

# Deep Functional Maps for Simultaneously Computing Direct and Symmetric Correspondences of 3D Shapes

Hui Wang

Shijiazhuang Tiedao University  
Shijiazhuang City, Hebei Province, P.R.China  
wanghui198411109@163.com

Junjie Cao

Dalian University of Technology  
Dalian City, Liaoning Province, P.R.China  
jjcao@dlut.edu.cn

Bitao Ma

Shijiazhuang Tiedao University  
Shijiazhuang City, Hebei Province, P.R.China  
btma308@163.com

Xiuping Liu

Dalian University of Technology  
Dalian City, Liaoning Province, P.R.China  
xpliu@dlut.edu.cn

Hui Huang

Shenzhen University  
Shenzhen City, Guangdong Province, P.R.China  
hhzhiyan@gmail.com

## Abstract

We introduce a novel method of isometric correspondences for 3D shapes, designed to address the problem of multiple solutions associated with deep functional maps when matching shapes with left-to-right reflectional intrinsic symmetries. Unlike the existing methods that only find the direct correspondences using single Siamese network, our proposed method is able to detect both the direct and symmetric correspondences among shapes simultaneously. Furthermore, our method detects the reflectional intrinsic symmetry of each shape. Key to our method is the using of two Siamese networks that learn consistent direct descriptors and their symmetric ones, combined with carefully designed regularized functional maps and supervised loss. This leads to the first deep functional map capable of both producing two high-quality correspondences of shapes and detecting the left-to-right reflectional intrinsic symmetry of each shape. Extensive experiments demonstrate that the proposed method obtains more accurate results than state-of-the-art methods for shape correspondences and reflectional intrinsic symmetries detection.

## 1. Introduction

The problem of shape correspondence is fundamental in the fields of computer vision and computer graphics with wide applications in deformation transfer[37], statistical shape analysis[5] and so on. While the most classi-

cal non-rigid shape correspondence methods are based on hand-crafted features or deformation models [43], more recent approaches have focused on *learning* an optimal model directly from 3D data. This includes approaches based on template shape deformation [14] and methods that exploit different definitions of convolution and phrase correspondence as a dense labeling problem [46, 40, 9].

One prominent direction in learning-based shape correspondence is the *deep functional map* either in supervised [20, 10, 36] or unsupervised setting [15, 33], where one Siamese network of two branches of neural networks with shared weights learns geometrical features that recover optimal represented matrix of the functional map [26]. Despite significant progress in this area, a key drawback of these methods is that cannot address the *multiple solutions* problem, that is when matching a pair of shapes with left-to-right reflectional intrinsic symmetries, such as a pair of humans or animals, there exist two exact solutions, i.e., direct (left-to-left) and symmetric (left-to-right) correspondences. The previous methods [20, 33, 15, 10] only can predict the direct correspondence. Furthermore, the above deep functional maps cannot be used for intrinsic symmetry detection. When the two input shapes are the same, the shared weight Siamese network making the computed represented matrix of the functional map is the identical matrix, which lead to the identical mapping.

Based on the above limitations, we introduce a novel deep functional map for computing the two correspondences and reflectional intrinsic symmetry of each shape illustrated in Figure 1. Given a pair of shapes, two Siamese

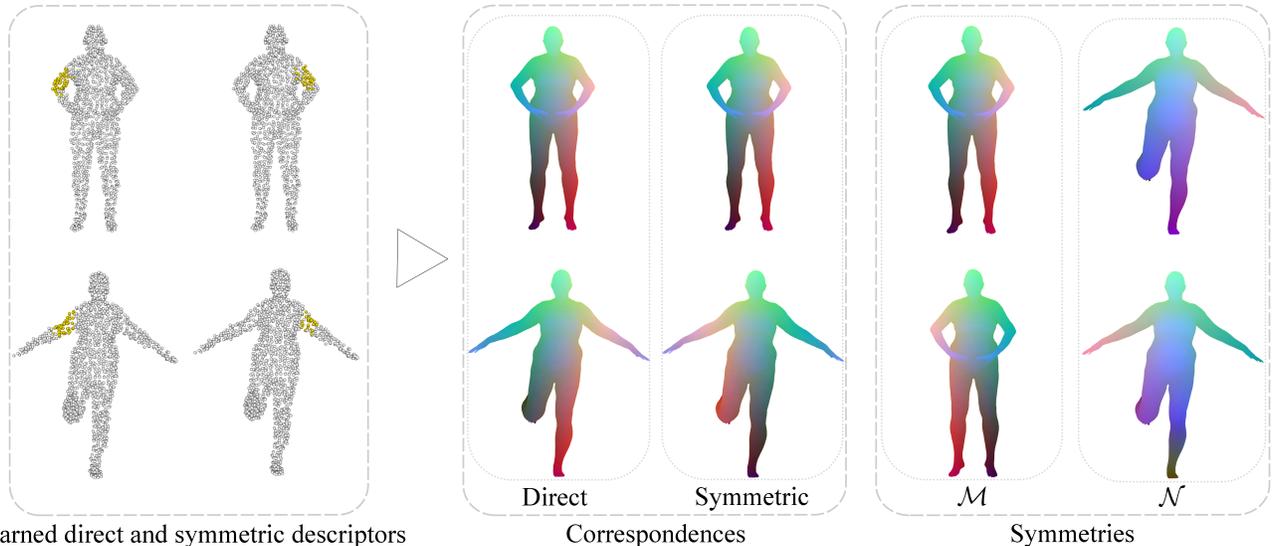


Figure 1. Given a pair of shapes, our method learns consistent direct descriptors and their symmetric ones (left), and automatically computes the direct and symmetric correspondences among the shapes (middle) and intrinsic symmetry of each shape (right). The corresponding points are indicated through the same colors in each column.

networks are used in our work, where the first one with shared weights learns consistent descriptors for the two shapes, while the second one with another shared weights learns symmetric descriptors of ones learned in the first Siamese network. Then the above learned descriptors are fed into the regularized functional map layers to compute both the direct and symmetric correspondences among the two input shapes and reflectional intrinsic symmetry of each shape. Our main technical contributions are as follows:

- We introduce a novel deep functional map with two Siamese networks combined with carefully designed regularized functional maps and supervised loss for learning consistent direct descriptors and their symmetric ones on 3D shapes.
- The proposed method is the first deep functional map that can solve the problem of multiple solutions when matching shapes with left-to-right reflectional intrinsic symmetries. Furthermore, our approach detects the reflectional intrinsic symmetry of each shape.
- Extensive experiments show that our method achieves better results than previous works of shape correspondences and reflectional intrinsic symmetries detection.

### 1.1. Related works

Computing correspondences between 3D shapes is a well-studied area of computer vision and computer graphics. Below we only review the most closely related methods and refer the interested readers to recent surveys including [43, 4, 34] for more in-depth discussions.

**Functional maps.** Our method is built on the functional map representation, which was originally introduced in [26] for solving non-rigid shape matching problems, and then extended in follow-up works [27]. The functional map expresses correspondences as smaller matrices encoded in a reduced basis, which converts complex geometry problems into simpler optimization of linear algebra.

A range of recent works, including [1, 19, 17, 11, 8, 31, 25] among many others, have extended the generality and improved the robustness of the functional map estimation pipeline, by using different regularizers, robust penalties and powerful post-processing methods. Nevertheless, a common problem with the above non-learning approaches is their over-reliance on manually choice of descriptors, which greatly restricts the results of correspondences.

**Deep functional maps.** Inspired by the deep geometry learning [47], many recent methods [20, 15, 33, 10, 35] have proposed to learn the optimal features by deep functional maps.

FMNet [20] introduces a Siamese network to learn optimal consistent descriptors on the two shapes for computing the functional map. This architecture is based on optimizing a non-linear transformation of SHOT descriptors to obtain functional map that is as close as possible to the ground truth correspondence. Follow-up works have extended this approach to the unsupervised setting [15, 33] by replacing supervised loss with structural properties of the resulting maps, but still use pre-defined descriptors for optimization. Most recently, [10, 35] have proposed to learn descriptors directly from the raw 3D data without relying on pre-defined descriptors, resulting in significantly more

robust and accurate methods.

