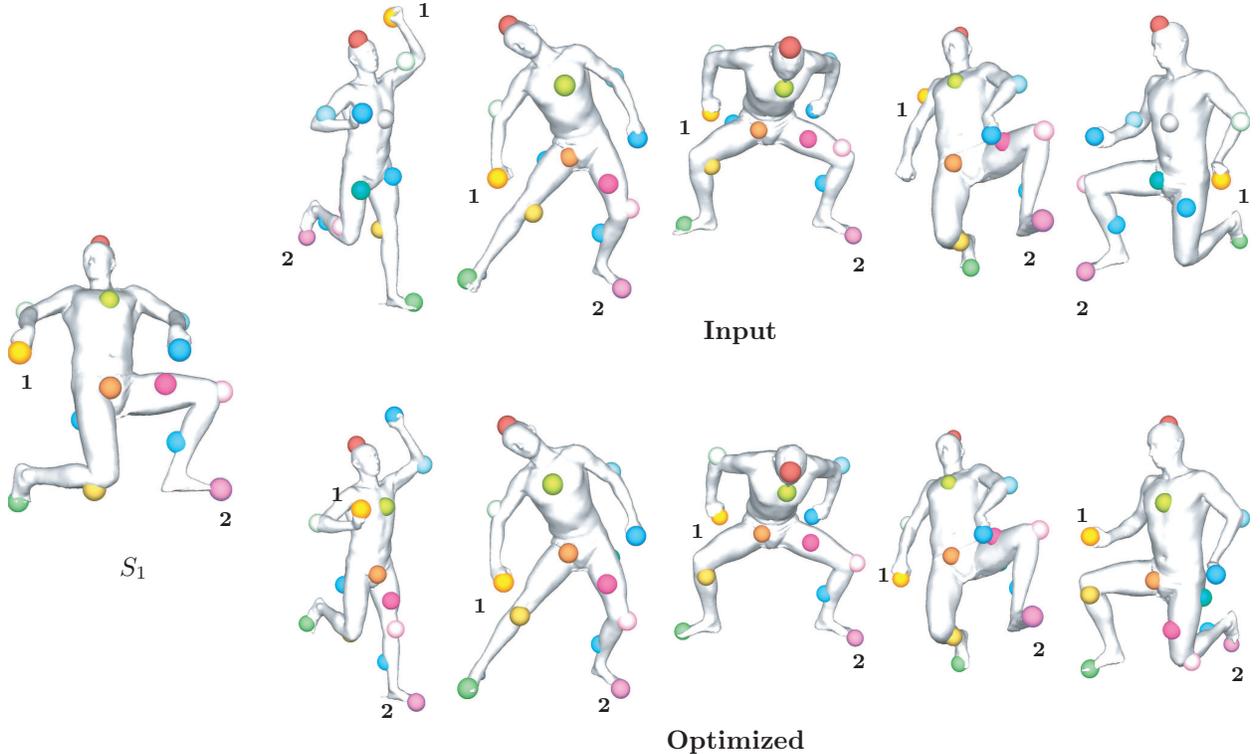


Figure 3: Comparison between the input maps and the optimized maps from a source shape to selected target shapes from the SCAPE dataset. Only correspondences of the first 16 samples are shown, for clarity.

## 4.2 Optimization via convex relaxation

**Convex relaxation.** We solve (6) by solving its convex relaxation, where we allow elements of  $\mathbf{X}$  to take real values between 0 and 1:

$$\begin{aligned}
 \min_{\mathbf{X}} \quad & \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{e}\mathbf{e}^T - 2\mathbf{X}_{ij}^{in}, \mathbf{X}_{ij} \rangle \\
 \text{s.t.} \quad & \mathbf{X} \geq 0, \mathbf{X} \leq \mathbf{1}, \\
 & \mathbf{X}_{ii} = I_m, \quad 1 \leq i \leq n \\
 & \mathbf{X}_{ij}\mathbf{1} = \mathbf{1}, \mathbf{X}_{ij}^T\mathbf{1} = \mathbf{1}, \quad 1 \leq i < j \leq n
 \end{aligned} \tag{7}$$

Note that the constraint  $\mathbf{X} \leq \mathbf{1}$  is removed since it can be derived from the fact that  $\mathbf{X} \geq 0$  and  $\mathbf{X}_{ij}\mathbf{1} = \mathbf{1}, 1 \leq i < j \leq n$ .

**Numerical optimization.** For numerical optimization, we employ the alternating direction method of multipliers technique (ADMM) [BPC\*11]. Our current implementation applies the ADMM method described in [WGY10] for solving semidefinite programs. Due to space constraint, we defer the explicit expressions Appendix B.

**Rounding strategy.** As  $\mathbf{X}$  can take real values in (7), it is possible that the optimal solution takes real values (the conditions that it returns integer solutions will be discussed in the next section). In this case, we propose to use a simple greedy rounding procedure to round real value solutions into integer solutions. Specifically, we solve a linear assignment problem to round each  $\mathbf{X}_{1i}$  into a permutation matrix  $\mathbf{X}_{1i}^{int}$ . The linear programming formulation [Sch86] is given by

$$\begin{aligned}
 \min_{\mathbf{X}_{1i}^{int}} \quad & \langle \mathbf{X}_{1i}^{int}, \mathbf{1} - \mathbf{X}_{1i} \rangle \\
 \text{s.t.} \quad & \mathbf{X}_{1i}^{int}\mathbf{1} = \mathbf{1}, \mathbf{X}_{1i}^{int^T}\mathbf{1} = \mathbf{1}, \mathbf{X}_{1i}^{int} \geq 0.
 \end{aligned} \tag{8}$$

After obtaining  $\mathbf{X}_{1i}^{\text{int}}, 2 \leq i \leq n$ , the map from shape  $S_i$  to shape  $S_j$  is given by  $\mathbf{X}_{1j}^{\text{int}} \mathbf{X}_{1i}^{\text{int}T}$ . Figure 3 shows some representative results of our approach on the SCAPE dataset.

## 5 Convex Relaxation Analysis

In this section, we present conditions on  $\mathbf{X}_{ij}^{\text{in}}, (i, j) \in \mathcal{E}$  so that the underlying ground truth maps are recovered by solving (7). We first present an exact recovery theorem that is derived from analyzing the KKT optimality conditions. Then we show how to apply it to obtain more transparent exact recovery conditions. Without loss of generality, we assume that the samples on each shape are numbered/ordered in a way such that the map collection matrix  $\mathbf{X}^{\text{gt}}$  of the underlying ground truth maps are given simply by

$$\mathbf{X}^{\text{gt}} = \mathbf{e}\mathbf{e}^T \otimes I_m, \quad (9)$$

where  $\otimes$  denotes the kronecker product.

### 5.1 Exact recovery theorem

We derive an exact recovery theorem by following a two-step procedure used in analyzing the exact conditions of low-rank matrix recovery techniques [CR09, CLMW11]. Specifically, we first apply the KKT conditions to derive an exact recovery condition formulated as the existence of certain dual variables. We then choose appropriate dual certificates to obtain an explicit exact recovery condition, which is summarized as the major theorem. As the derivation procedure is rather technically involved, we leave the details to the supplemental material.

Our exact recovery condition will be expressed as the positive definiteness of a set of matrices that capture the structure of the incorrect correspondences in the input maps  $\mathbf{X}_{ij}^{\text{in}}$ . Specifically, we introduce  $m$  weighted graphs  $\mathcal{G}_s^{\text{false}} = (\mathcal{S}, \mathcal{E}_s^{\text{false}}), 1 \leq s \leq m$ , each of which encodes the incorrect correspondences that are related to the  $s$ -th sample on all input shapes. To define each graph  $\mathcal{G}_s^{\text{false}}$ , we add an edge  $(i, j) \in \mathcal{E}_s^{\text{false}}$  and define its associated edge weight  $w_{ij}^s$  via the following two conditions: (i) If  $\phi_{ij}^{\text{in}}$  does not map the  $s$ -th sample of  $S_i$  to the  $s$ -th sample of  $S_j$ , we set  $w_{ij}^s = 1$ . (ii) If  $\phi_{ij}^{\text{in}}$  maps two or more samples (one is the  $s$ -th sample) on  $S_i$  to the  $s$ -th sample of  $S_j$ , we set  $w_{ij}^s = 1/2$ . Now we present the major exact recovery theorem as follows:

**Theorem 5.1.** *Define  $L_{\mathcal{G}}$  and  $L_{\mathcal{G}_s^{\text{false}}}$  as the unweighted and weighted graph Laplacians of graph  $\mathcal{G}$  and  $\mathcal{G}_s^{\text{false}}$ , respectively. Then  $\mathbf{X}^{\text{gt}}$  is the unique solution to (7) if*

$$\lambda_2(L_{\mathcal{G}} - 2L_{\mathcal{G}_s^{\text{false}}}) > 0, \quad 1 \leq s \leq m,$$

where  $\lambda_2(A)$  denotes the second smallest eigenvalue of  $A$ .

### 5.2 Derived exact recovery conditions

In this section, we apply known results in spectral graph theory to Theorem 5.1 in order to obtain more transparent exact recovery conditions. We consider both a deterministic setting, where  $\mathcal{G}$  and  $\mathcal{G}_s^{\text{false}}, 1 \leq s \leq m$  are fixed, and a randomized setting, where they are random graphs.

**Deterministic setting.** A straightforward relaxation of (5.1) is given by

$$\lambda_{\max}(L_{\mathcal{G}_s^{\text{false}}}) \leq \lambda_2(L_{\mathcal{G}})/2, \quad 1 \leq s \leq m. \quad (10)$$

where  $\lambda_2(L_{\mathcal{G}})$  is also called the algebraic connectivity [Fie73] of graph  $\mathcal{G}$ . As the largest eigenvalue of a graph Laplacian is bounded by two times of the maximum (weighted) degree of its vertices, we arrive at the following relaxed exact recovery condition:

**Corollary 1.**  $\mathbf{X}^{gt}$  is the unique optimal solution to (7) if

$$\sum_{(i,j) \in \mathcal{E}_s^{false}} (w_{ij}^s + w_{ji}^s)/2 < \frac{\lambda_2(L_{\mathcal{G}})}{4}, \quad \begin{array}{l} 1 \leq i \leq n, \\ 1 \leq s \leq m. \end{array}$$

In other words, on the average, we can tolerate a fraction of  $\lambda_2(L_{\mathcal{G}})/4(n-1)$  incorrect correspondences from each sample on one shape to other samples on the remaining shapes. Note that we do not require the complete correctness of any particular input map.

When the input model graph  $\mathcal{G}$  is a complete graph, we have  $\lambda_2(L_{\mathcal{G}}) = n$ . In this case, we can tolerate 25% incorrect correspondences. It turns out this 25% upper bound is essentially tight. Consider the case where the input shape collection is divided into two sets of equal size so that the 25% incorrect correspondences only happen between shapes from different sets. Then there are 50% incorrect correspondences among the correspondences between these two sets and if there is collusion among the incorrect correspondences it is impossible to decide between the two options. This shows that the 25% bound is tight.

On the other hand, the probability of having such worst case scenarios is low. This motivates us to consider the case where both  $\mathcal{G}$  and  $\mathcal{G}_s^{false}$  are generated by a random process. It turns out we can obtain better bounds in such a setting.

**Randomized setting.** In the randomized setting, we assume that the edge set  $\mathcal{E}$  is generated using the Erdős-Rényi random graph model, i.e., each edge  $(i, j), 1 \leq i < j \leq N$  appears in  $\mathcal{E}$  with probability of  $p \in (0, 1)$ , independently of all the others. We further assume that the incorrect correspondences specified by  $x_{ij}(s) = 1, (i, j) \in \mathcal{E}, 1 \leq s \leq m$  are also generated independently with probability  $q \in (0, 1)$ . It is easy to see that  $\mathcal{G}_s^{false}, 1 \leq s \leq m$  also satisfy the Erdős-Rényi model but with probability  $pq$ .