Despite significant progress, above approaches only can compute the direct correspondence for shapes with reflectional intrinsic symmetries. The proposed deep functional map is the first one that can compute both direct and symmetric correspondences.

**Multiple solutions of correspondences.** When computing an intrinsic isometry correspondence between a pair of symmetric shapes, such as a pair of humans or animals, there exists multiple equally good solutions. Some previous works have been proposed to address the above multiple solution problem [28, 30, 29, 39].

Performing functional map in an appropriate quotient space is able to find the multiple correspondences [28]. However, all intrinsic symmetries of one shape should be given in advance. The proposed orientation-preserving and orientation-reversing energies are added as soft constraints in the functional map for computing direct and symmetric correspondences in [30]. Nevertheless, the orientation-reversing energy is usually not strong enough to obtain desirable symmetric correspondence. A compact tree structure based on the functional map is introduced for encoding and enumerating possible rough initializations of correspondence in [29], which can be refined to produce multiple high-quality correspondences. A computational framework for joint symmetry and map synchronization is proposed using a lifting map representation [39].

Unlike the above methods based on hand-crafted features, in this paper we propose a novel learning-based approach for solving the problem of multiple solutions of correspondences for shapes with reflectional symmetries.

**Intrinsic symmetry detection.** Symmetry is a distance preserving self-homeomorphism of a shape and reflect high-level information about shape structure. Thus, symmetry detection has received significant attentions in computational geometry and computer graphics [23, 13], especially intrinsic symmetry detection via functional maps [21, 45, 24, 30, 29].

Functional maps with constraints of orthogonality [21] and block diagonal structure [45] on represented matrix are introduced for intrinsic symmetry detection. Because the symmetry is an isometry on the shape itself, the shape correspondence methods based on functional maps [30, 29], can also be used for symmetry detection when the two input shapes are the same. However, the above methods are dependent on the hand-crafted features. In this paper, a novel end-to-end deep functional map is proposed for reflectional intrinsic symmetry detection.

It is noted that the previous deep functional maps cannot be used for intrinsic symmetry detection [20, 33, 15, 10]. Because, when the two input shapes are the same, the shared weight Siamese network in the above deep functional maps making the computed represented matrix of the function-

al map is the identical matrix, which lead to the identical mapping.

## 1.2. Background and motivation

In this section, we give a brief overview of the functional map representation and deep functional map for computing correspondences of 3D shapes. We then provide the motivation of the proposed method.

## 1.3. Functional maps

The proposed method is based on the functional map representation [26]. For completeness, we briefly summarize the pipeline for estimating functional maps, and refer the interested reader to a recent course [27] for a more in-depth discussion.

Given a pair of 3D shapes,  $\mathcal{M}$  and  $\mathcal{N}$ , represented in a discrete setting as triangle meshes, and containing  $m$  and  $n$  vertices respectively, the main steps of the functional map framework for computing correspondences are as follows:

1. The first few eigenfunctions of the discrete Laplace-Beltrami operator are computed on each shape, namely  $k_{\mathcal{M}}$  and  $k_{\mathcal{N}}$  basis functions  $\Phi^{\mathcal{M}}$  and  $\Phi^{\mathcal{N}}$  respectively.
2. A set of descriptor functions on each shape that are expected to be approximately preserved by the unknown correspondence are computed. For instance, a descriptor function can correspond to a particular dimension of the Heat [38] or Wave [3] Kernel Signatures computed at every point. Their coefficients in the respective basis  $\Phi^{\mathcal{M}}$  and  $\Phi^{\mathcal{N}}$  are stored as columns of matrices  $\mathbf{A}$  and  $\mathbf{B}$ .
3. The optimal represented matrix  $\mathbf{C}_{opt}$  of the functional map is then computed by solving the following optimization problem:

$$\mathbf{C}_{opt} = \min_{\mathbf{C}} E_{desc}(\mathbf{C}) + \lambda E_{reg}(\mathbf{C}), \quad (1)$$

where the first term aims at the descriptor preservation:  $E_{desc} = \|\mathbf{C}\mathbf{A} - \mathbf{B}\|^2$ , whereas the second term regularizes the map by promoting the correctness of its overall structural properties. The simplest approach penalizes the failure of the unknown functional map to commute with the Laplace-Beltrami operators:

$$E_{reg}(\mathbf{C}) = \|\mathbf{C}\Delta_{\mathcal{M}} - \Delta_{\mathcal{N}}\mathbf{C}\|^2, \quad (2)$$

where  $\Delta_{\mathcal{M}}$  and  $\Delta_{\mathcal{N}}$  are diagonal matrices of the Laplace-Beltrami eigenvalues on the two shapes.

4. The estimated functional map  $\mathbf{C}_{opt}$  in the spectral domain is converted to a spatial point-to-point correspondence. Usually, a post-processing refinement process is needed [32, 12, 30, 22]. The refinement iteratively takes the map from spectral to spatial domain, until it reaches a local optimum.



Figure 2. Some representative learned direct and symmetric descriptors. In each rectangle, the left column shows the consistent direct descriptors  $F_1$  and  $G_1$  in pairs, and the right column illustrates  $F_2$  and  $G_2$  in pairs, which are symmetric with  $F_1$  and  $G_1$ .

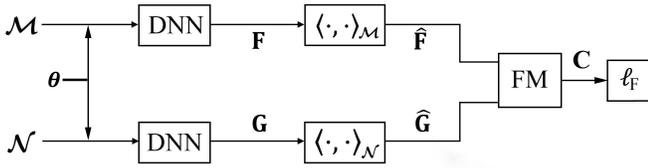


Figure 3. Network architecture of previous deep functional maps [20, 33, 15, 10].

#### 1.4. Deep functional maps

Despite its simplicity and efficiency, the above functional map is fundamentally error prone due to the initial choice of descriptor functions. To alleviate this dependence, several approaches of deep functional maps have been proposed to learn the optimal descriptors from data [20, 33, 15, 10, 35].

More specifically, the deep functional map framework uses the Deep Neural Network (DNN) with the standard Siamese setting for descriptors learning illustrated in Figure 3, in which two copies of the network with shared parameters produces the learned consistent descriptors  $F$  and  $G$  on shapes  $\mathcal{M}$  and  $\mathcal{N}$ . Then, the represented matrix  $C$  is computed with the Functional Map (FM) framework in Section 1.3 with a trained loss  $\ell_F$ . We summarize the previous works of deep functional maps via the type of used DNN, trained loss and choice of training.

**Type of DNN.** The first class of used DNN in previous works [20, 33, 15] is the residual network [16], which aims

at learning a non-linear transformation on the pre-computed descriptors (typically SHOT [42]). The second class of DNN is the KPConv [41] that learns descriptors directly from raw shape geometry [10, 35].

**Trained loss.** The trained loss  $\ell_F$  is defined on spectral [33, 10] or spatial [20, 15] domain. The spectral loss directly builds on the functional map  $C$  obtained by the FM block, while the spatial one converts  $C$  to a soft correspondence and minimizes the spatial pairwise geodesic distance distortion.

**Choice of training.** The loss  $\ell_F$  can be trained in supervised [20, 10] or unsupervised [33, 15] manner. In the supervised situation, ground truth of the point-to-point correspondence  $T^{gt}$  [20] or spectral functional map  $C^{gt}$  [10] should be known for the training shape pairs. While in the unsupervised methods, the loss is enforced spatial geodesic distances preserving criterion during isometric correspondence [15] or the desired structural properties on the resulting represented matrix  $C$  of functional map, such as its bijection orthogonality and commutability with the Laplacian [33]. A weakly supervised deep functional map is also proposed for correspondence [35].

#### 1.5. Motivation

Many shape classes, such as organic animal and human, naturally have a left-to-right reflectional intrinsic symmetry. Such symmetry implies that two equally-likely correspondences exist on pairs of shapes, i.e., the direct (left-to-left)

and symmetric (left-to-right) correspondences. Although very powerful, the above previous deep functional map only aims at recovering the direct correspondence. In this paper, given a pair of shapes, we propose the first deep functional map that can compute both the direct and symmetric correspondences among of shapes and reflectional symmetry of each shape at the same time.

## 2. Method

In this section, we propose a novel approach to learn both direct and their symmetric descriptors on a pair of shapes in order to compute two equally-likely correspondences, which are left-to-left direct and left-to-right symmetric maps, and reflectional intrinsic symmetry of each shape through the functional map framework shown in Figure 4. The key difference between our method and previous deep functional maps [20, 33, 15, 10] is that two Siamese networks on the source and target shapes are used to learn consistent direct descriptors and their symmetric ones.

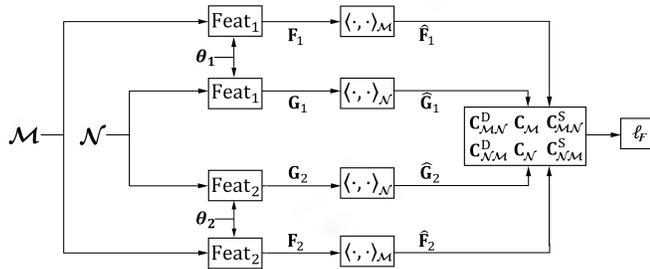


Figure 4. Overview of our deep functional map. Given a pair of shapes  $\mathcal{M}$  and  $\mathcal{N}$ , the first Siamese network with weight  $\theta_1$  learns consistent direct descriptors  $\mathbf{F}_1$  and  $\mathbf{G}_1$ , and the second Siamese network with weight  $\theta_2$  learns descriptors  $\mathbf{F}_2$  and  $\mathbf{G}_2$  which are symmetric with  $\mathbf{F}_1$  and  $\mathbf{G}_1$ . The above descriptors are fed into the functional map to compute the correspondences and symmetries with a supervised loss  $\ell_F$ .

### 2.1. Architecture

Given a pair of 3D shapes  $\mathcal{M}$  and  $\mathcal{N}$  each with one reflectional symmetry, the proposed method is mainly composed of two parts illustrated in Figure 4, feature extractor in section 2.2 and regularized functional map layer in section 2.3.

The first part, labeled as  $\text{Feat}_1$  and  $\text{Feat}_2$  in Figure 4, aims at extracting features of the raw geometry of the shapes. However, unlike the previous works of deep functional maps [20, 33, 15, 10] using only one Siamese network, we use two Siamese networks on the source and target shapes with shared learnable parameters  $\theta_1$  and  $\theta_2$  to learn consistent direct descriptors and their symmetric ones separately. Secondly, the above learned descriptors are projected in the spectral bases of the shapes and fed to the regularized func-

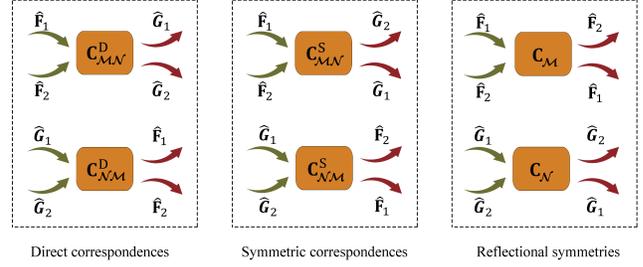


Figure 5. The corresponded relations between spectral descriptors and six represented matrices of functional maps of correspondences and symmetries.

tional map layer to compute the functional maps of direct correspondences  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^D$ ,  $\mathbf{C}_{\mathcal{N}\mathcal{M}}^D$  in two direction; symmetric correspondences  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^S$ ,  $\mathbf{C}_{\mathcal{N}\mathcal{M}}^S$  in two direction; and symmetries  $\mathbf{C}_{\mathcal{M}}$ ,  $\mathbf{C}_{\mathcal{N}}$  of  $\mathcal{M}$  and  $\mathcal{N}$  at the same time. Finally, we use a spectral loss in section 2.4 based on the differences between the computed and the ground truth functional maps of correspondences and symmetries.

### 2.2. The feature extractor

The goal of this part is to learn functional characterizations of point clouds that will later be used to compute spectral descriptors and then functional maps. We use two Siamese networks rather than using the only one in the previous works [20, 33, 15, 10]. The first Siamese network  $\text{Feat}_1$  is applied with the same weights  $\theta_1$  on the source shape  $\mathcal{M}$  and target one  $\mathcal{N}$  to obtain consistent descriptors  $\mathbf{F}_1$  and  $\mathbf{G}_1$ , which are preserved during the direct correspondences illustrated. The second Siamese network  $\text{Feat}_2$  with another shared weights  $\theta_2$  is used to learn descriptors  $\mathbf{F}_2$  and  $\mathbf{G}_2$ , which are symmetric with  $\mathbf{F}_1$  and  $\mathbf{G}_1$  on shapes  $\mathcal{M}$  and  $\mathcal{N}$  separately. Some represented learn descriptors of  $\mathbf{F}_1$ ,  $\mathbf{F}_2$ ,  $\mathbf{G}_1$  and  $\mathbf{G}_2$  are illustrated in Figure 2, where the descriptors are localized and seem to highlight one specific part of the body (first rectangle for foot, second for arm, third for thigh).

Like the previous deep functional map [10], in this part we also use the state-of-the-art point cloud learning method KPConv [41], by extending the segmentation network proposed in that work, as the feature extractor in the branch of the above two Siamese networks. The KPConv uses specific defined convolution and pooling operators on point cloud to learn features. Please refer to the original work for the network details [41].

### 2.3. The functional map layer

For the given pair of 3D shapes  $\mathcal{M}$  and  $\mathcal{N}$ , the main goal of this section is to compute correspondences between shapes in both directions, i.e., by treating the shapes as both source and target, and symmetries of each shape from the computed raw-data features  $\mathbf{F}_1$ ,  $\mathbf{G}_1$ ,  $\mathbf{F}_2$  and  $\mathbf{G}_2$  in section

Method/Dataset	F	S	F on S	S on F
BCICP	7.4	12.	*	*
ZoomOut	23.	24.	*	*
SURFMNet	6.8	3.5	23.	23.
SURFMNet + ICP	4.3	<b>2.4</b>	13.	13.
Unsup FMNet	8.6	5.0	21.	21.
Unsup FMNet + PMF	11.	8.3	14.	13.
FMNet	11.	28.	39.	39.
FMNet + PMF	6.8	14.	15.	16.
GeomFmap	3.2	4.7	12.	3.5
GeomFmap + ZO	<b>1.9</b>	3.5	10.	2.0
Ours	3.1	3.6	11.	3.3
Ours + ZO	<b>1.9</b>	<b>2.7</b>	<b>9.1</b>	<b>1.9</b>

Table 1. Comparative results (\*100) of the different methods for the direct correspondence.

**2.2.** The functional maps of direct correspondences in two directions are denoted as  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^D$  and  $\mathbf{C}_{\mathcal{N}\mathcal{M}}^D$ , symmetric correspondences are represented as  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^S$  and  $\mathbf{C}_{\mathcal{N}\mathcal{M}}^S$ , and reflectional symmetries of  $\mathcal{M}$  and  $\mathcal{N}$  are respectively denoted as  $\mathbf{C}_{\mathcal{M}}$  and  $\mathbf{C}_{\mathcal{N}}$ . We stress again that in this block the above six functional maps are computed simultaneously shown in Figure 4.