As shown in [DJ10], under the Erdős-Rényi model, as the number of shapes  $n \rightarrow \infty$ , the second smallest and the largest eigenvalues of the Laplacian of a random graph  $\mathcal{G}$  all converge to  $np$  with probability 1, i.e.,

$$\lambda_2(L_{\mathcal{G}})/np \rightarrow 1, \quad \lambda_{\max}(L_{\mathcal{G}})/np \rightarrow 1.$$

Applying this result on  $\mathcal{G}$  and  $\mathcal{G}_s^{false}, 1 \leq s \leq m$ , respectively, we obtain the following asymptotic result:

**Corollary 2.** Suppose  $q < 1/2$ . Let  $m$  be fixed, while  $n \rightarrow \infty$ . Then

$$Pr(\lambda_2(L_{\mathcal{G}}) - 2\lambda_{\max}(L_{\mathcal{G}_s^{false}}) > 0, 1 \leq s \leq m) \rightarrow 1.$$

In other words, in the limiting case, we can tolerate almost 50% incorrect correspondences per sample. Note that this 50% bound is only achieved when  $n \rightarrow \infty$ . In practice, when  $n = 128, m = 16$  and  $p = 0.5$ , we found that (10) is satisfied with very high probability when  $q < 0.32$ .

## 6 Incorporating Sample Selection

In this section, we extend the constrained optimization formulation described in Section 4 to the generalized setting, where the variables to be optimized are the selected sub-shapes  $S'_i \subset S_i, |S'_i| = m_0, 1 \leq i \leq n$  and cycle-consistent one-to-one maps  $\Phi^{out} = \{\phi_{ij} : S'_i \rightarrow S'_j, 1 \leq i < j \leq n\}$  are sought between all pairs of selected sub-shapes.

### 6.1 Formulation

To represent the selected sub-shapes  $S'_i$ , we associate each sample  $s \in S_i, 1 \leq i \leq n$  with a binary indicator  $x_{i,s} \in \{0, 1\}$ , where  $x_{i,s} = 1$  if sample  $s$  is selected, and  $x_{i,s} = 0$  otherwise. With  $\mathbf{x}_i$  we denote the

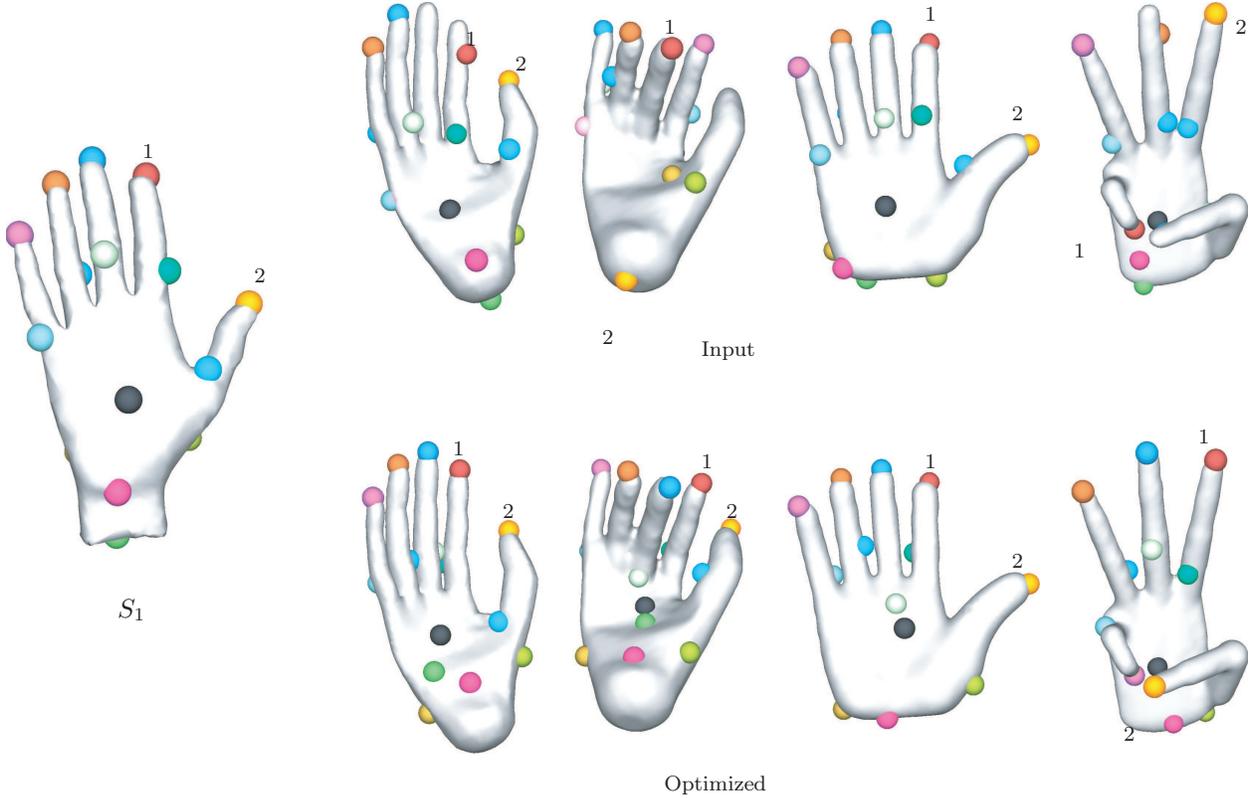


Figure 4: Comparison between the input maps and the optimized maps from a source shape to selected target shapes from the Hand dataset, The proposed approach is able to obtain consistent correspondences across the shapes.

binary indicator vector that collects all indicators of samples from  $S_i$ . It is easy to see that the constraints  $|S'_i| = m_0, 1 \leq i \leq n$  are equivalent to:

$$\mathbf{1}^T \mathbf{x}_i = m_0, \quad 1 \leq i \leq n. \quad (11)$$

The constraint that the selected samples on  $S_1$  are the first  $m_0$  samples is given by

$$\mathbf{x}_1 = (\mathbf{1}_{m_0}^T, \mathbf{0}_{m-m_0}^T)^T. \quad (12)$$

We still use matrix  $\mathbf{X}_{ij} \in \{0, 1\}^{m \times m}$  to encode the map  $\phi_{ij}$  between  $S'_i$  and  $S'_j$ , i.e.,  $\mathbf{X}_{ij}(s, s') = 1$  if and only if  $(s, s') \in \phi_{ij}$ . In this case, it is easy to see that  $\mathbf{X}_{ij}$  specifies a one-to-one map between  $S'_i$  and  $S'_j$  if and only if

$$\mathbf{X}_{ij} \mathbf{1} = \mathbf{x}_i, \quad \mathbf{X}_{ij}^T \mathbf{1} = \mathbf{x}_j. \quad (13)$$

Using this parameterization, the map collection matrix of  $\Phi = \{\phi_{ij} : S'_i \leftrightarrow S'_j, 1 \leq i < j \leq n\}$  is the submatrix of  $\mathbf{X}$  whose rows and columns are specified by  $\mathbf{x}_i, 1 \leq i \leq n$ . As the elements in the remaining rows or columns of  $\mathbf{X}$  are zero, it turns out we only need to enforce the positive semidefiniteness of  $\mathbf{X}$ . In summary, we have the following proposition.

**Proposition 2.** *Suppose the indicator vector for shape  $S'_i$  is given by vector  $\mathbf{x}_i$ . Then a binary matrix  $\mathbf{X} \in \{0, 1\}^{nm \times nm}$  specifies a set of cycle-consistent one-to-one maps between all pairs of shapes  $(S'_i, S'_j), 1 \leq i < j \leq n$  if and only if*

$$\begin{aligned} \mathbf{X}_{ii} &= \text{Diag}(\mathbf{x}_i), & 1 \leq i \leq n \\ \mathbf{X}_{ij} \mathbf{1} &= \mathbf{x}_i, \quad \mathbf{X}_{ij}^T \mathbf{1} = \mathbf{x}_j, & 1 \leq i < j \leq n \\ \mathbf{X} &\succeq 0. & \end{aligned} \quad (14)$$

We still use  $f_{align}$  described in (4) as the objective function to be minimized. Together with (12) and (14), we formulate the following constrained optimization problem for the generalized setting:

$$\begin{aligned}
& \underset{\mathbf{X}}{\text{minimize}} && \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{1}\mathbf{1}^T - 2\mathbf{X}_{ij}^{in}, \mathbf{X}_{ij} \rangle \\
& \text{s.t.} && \mathbf{X} \succeq 0, \mathbf{X} \in \{0, 1\}^{nm \times nm}, \mathbf{x}_1 = (\mathbf{1}_{m_0}^T, \mathbf{0}_{m-m_0}^T)^T \\
& && \mathbf{X}_{ii} = \text{Diag}(\mathbf{x}_i), \mathbf{1}^T \mathbf{x}_i = m_0, \quad 1 \leq i \leq n \\
& && \mathbf{X}_{ij}\mathbf{1} = \mathbf{x}_i, \mathbf{X}_{ij}^T \mathbf{1} = \mathbf{x}_j, \quad 1 \leq i < j \leq n
\end{aligned} \tag{15}$$

**Remark 1.** Based on the characteristics of the input, one may replace  $\mathbf{1}\mathbf{1}^T - 2\mathbf{X}_{ij}^{in}$  by other coefficient matrices. For example, if the input samples are highly irregular, we can utilize the distance metric associated with discrete metric spaces to define the objective

$$f_{align} = \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{W}_{ij}, \mathbf{X}_{ij} \rangle, \quad W_{ij}(p, p') = d_{S_j}(p', \phi_{ij}^{in}(p)), 1 \leq p, p' \leq m. \tag{16}$$

## 6.2 Convex relaxation

Let each element of  $X$  take real values between 0 and 1, we arrive the following convex relaxation to (15):

$$\begin{aligned}
& \min_X && \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{1}\mathbf{1}^T - 2X_{ij}^{in}, X_{ij} \rangle \\
& \text{s.t.} && X \succeq 0, X \geq 0, \mathbf{x}_1 = (\mathbf{1}_{m_0}^T, \mathbf{0}_{m-m_0}^T)^T, \\
& && X_{ii} = \text{Diag}(\mathbf{x}_i), \mathbf{1}^T \mathbf{x}_i = m_0, \quad 1 \leq i \leq n \\
& && X_{ij}\mathbf{1} = \mathbf{x}_i, X_{ij}^T \mathbf{1} = \mathbf{x}_j, \quad 1 \leq i < j \leq n
\end{aligned} \tag{17}$$

We adapt the rounding procedure used in the basic setting to obtain integer solutions to (15). Given  $X_{1j}$ , we first solve (8) to obtain a one-to-one map  $X_{1j}^{int}$  between  $S_1$  and  $S_j$ . As the first  $m_0$  samples of  $S_1$  is fixed, the restriction of  $X_{1j}^{int}$  on these fixed samples gives the selected shape  $S'_j \subset S_j$  as well as the one-to-one map between  $S'_1$  and  $S'_j$ . Figure 4 and Figure 5 shows some representative results on the Hand and Human datasets, respectively.