Firstly, we express the computed feature function in the respective spectral basis  $\Phi^{\mathcal{M}}$  and  $\Phi^{\mathcal{N}}$  on  $\mathcal{M}$  and  $\mathcal{N}$ . This leads to the spectral descriptors  $\hat{\mathbf{F}}_1 = (\Phi^{\mathcal{M}})^\dagger \mathbf{F}_1$ ,  $\hat{\mathbf{F}}_2 = (\Phi^{\mathcal{M}})^\dagger \mathbf{F}_2$ ,  $\hat{\mathbf{G}}_1 = (\Phi^{\mathcal{N}})^\dagger \mathbf{G}_1$  and  $\hat{\mathbf{G}}_2 = (\Phi^{\mathcal{N}})^\dagger \mathbf{G}_2$ , where  $(\Phi^{\mathcal{M}})^\dagger$  and  $(\Phi^{\mathcal{N}})^\dagger$  are the Moore pseudo inverses of  $\Phi^{\mathcal{M}}$  and  $\Phi^{\mathcal{N}}$  respectively. This step is where we shift focus from the spatial to the spectral domain corresponded to the dot product blocks in Figure 4.

Next, we use the spectral descriptors  $\hat{\mathbf{F}}_1$ ,  $\hat{\mathbf{F}}_2$ ,  $\hat{\mathbf{G}}_1$  and  $\hat{\mathbf{G}}_2$  to carefully compute the six functional maps, i.e., direct correspondences of  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^D$ ,  $\mathbf{C}_{\mathcal{N}\mathcal{M}}^D$ ; symmetric correspondences of  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^S$ ,  $\mathbf{C}_{\mathcal{N}\mathcal{M}}^S$ ; and reflectional symmetries  $\mathbf{C}_{\mathcal{M}}$  and  $\mathbf{C}_{\mathcal{N}}$ . It should be emphasized that the descriptors of  $\hat{\mathbf{F}}_1$ ,  $\hat{\mathbf{F}}_2$  on shape  $\mathcal{M}$  are corresponded to  $\hat{\mathbf{G}}_1$ ,  $\hat{\mathbf{G}}_2$  on shape  $\mathcal{N}$  respectively during the direct correspondences of  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^D$  and  $\mathbf{C}_{\mathcal{N}\mathcal{M}}^D$ ;  $\hat{\mathbf{F}}_1$  is reflectional symmetric with  $\hat{\mathbf{F}}_2$  on shape  $\mathcal{M}$ ; and  $\hat{\mathbf{G}}_1$  is reflectional symmetric with  $\hat{\mathbf{G}}_2$  on shape  $\mathcal{N}$ . The relations between the spectral descriptors  $\hat{\mathbf{F}}_1$ ,  $\hat{\mathbf{F}}_2$ ,  $\hat{\mathbf{G}}_1$  and  $\hat{\mathbf{G}}_2$  and six functional maps are illustrated in Figure 5. Thus, the minimized energies of the six regularized functional maps are as follows:

#### Direct correspondences.

$$\begin{aligned}
\mathbf{C}_{\mathcal{M}\mathcal{N}}^D &= \min_{\mathbf{C}} \|\mathbf{C}(\hat{\mathbf{F}}_1, \hat{\mathbf{F}}_2) - (\hat{\mathbf{G}}_1, \hat{\mathbf{G}}_2)\|^2 + \\
&\quad \lambda \|\mathbf{C}\Delta_{\mathcal{M}} - \Delta_{\mathcal{N}}\mathbf{C}\|^2, \\
\mathbf{C}_{\mathcal{N}\mathcal{M}}^D &= \min_{\mathbf{C}} \|\mathbf{C}(\hat{\mathbf{G}}_1, \hat{\mathbf{G}}_2) - (\hat{\mathbf{F}}_1, \hat{\mathbf{F}}_2)\|^2 + \\
&\quad \lambda \|\mathbf{C}\Delta_{\mathcal{N}} - \Delta_{\mathcal{M}}\mathbf{C}\|^2.
\end{aligned} \tag{3}$$

#### Symmetric correspondences.

$$\begin{aligned}
\mathbf{C}_{\mathcal{M}\mathcal{N}}^S &= \min_{\mathbf{C}} \|\mathbf{C}(\hat{\mathbf{F}}_1, \hat{\mathbf{F}}_2) - (\hat{\mathbf{G}}_2, \hat{\mathbf{G}}_1)\|^2 + \\
&\quad \lambda \|\mathbf{C}\Delta_{\mathcal{M}} - \Delta_{\mathcal{N}}\mathbf{C}\|^2, \\
\mathbf{C}_{\mathcal{N}\mathcal{M}}^S &= \min_{\mathbf{C}} \|\mathbf{C}(\hat{\mathbf{G}}_1, \hat{\mathbf{G}}_2) - (\hat{\mathbf{F}}_2, \hat{\mathbf{F}}_1)\|^2 + \\
&\quad \lambda \|\mathbf{C}\Delta_{\mathcal{N}} - \Delta_{\mathcal{M}}\mathbf{C}\|^2.
\end{aligned} \tag{4}$$

#### Reflectional symmetries.

$$\begin{aligned}
\mathbf{C}_{\mathcal{M}} &= \min_{\mathbf{C}} \|\mathbf{C}(\hat{\mathbf{F}}_1, \hat{\mathbf{F}}_2) - (\hat{\mathbf{F}}_2, \hat{\mathbf{F}}_1)\|^2 + \\
&\quad \lambda \|\mathbf{C}\Delta_{\mathcal{M}} - \Delta_{\mathcal{M}}\mathbf{C}\|^2, \\
\mathbf{C}_{\mathcal{N}} &= \min_{\mathbf{C}} \|\mathbf{C}(\hat{\mathbf{G}}_1, \hat{\mathbf{G}}_2) - (\hat{\mathbf{G}}_2, \hat{\mathbf{G}}_1)\|^2 + \\
&\quad \lambda \|\mathbf{C}\Delta_{\mathcal{N}} - \Delta_{\mathcal{N}}\mathbf{C}\|^2.
\end{aligned} \tag{5}$$

Thirdly, we compute the above six linear square minimization problems of equations (3), (4) and (5) using the first one as example, the others are similarly. As in the previous work [10], the matrix  $\mathbf{C}_{\mathcal{M}\mathcal{N}}^D$  is computed by each row  $\mathbf{c}_i$  to a separate linear system:

$$(\hat{\mathbf{F}}_1, \hat{\mathbf{F}}_2)(\hat{\mathbf{F}}_1, \hat{\mathbf{F}}_2)^T + \lambda \text{diag}((\mu_i^{\mathcal{M}} - \mu_i^{\mathcal{N}})^2) \mathbf{c}_i = (\hat{\mathbf{F}}_1, \hat{\mathbf{F}}_2) \mathbf{b}_i, \tag{6}$$

where  $\mu_i^{\mathcal{M}}$  and  $\mu_i^{\mathcal{N}}$  respectively correspond to the  $i^{\text{th}}$  eigenvalues of the Laplace-Beltrami operators of  $\mathcal{M}$  and  $\mathcal{N}$ , and  $\mathbf{b}_i$  is the  $i^{\text{th}}$  row of  $(\hat{\mathbf{G}}_1, \hat{\mathbf{G}}_2)$ .