# 7 Experimental Evaluation

## 7.1 Experimental setup

**Datasets.** We have evaluated the proposed approach on three popular shape matching benchmark datasets: TOSCA [BBK08], SCAPE [ASK\*05] and SHREC07 [BBK08, KLF11]. Each dataset consists of one or multiple categories of similar shapes. Shapes in the TOSCA and SCAPE benchmark are generated by deforming template meshes, so we directly use the corresponding vertices as ground truth for evaluation. The SHREC07 benchmark consists of shapes that are triangulated differently. Following the setup in [KLF11, HZG\*12], we choose 11 categories from SHREC07, where point-based maps are meaningful, and evaluate against sparse manual correspondences as provided in [KLF11].

**Input maps.** We use blended intrinsic maps [KLF11] to establish initial maps between pairs of shapes. As blended intrinsic maps generate dense vertex-based correspondences, we snap them into point-to-point maps between the selected samples [HZG\*12]. Moreover, as the edge set  $\mathcal{E}$ , which specifies the pairs of shapes for computing initial maps, is adjustable, we consider a complete setting and a sparse setting. In the complete setting, we let  $\mathcal{E}$  connect all pairs of shapes. In the sparse setting, we let  $\mathcal{E}$  connect each shape

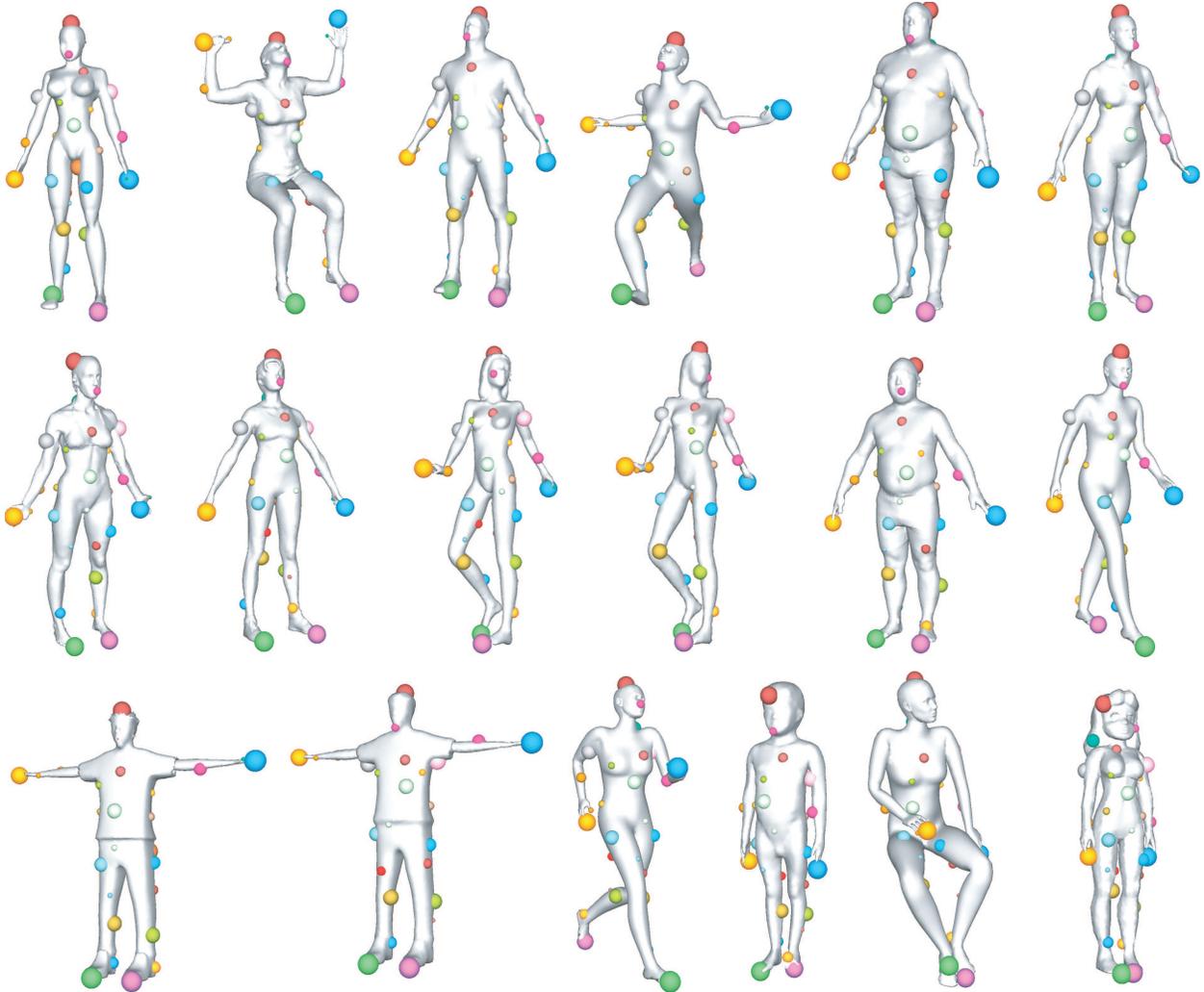


Figure 5: Consistent correspondences across the Human dataset. Matched points share the same color and radius.

with its  $k$ -nearest neighbor shapes ( $k = 5$  in this paper) in terms of the intrinsic D2-descriptor [OFC02] (a histogram of squared geodesic distances between pairs of points).

**Sampling.** Shapes in each category of TOSCA and SCAPE represent the same object, so we place  $m = 64$  samples on each shape and compute full one-to-one maps between these samples, i.e., we use the basic setting of Section 2.2. In contrast, shapes in each category of SHREC07 exhibit large geometric variance. In this case, we generate  $m = 128$  samples on each shape and compute one-to-one maps between  $m_0 = 32$  selected samples from each shape, as described in Section 6. The influences on using different number of samples will be discussed later in Section 7.4.

**Baseline.** We compare the proposed approach with three existing approaches: optimized composite maps (OCM) [NBCW\*11], fuzzy correspondences (FC) [KLM\*12] and hub-and-spoke network (HAS) [HZG\*12]. Note that both FC and HAS are sample based. As suggested by [HZG\*12], we used 512 samples for these two methods. To make a fair comparison between the proposed approach and previous approaches, we extrapolate sample based maps into vertex based maps, and compare the quality of the induced vertex based maps (See Figure 6). Given a sample based map  $\phi_{ij} : S_i \rightarrow S_j$ , we define the corresponding vertex based

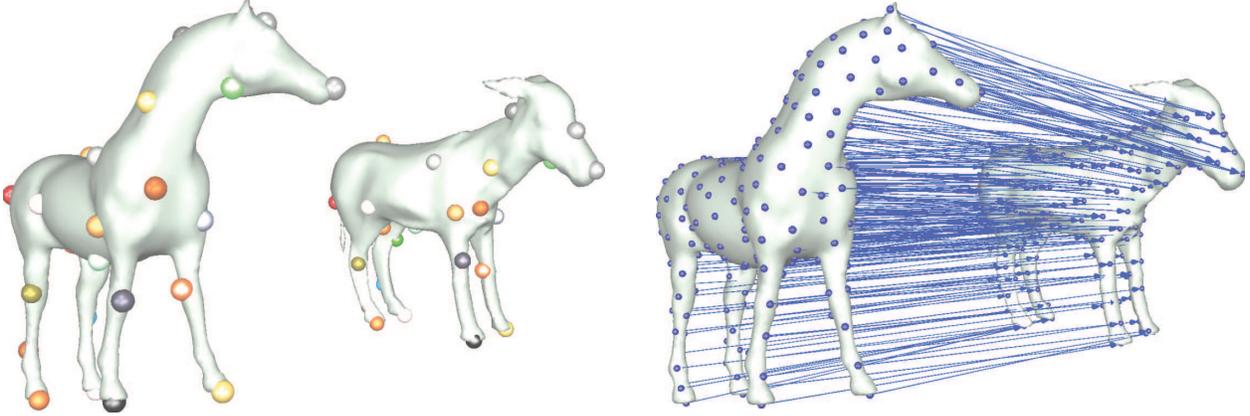


Figure 6: (Left) Color-coded correspondences between 32 sample points. (Right) Interpolated correspondences between vertices (256 are shown here).

map  $\bar{\phi}_{ij} : M_i \rightarrow M_j$  as:

$$\bar{\phi}_{ij}(v) = \arg \min_{v' \in M_j} \sum_{s \in S_i} e^{-\frac{d_{M_i}^2(v, s)}{2\sigma_i^2}} d_{M_j}^2(v', \phi_{ij}(s)), \quad v \in M_i, \quad (18)$$

where  $\sigma_i := \min_{s \neq s' \in S_i} d_{M_i}(s, s')$ .

**Evaluation metrics.** Denote  $\mathcal{C}_{ij}^{\text{gt}} \subset M_i \times M_j, 1 \leq i \neq j \leq N$  as the ground-truth correspondences. For the maps  $\bar{\phi}_{ij}, 1 \leq i \neq j \leq N$  generated by each method, we collect statistics of geodesic error of each correspondence  $(v_i, v_j) \in \mathcal{C}_{ij}^{\text{gt}}$ :

$$e_{\text{geo}}(\bar{\phi}_{ij}, (v_i, v_j)) = d_{M_j}(v_j, \bar{\phi}_{ij}(v_i)).$$

Following [KLF11], we report the percentage of correspondences  $p_\epsilon$  whose geodesic error is smaller than  $\epsilon$  times the mean of the maximum pairwise geodesic distance on each shape. When reporting  $p_\epsilon$ , we primarily focus on two values for  $\epsilon$ , a generous  $\epsilon = 0.16$ , which we take to capture the global accuracy of a method, and a tight  $\epsilon = 0.02$ , which captures its local accuracy. We provide the plots of  $p_\epsilon$  against  $\epsilon$  in the supplemental material.

## 7.2 Analysis of results

Table 1 and Figure 7 collect the statistics of the proposed approach on benchmark datasets. In the following, we break down in more detail the observed performance. As in [HZG\*12], we divide the 11 categories of SHREC07 into a NonSym group (i.e., Armadillo, Fourleg, Fish, Hand and Human), where input maps are not seriously influenced by the underlying symmetries of the shapes (blended intrinsic maps [KLF11] are resilient to reflectional intrinsic symmetries), and a Sym group (i.e., Ant, Bird, Glasses, Plane, Plier and Teddy), where a significant portion of the input maps are affected by shape symmetries.

**TOSCA and SCAPE.** The proposed approach generates nearly perfect results on the TOSCA and SCAPE datasets, i.e.,  $p_{0.16} = 100\%$  on both datasets. This is expected because shapes in each category are quite similar to each other so that the input maps are of high quality, i.e.,  $p_{0.16}^{\text{input}} = 83.4\%$  for TOSCA and  $p_{0.16}^{\text{input}} = 86.1\%$  for SCAPE. Moreover, the performance on the SCAPE dataset is slightly better than the performance on TOSCA. This is due to a data-driven effect [KLM\*12], i.e., SCAPE contains many more shapes than each category of the TOSCA dataset.