#### 2.4. The supervised spectral loss

Similar to the deep functional map in [10], the proposed method also uses a loss with respect to the ground truth functional map in the spectral domain. However, in addition to labels in direct correspondences ( $\mathbf{C}_{\mathcal{M}\mathcal{N}}^D$ )<sup>gt</sup> and ( $\mathbf{C}_{\mathcal{N}\mathcal{M}}^D$ )<sup>gt</sup> referred to [10], we add symmetric correspondences ground truths ( $\mathbf{C}_{\mathcal{M}\mathcal{N}}^S$ )<sup>gt</sup> and ( $\mathbf{C}_{\mathcal{N}\mathcal{M}}^S$ )<sup>gt</sup> and reflectional symmetries ground truths ( $\mathbf{C}_{\mathcal{M}}$ )<sup>gt</sup> and ( $\mathbf{C}_{\mathcal{N}}$ )<sup>gt</sup> to optimize the Siamese networks. The loss is defined as sum of the Frobenius norm of differences between the above ground truth functional maps and the computed six functional maps in the section 2.3.

$$\begin{aligned}
\ell_F &= \|\mathbf{C}_{\mathcal{M}\mathcal{N}}^D - (\mathbf{C}_{\mathcal{M}\mathcal{N}}^D)^{\text{gt}}\|^2 + \|\mathbf{C}_{\mathcal{N}\mathcal{M}}^D - (\mathbf{C}_{\mathcal{N}\mathcal{M}}^D)^{\text{gt}}\|^2 + \\
&\quad \|\mathbf{C}_{\mathcal{M}\mathcal{N}}^S - (\mathbf{C}_{\mathcal{M}\mathcal{N}}^S)^{\text{gt}}\|^2 + \|\mathbf{C}_{\mathcal{N}\mathcal{M}}^S - (\mathbf{C}_{\mathcal{N}\mathcal{M}}^S)^{\text{gt}}\|^2 + \\
&\quad \|\mathbf{C}_{\mathcal{M}} - (\mathbf{C}_{\mathcal{M}})^{\text{gt}}\|^2 + \|\mathbf{C}_{\mathcal{N}} - (\mathbf{C}_{\mathcal{N}})^{\text{gt}}\|^2.
\end{aligned} \tag{7}$$

#### 2.5. Postprocessing

Once the proposed model is trained, we test it on a pair of shapes and get a functional map, which can either directly be converted to a point-to-point correspondence or refined further. We use the ZoomOut [22] refining algorithm based on navigating between spatial and spectral domains while

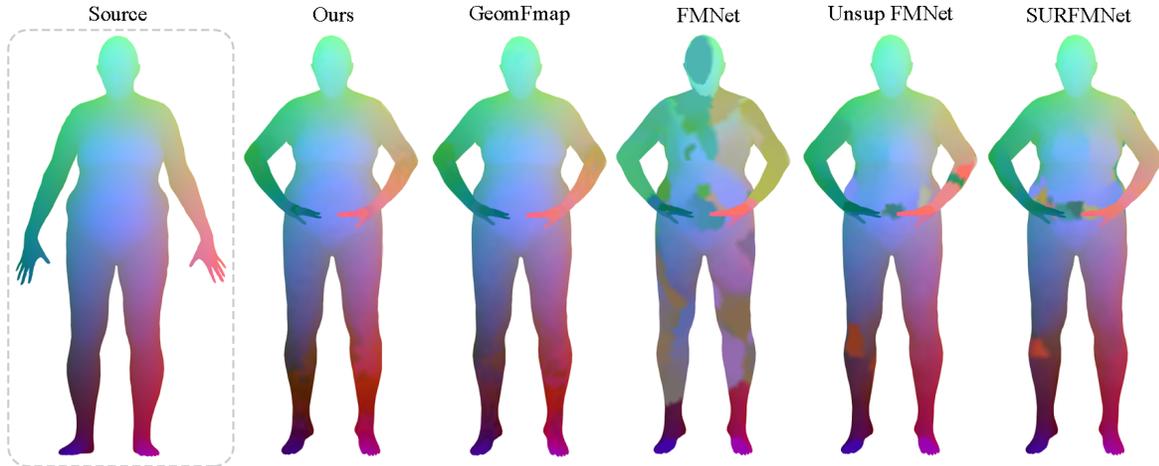


Figure 6. Comparisons on the FAUST re-meshed for direct correspondence without refinement.

progressively increasing the number of spectral basis functions from 30 to 100. By virtue of this optimization algorithm, we have achieved better experimental results shown in section 3.

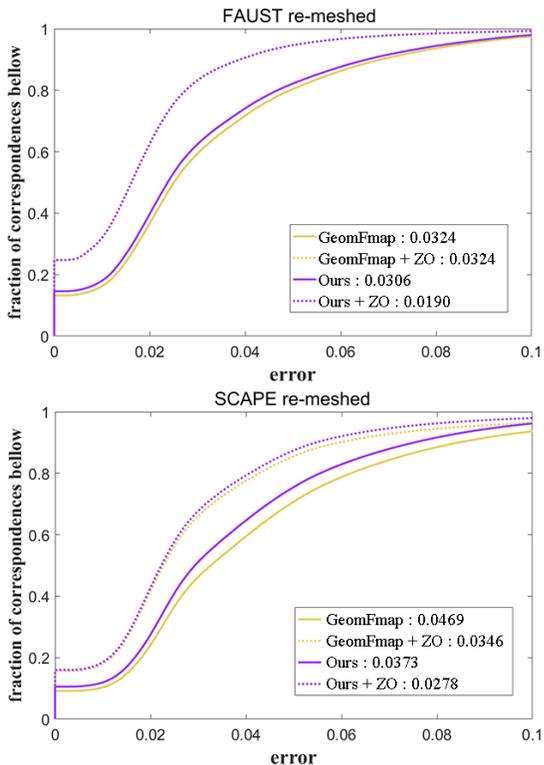


Figure 7. Geodesic distance error curves for the direct correspondence.

### 3. Experimental results

This section explains the implementations of our method and experimental results. Section 3.1 describes the datasets and evaluation measurement. Section 3.2 is the implementation detail. Sections 3.3, 3.4 and 3.5 are the comparisons results with previous works for direct correspondences, symmetric correspondences and reflectional symmetries. Finally, section 3.6 shows robustness of the proposed method. We show the shape matching by color transfer, where corresponding points on the source and target shapes have same colors.

#### 3.1. Datasets and evaluation measurement

We test our method on a wide spectrum of human datasets. First, the original FAUST [5] and SCAPE [2] datasets containing 100 and 71 shapes. We also highlight that the SCAPE

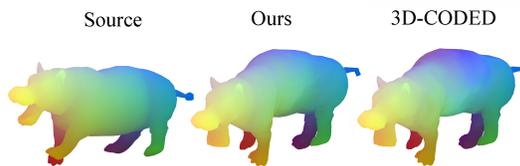


Figure 8. Comparisons on the SMAL [48] models for the direct correspondence with refinement.

dataset is slightly more challenging since the shapes are less regular, and two shapes never share the same pose. For convenience, the original FAUST and SCAPE are represented as O-F and O-S respectively. Second, the re-meshed versions of FAUST dataset containing shapes in 1-1 correspondence, and of SCAPE, made publicly available by Ren et al. [30]. These re-meshed datasets offer significantly more variability in terms of shape structures and connectivity, in-

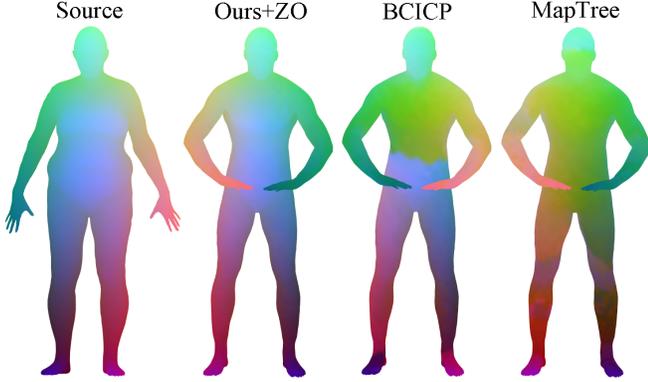


Figure 9. Comparisons on the FAUST re-meshed for the symmetric correspondence with refinement.

Method/Dataset	F	S
BCICP	19.	15.
MapTree	11.	6.3
Ours	<b>3.0</b>	<b>3.7</b>

Table 2. Comparative results (\*100) of the different methods for the symmetric correspondence.

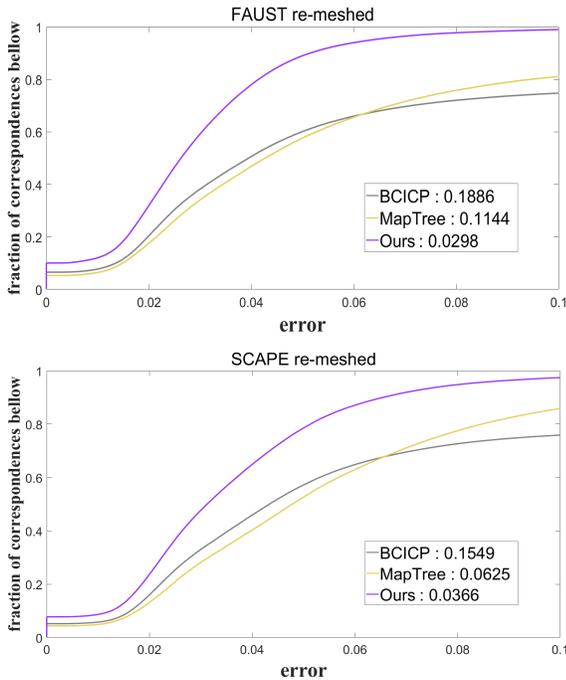


Figure 10. Geodesic distance error curves for the symmetric correspondence.

cluding for instance point sampling density, making them harder to match for existing algorithms. The re-meshed version of FAUST and SCAPE are denoted as F and S respectively. Third, hippopotamus generated by SMAL [48] with 5000 for training and 25 for test. These hippopotamus

Method/Dataset	F	S	O-F	O-S
GIS	7.3	*	10.	7.3
BCICP	4.4	4.0	2.6	<b>6.6</b>
MapTree	3.7	5.7	4.7	6.7
Ours	<b>2.6</b>	<b>3.5</b>	<b>2.2</b>	7.3

Table 3. Comparative results (\*100) of the different methods for reflectional symmetries detection.

are generated using random parameters, which are from a Gaussian distribution of ad-hoc variance 0.2.

We use the measurement introduced in [18] to evaluate the results, where the per-point geodesic distance between the

ground truth correspondence of direct/symmetric map and the computed one is reported. All results are multiplied by 100 for the sake of readability.

### 3.2. Implementation detail

We implement our method in Tensorflow by adapting the open-source implementation of GeomFmap [10]. Like the GeomFmap [10], the size of the functional basis is 30 and the regularizer  $\lambda$  is 0.001 by default. However, our method contains two Siamese networks, each of which shares parameters. We train our network with a batch size of 4 shape pairs for 10000 steps. We use a learning rate of 0.001 and gradually decreasing it to 0.0001 with ADAM optimizer. Following the pipeline of KPConv, we start with a sub-sampled version of the point clouds with a grid sub-sampling of step 0.03.

### 3.3. Results of direct correspondences

In this section, we compare our method with previous works of direct correspondences based on functional map [30, 22, 33, 15, 20, 10] and deep deformation [14].

**Functional map.** We compare our method with several state-of-the-art works via functional map. The first category contains non-learning methods: BCICP [30] and ZoomOut [22]. The second category includes unsupervised deep learning methods: SURFMNet [33] and Unsup FMNet [15] with and without refinement algorithm (ICP [26] for SURFMNet and PMF [44] for Unsup FMNet). The third category includes supervised deep learning methods: FMNet [20] and GeomFmap [10] with and without refinement algorithm (PMF [44] for FMNet and ZoomOut [22] for GeomFmap). For a fair comparison with other methods, we show our results with and without ZoomOut [22] post-processing refinement in direct correspondence. For convenience, we refer to ZoomOut as ZO, and our method without ZoomOut as Ours.

Following the standard protocol, we split the FAUST re-meshed and SCAPE re-meshed into training and test sets containing 80 and 20 shapes for FAUST, and 51 and 20

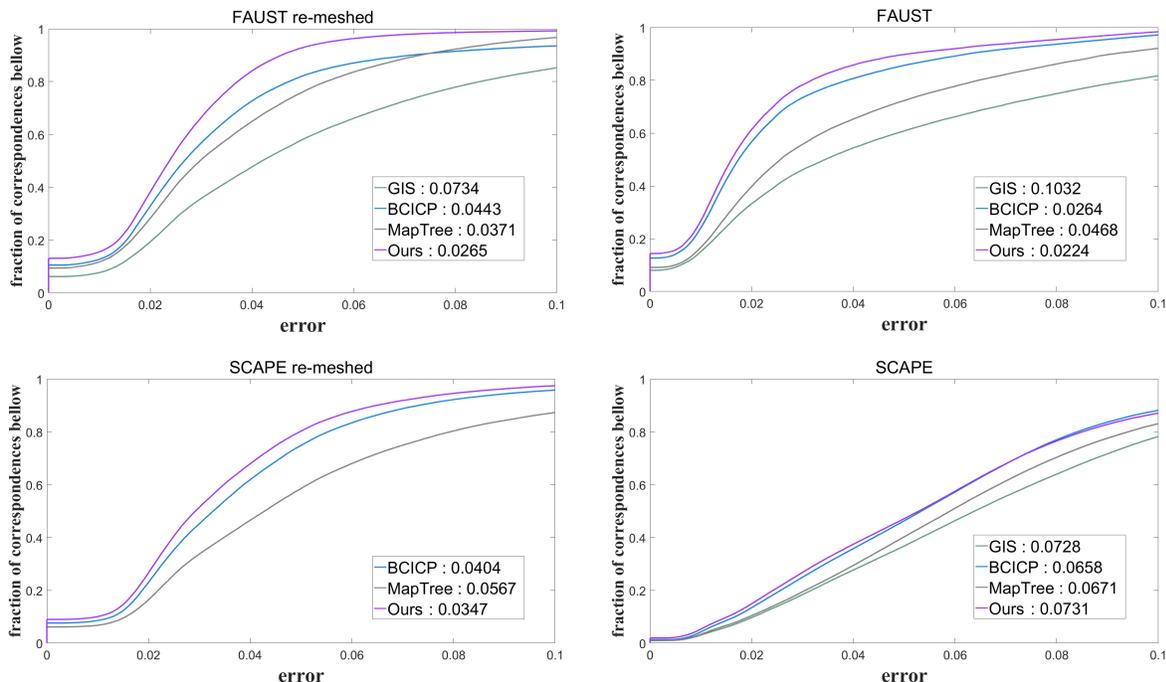


Figure 11. Geodesic distance error curves for the reflectional symmetry detections.

shapes for SCAPE. In Table 1, F and S show the results for training and testing on the same dataset, whereas F on S means trained on FAUST re-meshed and tested on SCAPE re-meshed. It is noted that axiomatic methods do not require a learning process, so we just compare them on the test sets. As evident in Table 1, our method obtains a particularly comparable or superior performance to other methods. Figure 6 is a comparison result without refinement, which shows that our method gets a better performance. Furthermore, as show in Figure 7, we obtain a more accurate performance especially on SCAPE dataset.

**Deep deformation.** In Figure 8, our method is compared with the 3D-CODED [14], which learns deformation of the template for computing direct correspondences. These two methods are all trained on the hippopotamus introduced in section 3.1. The average geodesic distances of our method and 3D-CODED are 3.0 and 1.3. The comparison result shows that our method performs better than 3D-CODED, especially in the back region.