**SHREC07-NonSym.** The proposed approach generates similar results on the non-symmetric categories of the SHREC07 benchmark. On three categories, Human, Fourleg and Armadillo,  $p_{0.16} = 100\%$ , i.e.,

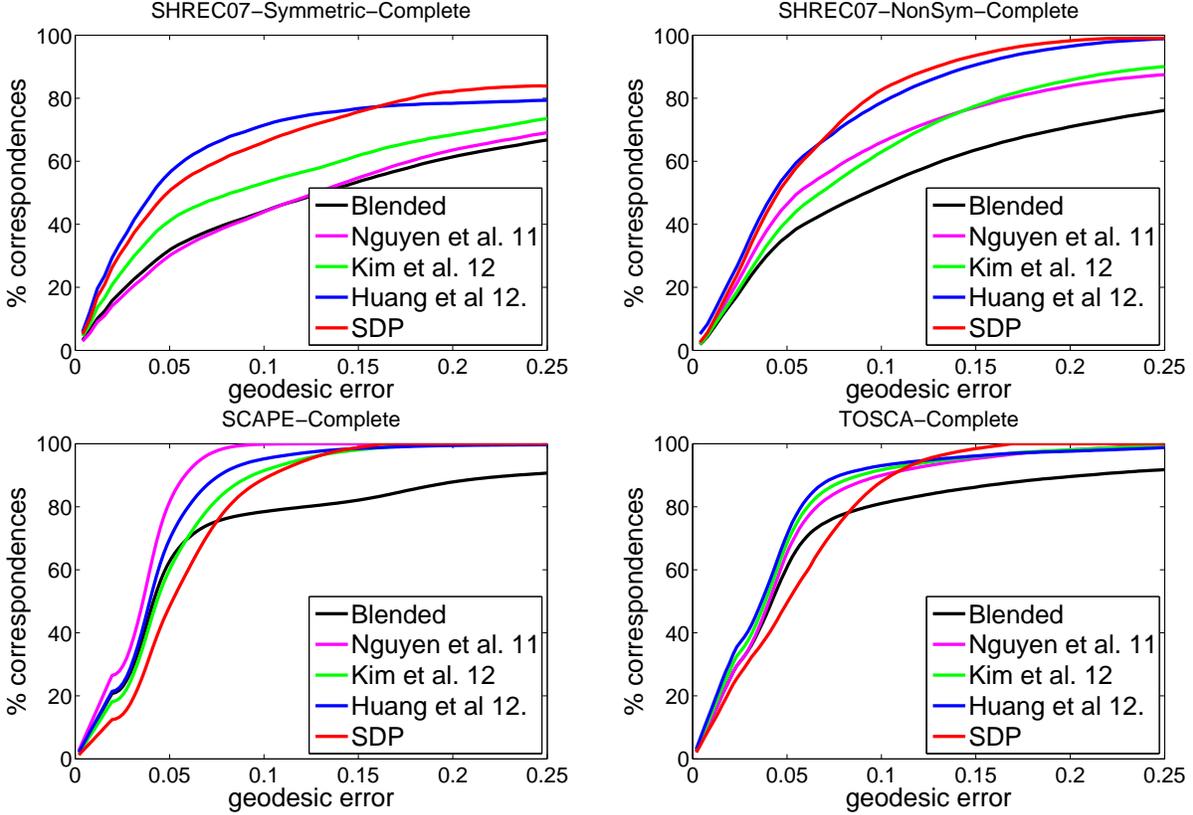


Figure 7: Comparison among optimized composite maps [NBCW\*11], fuzzy correspondences [KLM\*12], hub-and-spoke network [HZG\*12] and the proposed approach (SDP). The input maps are given by blended intrinsic maps [KLF11].

the ground truth correspondences are approximately fully recovered. The Fish and the Hand datasets are more challenging due to large geometric variance present in both datasets. Still, over 90% of ground truth correspondences are approximately recovered.

**SHREC07-Sym.** Similar to other approaches, the performance of the proposed approach drops on categories where the input maps are seriously affected by underlying global symmetric transforms and the ambiguities they introduce. This is because cycle-consistent maps can commute with global symmetric transforms, e.g., one can flip the maps between one shape and the remaining shapes without breaking the cycle-consistency constraint.

**Sparse versus Complete.** Overall, the proposed algorithm generates similar results in the complete graph and the sparse graph settings. The subtle differences between these two settings can be explained as follows. As the quality of input maps in the sparse graph setting is generally higher than that in the complete graph setting, the performance of the proposed approach in the sparse graph setting is overall slightly higher than the complete graph setting. One exception is the SCAPE dataset, where all shapes are very similar to each other. In that case, having a dense graph provides more high-quality routing options to establish maps between pairs of shapes than having a sparse graph, helping the results.

**Timing.** Based on a MATLAB implementation of the ADMM algorithm, on a machine with QuadCore 3.2GHZ Intel CPU and 12GB main memory, our approach took  $20m12s$  on the average to process a dataset of 20 shapes with 128 samples each. The most time-consuming part is the eigen-decomposition of a matrix of dimension  $NM \times NM$  at each iteration. In the future, we plan to improve the computational efficiency by utilizing the relations between the eigen-decompositions at consecutive iterations.

	#Shapes	Samples		Complete Graph		Sparse Graph	
		Input	Selected	$p_{0.16}^{in}$	$p_{0.16}$	$p_{0.16}^{in}$	$p_{0.16}$
TOSCA	<20	64	64	83.1	100	84.1	100
SCAPE	71	64	64	77.2	100	83.2	99.1
Armadillor	20	128	32	71.2	100	74.2	100
Fish	20	128	32	60.4	85.2	71.3	100
Fourleg	20	128	32	67.5	100	71.9	100
Hand	20	128	32	60.1	87.9	63.2	88.1
Human	18	128	32	74.5	100	79.1	100
Ant	20	128	32	54.1	77.1	58.1	79.1
Bird	20	128	32	57.1	78.4	59.1	81.4
Glasses	20	128	32	45.1	67.1	51.1	73.4
Plane	20	128	32	58.1	78.1	60.1	80.2
Plier	20	128	32	60.1	79.1	61.9	81.2
Teddy	20	128	32	61.1	81.1	64.1	83.1

Table 1: The performance of the proposed approach on benchmark datasets. We report  $p_{0.16}$  as a representative to access of the global behavior of the proposed approach.

	SDP		Baseline	
	Complete	Sparse	Complete	Sparse
Global ( $p_{0.16}$ )				
TOSCA	100	100	97.6 [NBCW*11]	97.2 [HZG*12]
SCAPE	100	99.1	100 [NBCW*11]	99.3 [HZG*12]
NonSym	94.6	97.4	90.3 [HZG*12]	90.5 [HZG*12]
Sym	79.3	80.2	77.2 [HZG*12]	78.5 [HZG*12]
Local ( $p_{0.02}$ )				
TOSCA	34.1	35.7	37.5 [NBCW*11]	38.4 [HZG*12]
SCAPE	41.2	42.1	48.6 [KLM*12]	44.4 [HZG*12]
NonSym	32.3	35.2	37.5 [HZG*12]	39.1 [HZG*12]
Sym	33.1	31.1	32.8 [HZG*12]	31.9 [HZG*12]

Table 2: Comparison between the proposed approach and the baseline approaches in terms of local ( $p_{0.02}$ ) and global accuracy ( $p_{0.16}$ ). Due to space constraint, we only show the best baseline algorithm for each dataset.

### 7.3 Comparison

Table 2 and Figure 7 present the comparisons between our approach and the three baseline algorithms. For each algorithm, we report both its local accuracy  $p_{0.02}$  and its global accuracy  $p_{0.16}$ .

**Global accuracy.** The global accuracy of the proposed approach is better than all baseline algorithms. On NonSym, the proposed approach yields  $p_{0.16} = 94.2(95.1)$  in the complete(partial) graph setting, while the best baseline algorithm HAS [HZG\*12] yields  $p_{0.16} = 90.3(90.5)$  respectively. Note that this is a little surprising. because HAS incorporates additional higher-order regularization constraints, e.g., that neighboring vertices should be mapped to neighboring vertices, in the optimization process.

**Local accuracy.** The local accuracy of the proposed approach is somewhat worse than the baseline algorithms. For example, in the complete graph setting of the SCAPE dataset,  $p_{0.02} = 48.6$  for OCM [NBCW\*11], while  $p_{0.02} = 41.2$  for the proposed approach. This is expected due to sampling aliasing, as we use a much smaller number of samples than the baseline algorithms for computing the maps. However, in practice, one can improve the local accuracy of the resulting maps by applying non-rigid registration post facto.

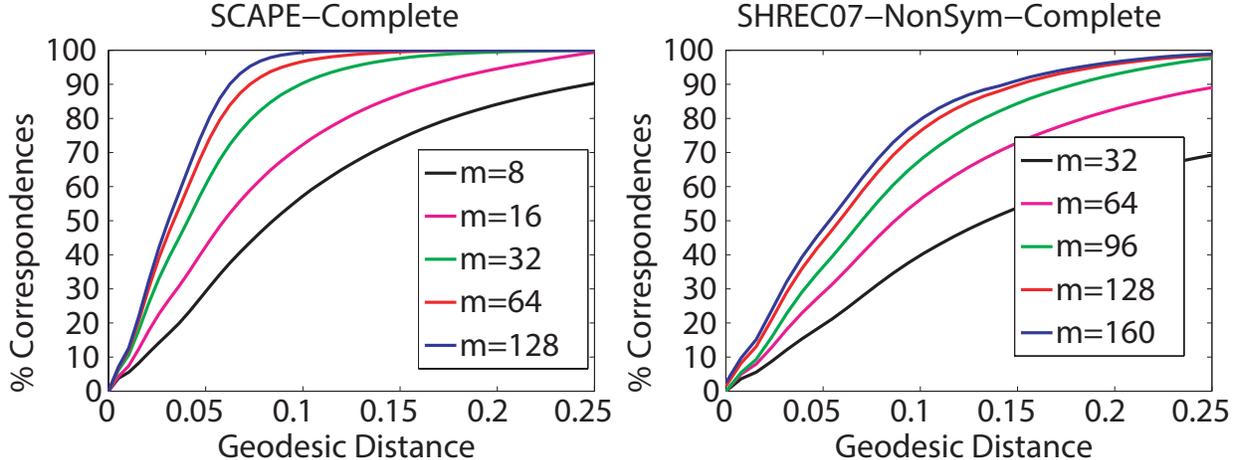


Figure 8: Influence of varying the number of samples  $m$ . (a) Vary  $m$  in the basic setting. We show the SCAPE dataset as the representative. (b) Fix the number of selected samples  $m_0 = 32$  and vary  $m$  in the generalized setting. We show the averaged result on SHREC07-NonSym as the representative.

#### 7.4 Influence of sampling density

**Basic setting.** Figure 8(a) shows the performance of our approach when using different number of samples  $m$ . For simplicity, we use the SCAPE dataset with complete graph  $\mathcal{G}$  as the representative. It is easy to see that the global behavior of the method becomes steady when  $m > 32$ . Moreover, using more samples always improves the local behavior of our approach. However, the improvement of using more samples from using 64 samples is not significant.

**Generalized setting.** We study the performance of our approach by fixing  $m_0 = 32$  and varying  $m$ . In this case, we use the NonSym group as the representative (See Figure 8(b)). Although the worst case analysis (See the supplemental material) suggests that we should choose  $m > 4m_0$ , we found that  $m$  can be made smaller in practice. On the NonSym group, the method becomes steady when  $m \geq 96$ .

## 8 Conclusions and Future Work

In this paper we have introduced a novel convex optimization framework for the problem of producing globally cycle-consistent maps between shapes in a collection, starting from maps computed between pairs of shapes in isolation. The presented approach formulates this multiple shape matching problem as a binary semi-definite program, and solves it through its convex relaxation. By analyzing the KKT optimality condition of this convex relaxation, we have provided upper bounds on the percentage of incorrect correspondence from each sample on one shape to samples on the remaining shapes, so that this convex relaxation is guaranteed to recover the ground-truth maps. Experimental results demonstrate that the presented approach improves on than state-of-art multiple shape matching methods on benchmark datasets.

There are ample opportunities for future research. We would like to extend the presented technique to handle shapes that are partially similar, e.g., range scans of a moving object. A potential approach is to augment the samples on each shape with pseudo samples, which represent the missing part of each shape. The presented approach can then be applied on the augmented samples. Moreover, the presented approach builds on sample-based surface representations. It would be interesting to explore how to apply the key idea to other surface and map representations, e.g., functional maps [OBCS\*12].