### 3.4. Results of symmetric correspondences

As we all know, exiting deep functional map methods only can detect direct correspondence, so we compare our pipeline with classic axiomatic techniques BCICP [30] and MapTree [29]. Same as the direct correspondence, we choose FAUST re-meshed and SCAPE re-meshed to train and test on the same dataset, while BCICP and MapTree are not learning-based techniques, so we just test them on

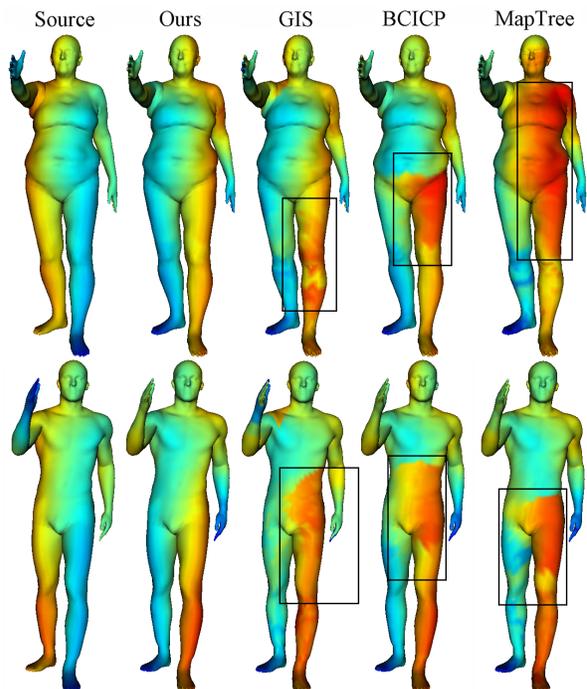


Figure 12. Comparisons on the FAUST for the reflectional symmetry detection with refinement. Corresponding points have the same colors. Black rectangles are used to mark regions with larger errors.

the test sets. And both axiomatic methods include post optimization, so we add ZoomOut to our method in this part and refer it to Ours. As evident in Table 2, our mean geodesic distance error is much lower than others on both FAUST re-meshed and SCAPE re-meshed datasets. Figure 9 demonstrates the comparisons of symmetric correspondences. Figure 10 further shows that our matching results have higher accuracy. This mainly because BCICP and MapTree excessively rely on initial manual descriptor while our method learns suitable descriptors directly from raw 3D data.

### 3.5. Results of reflectional symmetry detections

In this section, we compare our method with previous reflectional symmetry detection methods based on functional maps, i.e., GIS [45], BCICP [30] and MapTree [29]. The training process is the same as the correspondence experiment, in which two shapes are randomly selected from the training set. As for the test, each shape in test sets is fed into the network as the source and target simultaneously and  $C_M$  is eventually selected to evaluate our pipeline. For a fair comparison with other methods, we show our results with ZoomOut [22] refinement. And for the eight detected self-symmetries in the MapTree, we carefully choose the one with the smallest geodesic distances to the ground-truth of left-to-right symmetry. Comparison baselines are all belonged to axiomatic techniques, so we only test them on test sets.

As evident in Table 3, our method is comparable or superior to other baselines. And Figure 11 demonstrates our matching results have higher accuracy. In order to evaluate our method more intuitively, we use rectangle boxes to mark the regions with larger errors in Figure 12. It should be noted that the GIS method fails on the dataset of SCAPE re-meshed.

### 3.6. Robustness

We also perform the proposed method on the animal models with different perturbations. We collect a horse dataset containing 8 clean shapes from TOSCA [7] and 45 shapes from SHREC'10 [6], which are the deformable version of TOSCA. The training set is composed of 4 shapes from TOSCA and 24 from SHREC'10 which belongs to the first three levels of 8 classes perturbations, such as isometry, topology, hole, micro hole, local scale, noise, shot noise and sampling. The rest of shapes are used as test set. Since the horse shapes have much vertices, we increase the grid sub-sampling step to 2 for the KPConv. Figure 13 illustrates the comparisons of direct and symmetric correspondence from the sources to clean shape and its deformable ones. The average geodesic distance errors of direct and symmetric correspondences from sources to the clean shape are 1.83 and 2.55, and the errors for the deformable ones are 4.56 and 2.75. The qualitative results and quantitative errors demon-

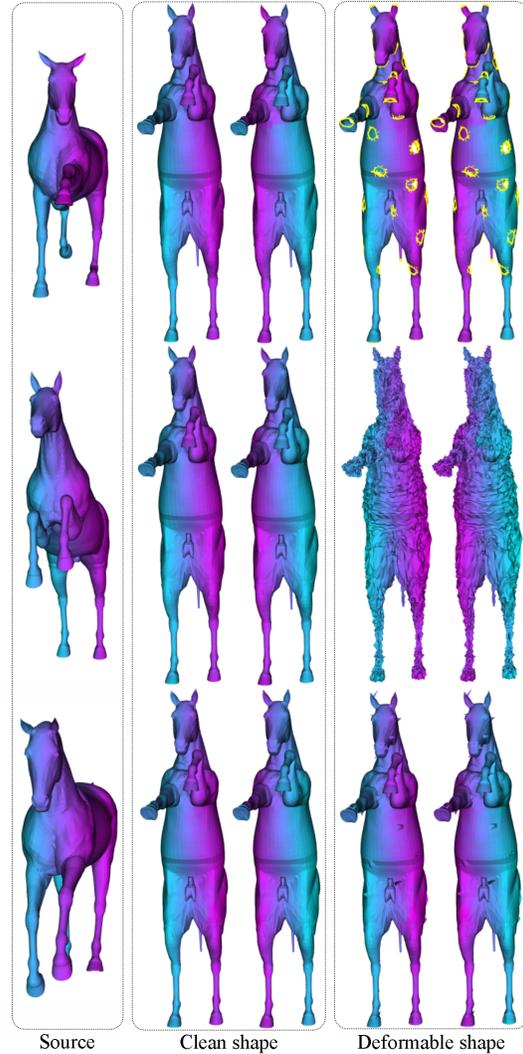


Figure 13. Direct and symmetric correspondences results from sources to clean shape and its deformable ones with holes (top), Gaussian noises (middle) and shot noises (bottom). Corresponding points have the same colors. The boundary edges are shown in yellow color.

strate the proposed method is robust to the above perturbations.

## 4. Conclusion, limitations and future works

In this paper, a novel deep functional map is proposed to solve the multiple solutions problem when matching shapes with left-to-right reflectional intrinsic symmetries. Both the direct and symmetric correspondences among a pair of shapes and reflectional intrinsic symmetry of each shape are computed at the same time. The key novelty is that the second Siamese network learns symmetric descriptors of the first Siamese network. We demonstrate through extensive experiments on correspondences and symmetries detection

that our method obtains better results than previous works.

Our method still has some limitations. Firstly, we only target multiple solutions of matching shapes with left-to-right reflectional intrinsic symmetries, and do not consider rotational and continuous intrinsic symmetries. Secondly, the proposed method is limited by requiring the ground truth maps of correspondences and symmetries of enough training examples. This makes it difficult to apply our method to new shape classes for which ground truth data is not available.

In the future, we would like to investigate learning methods for addressing the multiple solutions of correspondences of shapes with rotational intrinsic symmetries, such as octopus, in spirit of the work [29]. We also want to investigate possible unsupervised learning methods for the multiple solutions problems of matching shapes with intrinsic symmetries.

## 5. Acknowledgments

We sincerely thank the anonymous reviewers for their constructive comments. This work was supported in parts by NSFC (61972267, 62161146005, U21B2023, U2001206, 61976040, 61976041), National Key R&D Program of China (2020YFB1708902), DEGP Key Project (2020SFKC059), and Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ).