**Acknowledgements.** This work was supported by NSF grants CCF 1011228, CCF 1161480, DMS 1228304, AFOSR grant 1156110-1-TACAD, MURI grant N00014-13-1-0341, a Google Research Award, and the Max

Planck Center for Visual Computing and Communications. The authors would also like to thank Yuxin Chen, Xiaodong Li, Raif Rustamov, Hao Su and the anonymous reviewers for the helpful suggestions and comments.

## References

- [ASK\*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: Scape: shape completion and animation of people. In *ACM SIGGRAPH 2005 Papers* (2005), SIGGRAPH '05, pp. 408–416.
- [BBK08] BRONSTEIN A., BRONSTEIN M., KIMMEL R.: *Numerical Geometry of Non-Rigid Shapes*, 1 ed. Springer Publishing Company, Incorporated, 2008.
- [BPC\*11] BOYD S., PARIKH N., CHU E., PELEATO B., ECKSTEIN J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* 3, 1 (Jan. 2011), 1–122.
- [CLMW11] CANDÈS E. J., LI X., MA Y., WRIGHT J.: Robust principal component analysis? *J. ACM* 58, 3 (June 2011), 11:1–11:37.
- [CR09] CANDÈS E. J., RECHT B.: Exact matrix completion via convex optimization. *Found. Comput. Math.* 9, 6 (Dec. 2009), 717–772.
- [CT05] CANDES E. J., TAO T.: Decoding by linear programming. *Trans. Inf. Theor.* 51, 12 (Dec. 2005), 4203–4215.
- [DJ10] DING X., JIANG T.: Spectral distributions of adjacency and laplacian matrices of random graphs. *Annals of Applied Probability* 20(6) (2010), 2086–2117.
- [Fie73] FIEDLER M.: Algebraic Connectivity of Graphs. *Czechoslovak Mathematical Journal* 23 (1973), 298–305.
- [GBP07] GIORGI D., BIASDTTI S., PARABOSCHI L.: Shrec: shape retrieval context: Watertight models track, 2007.
- [HFG\*06] HUANG Q.-X., FLÖRY S., GELFAND N., HOFER M., POTTMANN H.: Reassembling fractured objects by geometric matching. *ACM Trans. Graph.* 25, 3 (2006), 569–578.
- [Hub02] HUBER D.: *Automatic Three-dimensional Modeling from Reality*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 2002.
- [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 (Nov. 2012), 167:1–167:11.
- [KLF11] KIM V. G., LIPMAN Y., FUNKHOUSER T.: Blended intrinsic maps. *ACM Trans. Graph.* 30, 4 (Aug. 2011), 79:1–79:12.
- [KLM\*12] KIM V. G., LI W., MITRA N., DIVERDI S., FUNKHOUSER T.: Exploring collections of 3d models using fuzzy correspondences. In *ACM SIGGRAPH 2012 papers* (2012), SIGGRAPH '12.
- [LF09] LIPMAN Y., FUNKHOUSER T.: Mobius voting for surface correspondence. *ACM Trans. Graph.* 28, 3 (July 2009), 72:1–72:12.
- [NBCW\*11] NGUYEN A., BEN-CHEN M., WELNICKA K., YE Y., GUIBAS L.: An optimization approach to improving collections of shape maps. In *Eurographics Symposium on Geometry Processing (SGP)* (2011), pp. 1481–1491.

- [OBCS\*12] OVSJANIKOV M., BEN-CHEN M., SOLOMON J., BUTSCHER A., GUIBAS L.: Functional maps: a flexible representation of maps between shapes. *ACM Trans. Graph.* 31, 4 (July 2012), 30:1–30:11.
- [OFCD02] OSADA R., FUNKHOUSER T., CHAZELLE B., DOBKIN D.: Shape distributions. *ACM Trans. Graph.* 21 (October 2002), 807–832.
- [Sch86] SCHRIJVER A.: *Theory of linear and integer programming*. John Wiley & Sons, Inc., New York, NY, USA, 1986.
- [SOG09] SUN J., OVSJANIKOV M., GUIBAS L.: A concise and provably informative multi-scale signature based on heat diffusion. In *SGP* (2009), pp. 1383–1392.
- [WGY10] WEN Z., GOLDFARB D., YIN W.: Alternating direction augmented Lagrangian methods for semidefinite programming. *Math. Prog. Comp.* 2, 3-4 (2010), 203–230.
- [WS12] WANG L., SINGER A.: Exact and stable recovery of rotations for robust synchronization.
- [ZKP10] ZACH C., KLOPSCHITZ M., POLLEFEYS M.: Disambiguating visual relations using loop constraints. In *CVPR* (2010), pp. 1426–1433.

## A A Support for the Choice of $m_0 = m/4$ in the Generalized Setting

In this section, we provide a support for the choice of  $m_0 = m/4$  by analyzing geodesic distortions of one-to-one maps between discrete metric spaces. Without losing generality, we use  $M, M', \dots$  to denote the original shapes. Accordingly, we denote their discrete metric spaces as  $S, S'$ . We begin with introducing several definitions, then we present the major result of the section.

**Definition 2.** We define the sampling density of a discrete metric space  $S$  as

$$\delta_S = \min_{s, s' \in S} d_M(s, s'),$$

where  $d_M(\cdot, \cdot)$  denotes the geodesic distance on  $M$ .

**Definition 3.** We define the covering radius of a discrete metric space  $S$  as

$$r_S = \max_{p \in M} \min_{s \in S} d_M(p, s).$$

It is easy to see that if discrete metric spaces are generated using furthest point sampling, then the sampling density of using  $m$  samples is identical to the covering radius of using  $m + 1$  samples.

$$\delta_{S \cup \{p\}} = r_S,$$

where  $p$  is the next sample on  $S$  produced by furthest point sampling.

Consider two approximately isometric shapes, i.e., a source shape  $M$  and a target shape  $M'$ . Denote  $f : M \leftrightarrow M'$  as the underlying bijective map between them. We model the approximate isometry as the fact that there exists small values of  $\epsilon_1, \epsilon_2$  such that

$$(1 - \epsilon_1)d_{M'}(f(s), f(s')) - \epsilon_2 \leq d_M(s, s') \leq (1 + \epsilon_1)d_{M'}(f(s), f(s')) + \epsilon_2, \quad \forall s, s' \in M.$$

Under this model, we have the following proposition.

**Proposition 3.** Suppose  $S$  and  $S'$  are the discrete metric spaces of shapes  $M$  and  $M'$ , respectively. Suppose

$$\delta_S > 2(1 + \epsilon_1)r_{S'} + \epsilon_2.$$

Then there exists a map  $\phi : S_1 \rightarrow S_2$  such that (i)  $\phi : S \leftrightarrow \phi(S)$  is a well-defined one-to-one map, and (ii)  $\phi$  is close to the underlying map between  $M$  and  $M'$ :

$$d_{M'}(\phi(s), f(s)) \leq r_{S'}, \quad \forall s \in S.$$

*Proof:* The proof is straight-forward. We simply define  $\phi$  as follows

$$\phi(s) := \arg \min_{s' \in S'} d(s', f(s)).$$

By the definition of  $r_{S'}$ , it is clear that

$$d_{M'}(\phi(s), f(s)) \leq r_{S'}, \quad \forall s \in S.$$

It remains to prove that  $\forall s_1 \neq s_2, \phi(s_1) \neq \phi(s_2)$ . Suppose there exist  $s_1 \neq s_2$  such that  $\phi(s_1) = \phi(s_2)$ . Then we have

$$\begin{aligned} \delta_S &\leq d_S(s_1, s_2) \leq (1 + \epsilon)d_{S'}(f(s_1), f(s_2)) + \epsilon_2 \\ &\leq (1 + \epsilon)(d_{S'}(f(s_1), \phi(s_1)) + d_{S'}(f(s_2), \phi(s_2))) + \epsilon_2 \\ &\leq 2(1 + \epsilon)r_{S'} + \epsilon_2 < \delta_S, \end{aligned}$$

which results in a contradiction.  $\square$

As the discrete metric spaces are generated using furthest point sampling. In practice, we found that the sampling density on a 2-manifold is decreased by half if we increase the number of samples by four times. In practice, we also found that  $\epsilon_1$  and  $\epsilon_2$  are rather small. This means that Proposition 1 is satisfied if  $m > 4m_0$ , which gives a theoretical support for the choice of  $m_0 = m/4$ .

Note that Proposition 1 is derived from a worst-case analysis. In practice,  $m$  can be made smaller than  $4m_0$ . Please refer to the main paper for an experimental analysis of the choice of  $m$  with fixed  $m_0$ .

## B Numerical Optimization Using ADMM

**Notations.** To simplify the expression, we agree with the following notations. Define  $\mathcal{A} : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^k$  as a matrix linear operator. Its conjugate operator is given by  $\mathcal{A}^*$  such that

$$\langle \mathcal{A}(bsX), \mathbf{x} \rangle = \langle \mathbf{X}, \mathcal{A}^*(\mathbf{x}) \rangle, \quad \forall \mathbf{X} \in \mathbb{R}^{n \times n}, \mathbf{x} \in \mathbb{R}^k.$$

Denote  $(\cdot)_+$  as the non-negative matrix operator, i.e., it applies  $x_+ = \max(x, 0)$  element-wise.

**Simplified program.** The optimization problem we are trying to solve is given by

$$\begin{aligned} \text{minimize}_{\mathbf{X}} \quad & \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{1}\mathbf{1}^T - 2\mathbf{X}_{ij}^{in}, \mathbf{X}_{ij} \rangle \\ \text{subject to} \quad & \mathbf{X} \succeq 0, \\ & \mathbf{X} \geq 0, \\ & \mathbf{X}_{ii} = \text{Diag}(\mathbf{x}_i), & (1 \leq i \leq n) \\ & \mathbf{1}^T \mathbf{x}_i = m_0, \mathbf{x}_i \leq \mathbf{1} & (1 \leq i \leq n) \\ & \mathbf{X}_{ij} \mathbf{1} = \mathbf{x}_i, \mathbf{X}_{ij}^T \mathbf{1} = \mathbf{x}_j, & (1 \leq i < j \leq n) \\ & \mathbf{x}_1 = (\mathbf{1}_{m_0}^T, \mathbf{0}_{m-m_0}^T)^T. \end{aligned} \tag{19}$$

Before proceeding, we present an equivalent program to (19), which has much fewer number of equality constraints, i.e., the equality constraints  $\mathbf{X}_{ij} \mathbf{1} = \mathbf{x}_i, \mathbf{X}_{ij}^T \mathbf{1} = \mathbf{x}_j, 1 \leq i < j \leq n$  are absorbed.

**Proposition 4.** (19) is equivalent to the following convex program

$$\begin{aligned}
& \text{minimize} && \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{1}\mathbf{1}^T - 2X_{ij}^{in}, X_{ij} \rangle \\
& \text{subject to} && \begin{bmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{bmatrix} \succeq 0, \\
& && \mathbf{X} \succeq 0, \\
& && \mathbf{X}_{ii} = \text{Diag}(\mathbf{x}_i), && (1 \leq i \leq n) \\
& && \mathbf{1}^T \mathbf{x}_i = m_0, \mathbf{x}_i \leq \mathbf{1} && (1 \leq i \leq n) \\
& && \mathbf{x}_1 = (\mathbf{1}_{m_0}^T, \mathbf{0}_{m-m_0}^T)^T.
\end{aligned} \tag{20}$$

where  $\mathbf{x} = (\mathbf{x}_1^T, \dots, \mathbf{x}_n^T)^T$ .

*Proof:* We first prove that

$$\begin{aligned}
& \mathbf{X} \succeq 0, \\
& \mathbf{X}_{ij}\mathbf{1} = \mathbf{1}, \mathbf{X}_{ij}^T\mathbf{1} = \mathbf{1}. \quad (1 \leq i < j \leq n) \\
& \mathbf{X}_{ii} = \text{Diag}(\mathbf{x}_i), \quad (1 \leq i \leq n) \\
& \mathbf{1}^T \mathbf{x}_i = m_0, \quad (1 \leq i \leq n)
\end{aligned} \Leftrightarrow \begin{aligned}
& \begin{bmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{bmatrix} \succeq 0, \\
& \mathbf{X}_{ii} = \text{Diag}(\mathbf{x}_i), \quad (1 \leq i \leq n) \\
& \mathbf{1}^T \mathbf{x}_i = m_0. \quad (1 \leq i \leq n)
\end{aligned}$$

$\Rightarrow$ : Any any vector  $\mathbf{y} = (y_0, \mathbf{y}_1^T, \dots, \mathbf{y}_n^T)^T$ , we have

$$\begin{aligned}
\mathbf{y}^T \begin{bmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{bmatrix} \mathbf{y} &= m_0 y_0^2 + 2 \sum_{i=1}^n \mathbf{x}_i^T \mathbf{y}_i + \sum_{1 \leq i, j \leq n} \mathbf{y}_i^T \mathbf{X}_{ij} \mathbf{y}_j \\
&= \sum_{1 \leq i, j \leq n} (\mathbf{y}_i + \frac{m_0 \mathbf{1}}{n})^T \mathbf{X}_{ij} (\mathbf{y}_j + \frac{m_0 \mathbf{1}}{n}) \\
&\geq 0.
\end{aligned}$$

It follows that  $\begin{bmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{bmatrix} \succeq 0$ .

$\Leftarrow$ : Let  $\mathbf{D} = \begin{pmatrix} 1 & 0 \\ 0 & \mathbf{I}_n \otimes \mathbf{1} \end{pmatrix} \in \mathbb{R}^{(nm+1) \times m}$ . Introduce  $\mathbf{Y} = \mathbf{D}^T \begin{pmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{pmatrix} \mathbf{D} \in \mathbb{R}^{(n+1) \times (n+1)}$ . It is easy to see that  $\mathbf{Y} \succeq 0$  and  $Y(i, i) = Y(1, i) = 1, 1 \leq i \leq n+1$ . Applying Lemma 1, we have that

$$\mathbf{Y} = m_0 \mathbf{1}\mathbf{1}^T.$$

Denote  $r = \text{rank} \left( \begin{bmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{bmatrix} \right)$ . As  $\begin{pmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{pmatrix} \succeq \mathbf{0}$ , we can find a matrix  $\mathbf{Z} = (\mathbf{z}, \mathbf{Z}_1, \dots, \mathbf{Z}_n) \in \mathbb{R}^{r \times (nm+1)}$  with  $\mathbf{z} \in \mathbb{R}^r, \mathbf{Z}_i \in \mathbb{R}^{r \times n}, 1 \leq i \leq n$  such that

$$\begin{pmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{pmatrix} = \mathbf{Z}^T \mathbf{Z}.$$

It follows from that

$$\mathbf{D}^T \begin{pmatrix} m_0 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{pmatrix} \mathbf{D} = (\mathbf{Z}\mathbf{D})^T (\mathbf{Z}\mathbf{D}) = m_0 \mathbf{1}\mathbf{1}^T.$$

Simple algebraic manipulation indicates that: there exists a unitary matrix  $\mathbf{U} \in \mathbb{R}^{r \times r}$  such that

$$\begin{aligned}
& \mathbf{U}\mathbf{Z}\mathbf{D} = \mathbf{e}_1 \cdot \mathbf{1}^T, \\
& \Rightarrow [\mathbf{U}\mathbf{z}, \mathbf{U}\mathbf{Z}_1\mathbf{1}, \dots, \mathbf{U}\mathbf{Z}_n\mathbf{1}] = \mathbf{e}_1 \cdot \mathbf{1}^T,
\end{aligned}$$

which immediately follows that

$$\mathbf{Z}_i \mathbf{1} = \mathbf{z}, \quad 1 \leq i \leq n.$$

One can then derive

$$\begin{aligned} \mathbf{X}_{ij} \mathbf{1} &= \mathbf{Z}_i^T \mathbf{Z}_j \mathbf{1} = \mathbf{Z}_i^T \mathbf{z} = \mathbf{x}_i, \\ \mathbf{X}_{ij}^T \mathbf{1} &= \mathbf{Z}_j^T \mathbf{Z}_i \mathbf{1} = \mathbf{Z}_j^T \mathbf{z} = \mathbf{x}_j, \end{aligned}$$

which concludes the proof.  $\square$

**ADMM.** Let us convert (20) into the standard form of semidefinite programs:

$$\begin{aligned} &\text{minimize} && \langle \mathbf{W}, \overline{\mathbf{X}} \rangle && \text{dual variables} \\ &\text{subject to} && \mathcal{A}(\overline{\mathbf{X}}) = \mathbf{b}, && \mathbf{y} \\ &&& \text{diag}(\overline{\mathbf{X}}) \leq \mathbf{1}, && \mathbf{z} \geq 0 \\ &&& \overline{\mathbf{X}} \succeq 0, && \mathbf{Y} \succeq 0 \\ &&& -\overline{\mathbf{X}} \leq 0, && \mathbf{Z} \geq 0 \end{aligned} \quad (21)$$

where  $\mathcal{A}(\overline{\mathbf{X}}) = \mathbf{b}$  encapsulate all equality constraints, e.g.,  $\mathbf{X}_{ii} = \text{diag}(\mathbf{x}_i)$ ,  $\mathbf{1}^T \mathbf{x}_i = m$ ,  $1 \leq i \leq n$ . Note that these linear constraints are nicely decoupled.

Let  $\mathbf{y}, \mathbf{z} (\geq 0), \mathbf{Y} (\succeq 0), \mathbf{Z} (\geq 0)$  be the dual variables of  $\mathcal{A}(\overline{\mathbf{X}}) = \mathbf{b}$ ,  $\text{diag}(\overline{\mathbf{X}}) \leq \mathbf{1}$ ,  $\overline{\mathbf{X}} \succeq 0$  and  $-\overline{\mathbf{X}} \leq 0$ , respectively. The Lagrangian multiplier of (21) is then given by

$$\begin{aligned} \mathcal{L} &= \langle \mathbf{W}, \overline{\mathbf{X}} \rangle - \langle \mathbf{Y}, \overline{\mathbf{X}} \rangle - \langle \mathbf{Z}, \overline{\mathbf{X}} \rangle \\ &\quad + \langle \mathcal{A}(\overline{\mathbf{X}}) - \mathbf{b}, \mathbf{y} \rangle + \langle \text{diag}(\overline{\mathbf{X}}) - \mathbf{1}, \mathbf{z} \rangle \\ &= -\mathbf{b}^T \mathbf{y} - \mathbf{1}^T \mathbf{z} + \langle \mathbf{W} - \mathbf{Y} - \mathbf{Z} + \mathcal{A}^*(\mathbf{y}) + \text{Diag}(\mathbf{z}), \overline{\mathbf{X}} \rangle \end{aligned}$$

Following [WGY10], we write down the augmented lagrangian of  $\mathcal{L}$  to be minimized as

$$\begin{aligned} \mathcal{L}' &= \mathbf{b}^T \mathbf{y} + \mathbf{1}^T \mathbf{z} + \langle \mathbf{Y} + \mathbf{Z} - \mathcal{A}^*(\mathbf{y}) - \text{Diag}(\mathbf{z}) - \mathbf{W}, \overline{\mathbf{X}} \rangle \\ &\quad + \frac{1}{2\mu} \|\mathbf{Y} + \mathbf{Z} - \mathcal{A}^*(\mathbf{y}) - \text{Diag}(\mathbf{z}) - \mathbf{W}\|_{\mathcal{F}}^2, \end{aligned}$$

Here  $\mu$  is a parameter, whose value is increased during the optimization process.

We optimize  $\mathcal{L}'$  by alternating the optimizations of  $\mathbf{y}, \mathbf{z}, \mathbf{Y}$  and  $\mathbf{Z}$  independently, and then update the primal variable  $\overline{\mathbf{X}}$ . Initially, we set  $\mathbf{y}_0 = \mathbf{z}_0 = 0$  and  $\mathbf{X}_0 = \mathbf{Y}_0 = \mathbf{Z}_0 = 0$ . At iteration  $k + 1$ , we perform the following alternating direction optimizations in order :

$$\begin{aligned} \mathbf{y}_{k+1} &= \arg \min_{\mathbf{y}} \mathcal{L}'(\mathbf{y}, \mathbf{z}_k, \mathbf{Y}_k, \mathbf{Z}_k, \overline{\mathbf{X}}_k) \\ &= (\mathcal{A}\mathcal{A}^*)^{-1}(\mathcal{A}(\mathbf{Y}_k + \mathbf{Z}_k + \mu \overline{\mathbf{X}}_k - \mathbf{z}_k - \mathbf{W}) - \mu \mathbf{b}) \\ \mathbf{z}_{k+1} &= \arg \min_{\mathbf{z}} \mathcal{L}'(\mathbf{y}_{k+1}, \mathbf{z}, \mathbf{Y}_k, \mathbf{Z}_k, \overline{\mathbf{X}}_k) \\ &= (\text{diag}(\mathbf{Y}_k + \mathbf{Z}_k + \mu \overline{\mathbf{X}}_k - \mathcal{A}^*(\mathbf{y}_{k+1}) - \mathbf{W}) - \mu \mathbf{1})_+ \\ \mathbf{Z}_{k+1} &= \arg \min_{\mathbf{Z}} \mathcal{L}'(\mathbf{y}_{k+1}, \mathbf{z}_{k+1}, \mathbf{Y}_k, \mathbf{Z}, \overline{\mathbf{X}}_k) \\ &= (\mathcal{A}^*(\mathbf{y}_{k+1}) + \text{Diag}(\mathbf{z}_{k+1}) + \mathbf{W} - \mathbf{Y}_k - \mu \overline{\mathbf{X}}_k)_+ \\ \mathbf{Y}_{k+1} &= \arg \min_{\mathbf{Y}} \mathcal{L}'(\mathbf{y}_{k+1}, \mathbf{z}_{k+1}, \mathbf{Y}, \mathbf{Z}_{k+1}, \overline{\mathbf{X}}_k) = \mathbf{Q}\Sigma_+ \mathbf{Q}^T, \end{aligned}$$

where  $\mathbf{Q}$  and  $\Sigma$  are given by the spectral decomposition of

$$\begin{aligned} \mathbf{V}_{k+1} &= \mathbf{W} + \mathcal{A}^*(\mathbf{y}_{k+1}) + \text{Diag}(\mathbf{z}_{k+1}) - \mathbf{Z}_{k+1} - \mu \overline{\mathbf{X}}_k \\ &= \mathbf{Q}\Sigma \mathbf{Q}^T. \end{aligned}$$

Finally,  $\bar{\mathbf{X}}_{k+1}$  is updated as

$$\begin{aligned}\bar{\mathbf{X}}_{k+1} &= \bar{\mathbf{X}}_k + (\mathbf{W} - \mathbf{Y}_{k+1} - \mathbf{Z}_{k+1} + \mathcal{A}^*(\mathbf{y}_{k+1}) + \text{Diag}(\mathbf{z}_{k+1}))/\mu \\ &= (\mathbf{V}_{k+1} - \mathbf{Y}_{k+1})/\mu = -\mathbf{Q}(-\Sigma)_+ \mathbf{Q}^T/\mu.\end{aligned}$$

For all the experiments, we set  $\mu = 1$ , and increase  $\mu = \mu * 1.02$  after each iteration. We set the stoppage criterion as  $\|\bar{\mathbf{X}}_{k+1} - \bar{\mathbf{X}}_k\|_{\mathcal{F}} \leq 10^{-4}$ . In practice, we found that 400-600 iterations are sufficient for convergence. As the equality constraints are nicely decoupled, which admit effective prefactorization, the most time-consuming operation of this ADMM is computing the spectral decomposition of  $\mathbf{V}_{k+1}$ .