## References

- [1] Y. Aflalo, A. Dubrovina, and R. Kimmel. Spectral generalized multi-dimensional scaling. *International Journal of Computer Vision*, 118:380–392, 2016. 2
- [2] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Davis, and J. Rodgers. SCAPE: Shape completion and animation of people. *ACM Transactions on Graphics*, 24(3):408–416, 2005. 7
- [3] M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *International Conference on Computer Vision*, pages 1626–1633, 2011. 3
- [4] S. Biasotti, A. Cerri, A. Bronstein, and M. Bronstein. Recent trends, applications, and perspectives in 3D shape similarity assessment. *Computer Graphics Forum*, 35(6):87–119, 2016. 2
- [5] F. Bogo, J. Romero, M. Loper, and M. J. Black. FAUST: Dataset and evaluation for 3D mesh registration. In *Computer Vision and Pattern Recognition*, pages 3794–3801, 2014. 1, 7
- [6] A. Bronstein, M. Bronstein, U. Castellani, A. Dubrovina, and A. Sharma. SHREC 2010: Robust correspondence benchmark. In *Eurographics Workshop on 3D Object Retrieval*, pages 87–91, 2010. 10
- [7] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer Science & Business Media, 2008. 10
- [8] O. Burghard, A. Dieckmann, and R. Klein. Embedding shapes with Green’s functions for global shape matching. *Computers & Graphics*, 68:1–10, 2017. 2
- [9] M. Chen, C. Wang, and H. Qin. Jointly learning shape descriptors and their correspondence via deep triplet CNNs. *Computer Aided Geometric Design*, 62:192–205, 2018. 1
- [10] N. Donati, A. Sharma, and M. Ovsjanikov. Deep geometric maps: Robust feature learning for shape correspondence. In *Computer Vision and Pattern Recognition*, pages 8589–8598, 2020. 1, 2, 3, 4, 5, 6, 8
- [11] D. Eynard, E. Rodolà, K. Glashoff, and M. M. Bronstein. Coupled functional maps. In *International Conference on 3D Vision*, pages 399–407, 2016. 2
- [12] D. Ezuz and M. Ben-Chen. Deblurring and denoising of maps between shapes. *Computer Graphics Forum*, 36(5):165–174, 2017. 3
- [13] L. Gao, L. Zhang, H. Meng, Y. Ren, Y. Lai, and L. Kobbelt. PRS-Net: Planar reflective symmetry detection net for 3D models. *IEEE Transactions on Visualization & Computer Graphics*, 27(6):3007–3018, 2020. 3
- [14] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry. 3D-CODED: 3D correspondences by deep deformation. In *European Conference on Computer Vision*, pages 257–275, 2018. 1, 8, 9
- [15] O. Halimi, O. Litany, E. Rodolà, A. M. Bronstein, and R. Kimmel. Unsupervised learning of dense shape correspondence. In *Computer Vision and Pattern Recognition*, pages 4370–4379, 2019. 1, 2, 3, 4, 5, 8
- [16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, pages 770–778, 2016. 4
- [17] Q. Huang, F. Wang, and L. Guibas. Functional map networks for analyzing and exploring large shape collections. *ACM Transactions on Graphics*, 33(4):36:1–36:11, 2014. 2
- [18] V. G. Kim, Y. Lipman, and T. Funkhouser. Blended intrinsic maps. *ACM Transactions on Graphics*, 30(4):79:1–79:12, 2011. 8
- [19] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, K. Glashoff, and R. Kimmel. Coupled quasi-harmonic bases. *Computer Graphics Forum*, 32(2):439–448, 2013. 2
- [20] O. Litany, T. Remez, E. Rodolà, A. Bronstein, and M. Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. In *International Conference on Computer Vision*, pages 5660–5668, 2017. 1, 2, 3, 4, 5, 8
- [21] X. Liu, S. Li, R. Liu, J. Wang, H. Wang, and J. Cao. Properly constrained orthonormal functional maps for intrinsic symmetries. *Computers & Graphics*, 46:198–208, 2015. 3
- [22] S. Melzi, J. Ren, E. Rodolà, A. Sharma, P. Wonka, and M. Ovsjanikov. ZoomOut: Spectral upsampling for efficient shape correspondence. *ACM Transactions on Graphics*, 38(6):155:1–155:14, 2019. 3, 6, 8, 10
- [23] N. J. Mitra, M. Pauly, M. Wand, and D. Ceylan. Symmetry in 3D geometry: Extraction and applications. *Computer Graphics Forum*, 32(6):1–23, 2013. 3
- [24] R. Nagar and S. Raman. Fast and accurate intrinsic symmetry detection. In *European Conference on Computer Vision*, pages 417–434, 2018. 3

- [25] D. Nogneng and M. Ovsjanikov. Informative descriptor preservation via commutativity for shape matching. *Computer Graphics Forum*, 36(2):259–267, 2017. 2
- [26] M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, and L. Guibas. Functional maps: A flexible representation of maps between shapes. *ACM Transactions on Graphics*, 31(4):30:1–30:11, 2012. 1, 2, 3, 8
- [27] M. Ovsjanikov, E. Corman, M. Bronstein, E. Rodolà, and A. Bronstein. Computing and processing correspondences with functional maps. In *ACM SIGGRAPH Courses*, pages 1–62, 2017. 2, 3
- [28] M. Ovsjanikov, Q. Mérigot, V. Patraucean, and L. Guibas. Shape matching via quotient spaces. *Computer Graphics Forum*, 32(5):1–11, 2013. 3
- [29] J. Ren, S. Melzi, M. Ovsjanikov, and P. Wonka. Maptree: Recovering multiple solutions in the space of maps. *ACM Transactions on Graphics*, 39(6):264:1–264:17, 2020. 3, 9, 10, 11
- [30] J. Ren, A. Poulencard, P. Wonka, and M. Ovsjanikov. Continuous and orientation-preserving correspondences via functional maps. *ACM Transactions on Graphics*, 37(6):248:1–248:16, 2018. 3, 7, 8, 9, 10
- [31] E. Rodolà, L. Cosmo, M. M. Bronstein, A. Torsello, and D. Cremers. Partial functional correspondence. *Computer Graphics Forum*, 36(1):222–236, 2017. 2
- [32] E. Rodolà, M. Moeller, and D. Cremers. Point-wise map recovery and refinement from functional correspondence. In *Vision, Modeling & Visualization*, pages 25–32, 2015. 3
- [33] J.-M. Roufousse, A. Sharma, and M. Ovsjanikov. Unsupervised deep learning for structured shape matching. In *International Conference on Computer Vision*, pages 1617–1627, 2019. 1, 2, 3, 4, 5, 8
- [34] Y. Sahillioalu. Recent advances in shape correspondence. *The Visual Computer*, 36:1705–1721, 2020. 2
- [35] A. Sharma and M. Ovsjanikov. Weakly supervised deep functional map for shape matching. In *Advances Neural Information Processing Systems*, pages 19264–19275, 2020. 2, 4
- [36] N. Sharp, S. Attaiki, K. Crane, and M. Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces, 2020. 1
- [37] R. W. Sumner and J. Popovic. Deformation transfer for triangle meshes. *ACM Transactions on Graphics*, 23(3):399–405, 2004. 1
- [38] J. Sun, M. Ovsjanikov, and L. J. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *Symposium on Geometry Processing*, 28(5):1383–1392, 2009. 3
- [39] Y. Sun, Z. Liang, X. Huang, and Q. Huang. Joint map and symmetry synchronization. In *European Conference on Computer Vision*, pages 257–275, 2018. 3
- [40] Z. Sun, Y. He, A. Gritsenko, A. Lendasse, and S. Baek. Embedded spectral descriptors: Learning the point-wise correspondence metric via siamese neural networks. *Journal of Computational Design and Engineering*, 7:18–29, 2020. 1
- [41] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *International Conference on Computer Vision*, pages 6411–6420, 2019. 4, 5
- [42] F. Tombari, S. Salti, and L. D. Stefano. Unique signatures of histograms for local surface description. In *European Conference on Computer Vision*, pages 356–369, 2010. 4
- [43] O. van Kaick, H. Zhang, G. Hamarneh, and D. Cohen-Or. A survey on shape correspondence. *Computer Graphics Forum*, 30(6):1681–1707, 2011. 1, 2
- [44] M. Vestner, R. Litman, E. Rodolà, A. Bronstein, and D. Cremers. Product manifold filter: Non-rigid shape correspondence via kernel density estimation in the product space. In *Computer Vision and Pattern Recognition*, pages 6681–6690, 2017. 8
- [45] H. Wang and H. Huang. Group representation of global intrinsic symmetries. *Computer Graphics Forum*, 36(7):51–61, 2017. 3, 10
- [46] L. Wei, Q. Huang, D. Ceylan, E. Vouga, and H. Li. Dense human body correspondences using convolutional networks. In *Computer Vision and Pattern Recognition*, pages 1544–1553, 2016. 1
- [47] Y. Xiao, Y. Lai, F. Zhang, C. Li, and L. Gao. A survey on deep geometry learning: From a representation perspective. *Computational Visual Media*, 6(2):113–133, 2020. 2
- [48] S. Zuffi, A. Kanazawa, D. W. Jacobs, and M. J. Black. 3D menagerie: Modeling the 3D shape and pose of animals. *Computer Vision and Pattern Recognition*, pages 6365–6373, 2017. 7, 8