## C Proof of Theorem 5.1

In this section, we present the derivation of Theorem 5.1 in the main text. For convenience, let us reintroduce the convex program to be analyzed as follows:

$$\begin{aligned}\text{minimize} \quad & \sum_{(i,j) \in \mathcal{E}} \langle \mathbf{1}\mathbf{1}^T - 2\mathbf{X}_{ij}^{in}, \mathbf{X}_{ij} \rangle \\ \text{subject to} \quad & \mathbf{X} \geq 0, \\ & \mathbf{X} \succeq 0, \\ & \mathbf{X}_{ii} = I_m, \quad (1 \leq i \leq n) \\ & \mathbf{X}_{ij}\mathbf{1} = \mathbf{1}, \mathbf{X}_{ij}^T\mathbf{1} = \mathbf{1}. \quad (1 \leq i < j \leq n)\end{aligned}\tag{22}$$

Furthermore, we assume the map collection matrix of the underlying ground truth maps is given by

$$\mathbf{X}^{gt} = (\mathbf{e}\mathbf{e}^T) \otimes I_m.$$

The derivation of Theorem 5.1 proceeds in two steps. In the first step, we analyze the KKT optimality conditions of (22), and derive exact recovery conditions via the existence of dual certificates. Then in the second step, we choose appropriate dual certificates to derive Theorem 5.1.

To simplify the notations, we introduce matrices  $\mathbf{W}_{ij} \in \mathbb{R}^{m \times m}$ ,  $1 \leq i < j \leq n$ , where  $\mathbf{W}_{ij} = (2\mathbf{1}\mathbf{1}^T - \mathbf{X}_{ij}^{in} - \mathbf{X}_{ji}^{inT})$ ,  $(i, j) \in \mathcal{E}$ , and otherwise  $\mathbf{W}_{ij} = 0$ . This allows us to rewrite the objective function in (22) as

$$f_{align} = \sum_{1 \leq i < j \leq n} \langle \mathbf{W}_{ij}, \mathbf{X}_{ij} \rangle.$$

### C.1 Optimality Condition via Dual Certificates

The goal of this section is to derive a sufficient condition on  $\mathbf{W}_{ij}$ ,  $1 \leq i < j \leq n$  such that  $\mathbf{X}^{gt}$  is the unique optimal solution to (22). We achieve this goal by analyzing the KKT optimality conditions of (7). Note that even though KKT conditions are only necessary conditions in general, when applied on semi-definite programs, they typically indicate sufficient conditions by changing certain greater or equal relations, i.e.,  $\geq$ , into greater relations, i.e.,  $>$ .

Let  $\mathbf{Y} \succeq 0$  and  $\mathbf{Z}_{ij} \geq 0$ ,  $\mathbf{u}_{ij}$ ,  $\mathbf{v}_{ij}$ ,  $1 \leq i < j \leq n$  be the dual variables of the constraints  $\mathbf{X} \succeq 0$ ,  $-\mathbf{X}_{ij} \leq 0$ ,  $\mathbf{X}_{ij}\mathbf{1} = \mathbf{1}$ ,  $\mathbf{X}_{ij}^T\mathbf{1} = \mathbf{1}$ ,  $1 \leq i < j \leq n$ , respectively. We obtain the following necessary KKT conditions such that  $\mathbf{X}^{gt} = (\mathbf{1}\mathbf{1}^T) \otimes I_m$  is an optimal solution to (22):

$$\mathbf{W}_{ij} - \mathbf{Y}_{ij} - \mathbf{Z}_{ij} - \mathbf{1}\mathbf{u}_{ij}^T - \mathbf{v}_{ij}\mathbf{1}^T = 0, \quad (1 \leq i < j \leq n)\tag{23}$$

$$\mathbf{Z}_{ij} \geq 0, \quad \text{diag}(\mathbf{Z}_{ij}) = 0, \quad (1 \leq i < j \leq n)\tag{24}$$

$$\sum_{j=1}^n \mathbf{Y}_{ij} = 0, \quad (1 \leq i \leq n)\tag{25}$$

$$\mathbf{Y} \succeq 0.\tag{26}$$

Denote  $\mathbf{V}_m = \mathbf{I}_m - \frac{1}{m}\mathbf{1}\mathbf{1}^\top$ , it is easy to see that (23) is equivalent to

$$\mathbf{V}_m \mathbf{Y}_{ij} \mathbf{V}_m = \mathbf{V}_m (\mathbf{W}_{ij} - \mathbf{Z}_{ij}) \mathbf{V}_m. \quad (1 \leq i < j \leq n) \quad (27)$$

In other words, if (27) is true, we can always choose  $\mathbf{u}_{ij}, \mathbf{v}_{ij}$  such that (23) is satisfied. Combing (27) and (25), we obtain the following necessary optimality condition for  $\mathbf{X}^{gt}$ :

$$(\mathbf{I}_n \otimes \mathbf{V}_m) \mathbf{L} (\mathbf{I}_n \otimes \mathbf{V}_m) \succeq 0, \quad (28)$$

where  $\mathbf{L}$  is a matrix of  $n \times n$  blocks, and its  $(i, j)$ -th block is given by

$$\mathbf{L}_{ij} = \begin{cases} -\sum_{k \neq i} \mathbf{L}_{ik} & i = j \\ \mathbf{V}_m (\mathbf{W}_{ij} - \mathbf{Z}_{ij}) \mathbf{V}_m & i < j \\ \mathbf{L}_{ji}^\top & j > i \end{cases}$$

We want to point out there exist implicit constraints on  $\mathbf{Z}_{ij}$ , i.e., the diagonal blocks  $\mathbf{L}_{ii}$  have to be symmetric.

Denote  $\mathbf{V}_{n,m} = \mathbf{I}_{nm} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top \otimes \mathbf{I}_m$ . It is easy to see that the eigenvalues of  $\mathbf{L}$  are zero in the union of null spaces

$$\mathcal{N}_{n,m} = \{\mathbf{y} | (\mathbf{I}_n \otimes \mathbf{V}_m) \mathbf{y} = 0 \text{ or } \mathbf{V}_{n,m} \mathbf{y} = 0\}.$$

It turns out that if we force  $\mathbf{L}$  to be positive definite in the dual space  $\mathcal{N}_{n,m}^\perp$ , i.e., the remaining eigenvalues of  $\mathbf{L}$  are positive, then (28) becomes sufficient.

**Lemma 2.** Let  $\mathbb{R}_{+/d}^{m \times m}$  denote the space of  $m \times m$  matrices, whose entries are positive, and whose diagonal entries are zero. Then  $\mathbf{X}^{gt}$  is the unique optimal solution to (22) if there exist dual certificates  $\mathbf{Z}_{ij} \in \mathbb{R}_{+/d}^{m \times m}$ ,  $1 \leq i < j \leq n$  such that  $\mathbf{L}_{ii}$ ,  $1 \leq i \leq n$  are symmetric and

$$\mathbf{y}^\top \mathbf{L} \mathbf{y} > 0, \quad \forall \mathbf{0} \neq \mathbf{y} \in \mathcal{N}_{n,m}^\perp.$$

To prove Lemma 2, we show that a small feasible perturbation of  $\mathbf{X}^{gt}$  would always increase the value of the objective function  $f_{align}$ . We first present three useful propositions.

**Proposition 5.** A small feasible perturbation  $\Delta$  to  $\mathbf{X}^{gt}$ , i.e.,  $\mathbf{X}^{gt} + \Delta$  is feasible, satisfies the following constraints:

$$\Delta_{ii} = 0, \quad (1 \leq i \leq n) \quad (29)$$

$$\Delta_{ji} = \Delta_{ij}^\top, \mathbf{I}_m + \Delta_{ij} \geq 0, \quad (1 \leq i < j \leq n) \quad (30)$$

$$\Delta_{ij} \mathbf{1} = 0, \quad \Delta_{ij}^\top \mathbf{1} = 0, \quad (1 \leq i < j \leq n) \quad (31)$$

$$\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} \succeq 0. \quad (32)$$

*Proof:* (29) to (31) are obvious. To prove (32), it is easy to see that  $\forall \mathbf{y}_1 \in \mathbb{R}^m$ :

$$\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} (\mathbf{1} \otimes \mathbf{I}_m) \mathbf{y}_1 = 0.$$

It follows that we only need to prove that  $\forall \mathbf{y} \in \mathbb{R}^{nm}$ , where  $(\mathbf{1}^\top \otimes \mathbf{I}_m) \mathbf{y} = 0$ ,

$$\mathbf{y}^\top \mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} \mathbf{y} \geq 0.$$

In fact, as  $\mathbf{X}^{gt} + \Delta \geq 0$ , we have

$$\begin{aligned} 0 &\geq \mathbf{y}^\top (\mathbf{X}^{gt} + \Delta) \mathbf{y} \\ &= \mathbf{y}^\top ((\mathbf{1}\mathbf{1}^\top) \otimes \mathbf{I}_m + ((\mathbf{1}\mathbf{1}^\top) \otimes \mathbf{I}_m) \Delta + \Delta ((\mathbf{1}\mathbf{1}^\top) \otimes \mathbf{I}_m) - ((\mathbf{1}\mathbf{1}^\top) \otimes \mathbf{I}_m) \Delta ((\mathbf{1}\mathbf{1}^\top) \otimes \mathbf{I}_m)) \mathbf{y} \\ &\quad + \mathbf{y}^\top \mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} \mathbf{y} \\ &= \mathbf{y}^\top \mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} \mathbf{y}. \end{aligned}$$

□

**Proposition 6.** Suppose a matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is positive definite on a subspace  $\mathcal{U} \in \mathbb{R}^n$ , i.e.,

$$\mathbf{y}^T \mathbf{A} \mathbf{y} > 0, \quad \forall \mathbf{0} \neq \mathbf{y} \in \mathcal{U}.$$

Then for all non-zero semi-positive definite matrix  $\mathbf{B} \succeq 0$  whose column space  $\mathcal{C}(\mathbf{B}) \subseteq \mathcal{U}$ ,

$$\langle \mathbf{A}, \mathbf{B} \rangle > 0.$$

*Proof:* Suppose the dimension of  $\mathcal{U}$  is  $m \leq n$ . Consider an orthonormal decomposition of

$$\mathbf{A} = (\mathbf{U}_1, \mathbf{U}_2) \text{Diag}(\sigma_1, \dots, \sigma_n) (\mathbf{U}_1, \mathbf{U}_2)^T,$$

where  $\mathbf{U}_1$  is an orthonormal basis of  $\mathcal{U}$ . As  $\mathbf{A}$  is positive definite on  $\mathcal{U}$ , it follows that  $\sigma_i > 0, 1 \leq i \leq m$ . As  $\mathcal{C}(\mathbf{B}) \subseteq \mathcal{U}$ , we have  $\mathbf{B} \mathbf{U}_2 = 0$ . It follows that

$$\begin{aligned} \langle \mathbf{A}, \mathbf{B} \rangle &= \text{Trace}(\mathbf{A} \mathbf{B}) = \text{Trace}((\mathbf{U}_1, \mathbf{U}_2) \text{Diag}(\sigma_1, \dots, \sigma_n) (\mathbf{U}_1, \mathbf{U}_2)^T \mathbf{B}) \\ &= \text{Trace}(\text{Diag}(\sigma_1, \dots, \sigma_n) (\mathbf{U}_1, \mathbf{U}_2)^T \mathbf{B} (\mathbf{U}_1, \mathbf{U}_2)) \\ &= \text{Trace}(\text{Diag}(\sigma_1, \dots, \sigma_m) \mathbf{U}_1^T \mathbf{B} \mathbf{U}_1) \\ &= \sum_{i=1}^m \sigma_i (\mathbf{U}_1^T \mathbf{B} \mathbf{U}_1)_{ii}. \end{aligned}$$

As  $\sigma_i > 0$ ,  $(\mathbf{U}_1^T \mathbf{B} \mathbf{U}_1)_{ii} > 0, 1 \leq i \leq m$ . It means that  $\langle \mathbf{A}, \mathbf{B} \rangle \geq 0$ , and the equality is satisfied if and only if  $(\mathbf{U}_1^T \mathbf{B} \mathbf{U}_1)_{ii} = 0, 1 \leq i \leq m$ , which is equivalent to  $\mathbf{U}_1^T \mathbf{B} \mathbf{U}_1 = 0$ . Together with  $\mathbf{B} \mathbf{U}_2 = 0$  we have  $(\mathbf{U}_1, \mathbf{U}_2)^T \mathbf{B} (\mathbf{U}_1, \mathbf{U}_2) = 0$ , which ends the proof.  $\square$

**Proposition 7.**

$$\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} = 0 \quad \leftrightarrow \quad \mathbf{V}_{n,m} = 0.$$

*Proof:* Denote

$$\bar{\Delta}_{i\cdot} = \frac{1}{n} \sum_{j=1}^n \Delta_{ij}, \quad \bar{\Delta}_{\cdot j} = \frac{1}{n} \sum_{i=1}^n \Delta_{ji}, \quad \bar{\Delta}_{\cdot\cdot} = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n \Delta_{ij}. \quad (33)$$

Then  $\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} = 0$  yields

$$0 = (\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m})_{ij} = \Delta_{ij} - \bar{\Delta}_{i\cdot} - \bar{\Delta}_{\cdot j} + \bar{\Delta}_{\cdot\cdot}, \quad 1 \leq i \leq j \leq n. \quad (34)$$

As the diagonal blocks of  $\Delta$  are zero, i.e.,  $\Delta_{ii} = 0, 1 \leq i \leq n$ , we have

$$\bar{\Delta}_{\cdot\cdot} = \bar{\Delta}_{i\cdot} + \bar{\Delta}_{\cdot i}, \quad 1 \leq i \leq n.$$

Summing the two sides of the equation above from 1 to  $n$ , we have  $n \bar{\Delta}_{\cdot\cdot} = 2n \bar{\Delta}_{\cdot\cdot}$ , which results in  $\bar{\Delta}_{\cdot\cdot} = 0$ . According to (30), we have  $\bar{\Delta}_{i\cdot} = \bar{\Delta}_{\cdot i}^T$ . Together with  $\bar{\Delta}_{i\cdot} + \bar{\Delta}_{\cdot i} = 0$ , we obtain  $\bar{\Delta}_{i\cdot} = \bar{\Delta}_{\cdot i} = 0$ . It follows from (34) that  $\Delta_{ij} = 0$ .  $\square$

**Proof of Lemma 2:** Suppose  $\Delta \neq 0$ . According to Proposition 7 we have  $\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} \neq 0$ . Based on (31), it is easy to see that the column space of  $\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m}$  lies in the space  $\mathcal{N}_{n,m}^\perp$ . Applying Proposition 6 on  $\mathbf{L}$  and  $\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m}$ , we have

$$\begin{aligned} &0 < \langle \mathbf{L}, \mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m} \rangle \\ &= \sum_{i=1}^n \langle \mathbf{L}_{ii}, (\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m})_{ii} \rangle + \sum_{i=1}^n \sum_{j \neq i} \langle \mathbf{L}_{ij}, (\mathbf{V}_{n,m} \Delta \mathbf{V}_{n,m})_{ij} \rangle \\ &= \sum_{i=1}^n \sum_{j \neq i} \langle \mathbf{V}_m (\mathbf{Z}_{ij} - \mathbf{W}_{ij}) \mathbf{V}_m, -\bar{\Delta}_{i\cdot} - \bar{\Delta}_{\cdot i} + \bar{\Delta}_{\cdot\cdot} \rangle \\ &+ \sum_{i=1}^n \sum_{j \neq i} \langle \mathbf{V}_m (\mathbf{W}_{ij} - \mathbf{Z}_{ij}) \mathbf{V}_m, \Delta_{ij} - \bar{\Delta}_{i\cdot} - \bar{\Delta}_{\cdot i} + \bar{\Delta}_{\cdot\cdot} \rangle. \end{aligned}$$

Applying (31), we have

$$\mathbf{V}_m \Delta_{ij} = \Delta_{ij} \mathbf{V}_m = \Delta_{ij}, \quad 1 \leq i, j \leq n.$$

It follows that

$$\begin{aligned} 0 &< \sum_{i=1}^n \sum_{j \neq i} \langle (\mathbf{Z}_{ij} - \mathbf{W}_{ij}), -\bar{\Delta}_{i\cdot} - \bar{\Delta}_{\cdot i} + \bar{\Delta}_{\cdot\cdot} \rangle \\ &+ \sum_{i=1}^n \sum_{j \neq i} \langle (\mathbf{W}_{ij} - \mathbf{Z}_{ij}), \Delta_{ij} - \bar{\Delta}_{i\cdot} - \bar{\Delta}_{\cdot j} + \bar{\Delta}_{\cdot\cdot} \rangle \\ &= 2 \sum_{i=1}^n \sum_{i < j} \langle \mathbf{W}_{ij} - \mathbf{Z}_{ij}, \Delta_{ij} \rangle \end{aligned}$$

As  $\mathbf{I}_m + \Delta_{ij} \geq 0$ , it means the off-diagonal entries of  $\Delta_{ij}$  are non-negative. As  $\mathbf{Z}_{ij} \in \mathbb{R}_{+/d}^{m \times m}$ , i.e, its diagonal entries are zero, and its off-diagonal entries are non-negative, we have

$$\langle \mathbf{Z}_{ij}, \Delta_{ij} \rangle \geq 0, \quad 1 \leq i < j \leq n.$$

This means

$$0 < \sum_{i=1}^n \sum_{i < j} \langle \mathbf{W}_{ij}, \Delta_{ij} \rangle,$$

which ends the proof.  $\square$

## C.2 Choosing dual certificates.

Note that any choose of  $\mathbf{Z}_{ij}, (i, j) \in \mathcal{E}$  such that  $\mathbf{L}_{ii}, 1 \leq i \leq n$  are symmetric would lead to an exact recovery condition.

In this paper, we choose  $\mathbf{Z}_{ij} = \mathbf{Z}'_{ij} + \mathbf{Z}'_{ji}, 1 \leq i < j \leq n$  as

$$\mathbf{Z}'_{ij} = \begin{cases} \mathbf{I}_m - 2\text{Diag}(\mathbf{w}_{ij}) + \hat{\mathbf{w}}_{ij} \hat{\mathbf{w}}_{ij}^T - \mathbf{X}_{ij}^{in} & (i, j) \in \mathcal{E} \\ 0 & \text{otherwise,} \end{cases}$$

where  $\mathbf{w}_{ij} \in \mathbb{R}^m$  denotes the vector that collects edge weights  $w_{ij}^s, 1 \leq s \leq m$ . Note that we set  $w_{ij}^s = 0$  if  $(i, j) \notin \mathcal{E}_s^{\text{false}}$ . Vector  $\hat{\mathbf{w}}_{ij} = \mathbf{w}_{ij} > 0$  denotes the binary vector that specifies the positiveness of each element of  $\mathbf{w}_{ij}$ .

It is easy to see that  $\mathbf{Z}'_{ij} \geq 0, \text{Diag}(\mathbf{Z}'_{ij}) = 0, \forall (i, j) \in \mathcal{E}$ , and

$$\begin{aligned} \mathbf{L}_{ij} &= \mathbf{V}_m (2\mathbf{1}\mathbf{1}^T - \mathbf{X}_{ij}^{in} - \mathbf{X}_{ji}^{in} - \mathbf{Z}_{ij}) \mathbf{V}_m \\ &= -2\mathbf{V}_m \text{Diag}(\mathbf{1} - (\mathbf{w}_{ij} + \mathbf{w}_{ji})/2) \mathbf{V}_m - \mathbf{V}_m (\hat{\mathbf{w}}_{ij} \hat{\mathbf{w}}_{ij}^T + \hat{\mathbf{w}}_{ji} \hat{\mathbf{w}}_{ji}^T) \mathbf{V}_m, \quad \forall (i, j) \in \mathcal{E}, \end{aligned}$$

and  $\mathbf{L}_{ij} = 0$  otherwise. As  $\mathbf{L}_{ij}$  is symmetric, it means that this particular choose of  $\mathbf{Z}_{ij}$  is valid.

Note that  $\mathbf{V}_m (\hat{\mathbf{w}}_{ij} \hat{\mathbf{w}}_{ij}^T + \hat{\mathbf{w}}_{ji} \hat{\mathbf{w}}_{ji}^T) \mathbf{V}_m \succeq 0$ , it means that we can drop this term to obtain a slightly weaker exact recovery condition, which is summarized below

**Corollary 3.** Define block Laplacian matrix  $\mathbf{L}^d \in \mathbb{R}^{nm \times nm}$  as

$$\mathbf{L}_{ij}^d = \begin{cases} \sum_{k \neq i} \text{Diag}(\mathbf{1} - (\mathbf{w}_{ik} + \mathbf{w}_{ki})/2) & i = j \\ -\text{Diag}(\mathbf{1} - (\mathbf{w}_{ik} + \mathbf{w}_{ki})/2) & i \neq j \end{cases}$$

Then  $\mathbf{X}^{gt}$  is the unique optimal solution to (7) if

$$\mathbf{y}^T \mathbf{L}^d \mathbf{y} > 0, \quad \forall 0 \neq \mathbf{y} \in \mathcal{N}_{n,m}^\perp.$$

**Proof of Theorem 5.1** We prove that the condition of Corollary 3 is satisfied. Suppose this is not the case, then  $\exists \mathbf{0} \neq \mathbf{y} \in \mathcal{N}_{n,m}^\perp$  such that

$$\mathbf{L}^d \mathbf{y} \neq \mathbf{0}.$$

Reshape  $\mathbf{y}$  into a  $m \times n$  matrix  $\mathbf{Y} = (\mathbf{y}_1, \dots, \mathbf{y}_m)^\top$ , we have

$$(\mathbf{L}_G - 2\mathbf{L}_{G_s^{\text{false}}})\mathbf{y}_s \neq \mathbf{0}, \quad \mathbf{e}^T \mathbf{y}_i = 0, \quad 1 \leq s \leq m.$$

As  $\lambda_2(\mathbf{L}_G - 2\mathbf{L}_{G_s^{\text{false}}}) > 0$  and  $\lambda_1(\mathbf{L}_G - 2\mathbf{L}_{G_s^{\text{false}}}) = 0$ , it follows that  $\mathbf{y}_s = c_s \mathbf{1}$ . Together with  $\mathbf{1}^\top \mathbf{y}_s = 0$  we obtain  $c_s = 0, 1 \leq s \leq m$ , which violate  $\mathbf{0} \neq \mathbf{y}$ .  $\square$