Integrating Efficient Optimal Transport and Functional Maps For Unsupervised Shape Correspondence Learning

Anonymous CVPR submission

Paper ID 16800

Abstract

In the realm of computer vision and graphics, accurately establishing correspondences between geometric 3D shapes is pivotal for applications like object tracking, registration, texture transfer, and statistical shape analysis. Moving beyond traditional hand-crafted and data-driven feature learning methods, we incorporate spectral methods with deep learning, focusing on functional maps (FMs) and optimal transport (OT). Traditional OT-based approaches, often reliant on entropy regularization OT in learning-based framework, face computational challenges due to their quadratic cost. Our key contribution is to employ the sliced Wasserstein distance (SWD) for OT, which is a valid fast optimal transport metric in an unsupervised shape matching framework. This unsupervised framework integrates functional map regularizers with a novel OT-based loss derived from SWD, enhancing feature alignment between shapes treated as discrete probability measures. We also introduce an adaptive refinement process utilizing entropy regularized OT, further refining feature alignments for accurate point-to-point correspondences. Our method demonstrates superior performance in non-rigid shape matching, including near-isometric and non-isometric scenarios, and excels in downstream tasks like segmentation transfer. The empirical results on diverse datasets highlight our framework's effectiveness and generalization capabilities, setting new standards in non-rigid shape matching with efficient OT metrics and an adaptive refinement module.

1. Introduction

Establishing precise correspondences between geometric 3D shapes is a core challenge in various domains of computer vision and graphics, including but not limited to, object tracking, registration, texture transfer, and statistical shape analysis [7, 14, 52, 61]. To facilitate the mapping between non-rigid shapes, early approaches [6, 9, 48] concentrated on the development of hand-crafted features, leveraging geometric invariance as a key principle. In the latter approaches [4, 10, 16, 27], there has been a shift towards the utilization of data-driven methods for feature learning, which has resulted in marked enhancements in terms of accuracy, efficiency, and robustness.

Recently, an increasing body of work has exploited the use of spectral methods [5, 18, 21, 32, 46], especially the functional map (FM) representation [39]. Specifically, the FM methods succinctly encode correspondences through compact matrices, utilizing a truncated spectral basis. With recent developments in deep learning, deep FM (DFM) is quickly employed in numerous settings [11, 12, 27, 54] by incorporating feature learning as geometric descriptors for FM frameworks. Most DFM works focus on learning features that optimize FM priors to express desirable map priors, e.g. area preservation, isometry, and bijectivity, which achieves remarkable results even without supervision [10, 12, 20, 21, 46]. On the other hand, less attention is paid to the problem of explicitly aligning features outputted from the feature extractor network, due to the lack of smoothness and consistency of linear assignment problems.

In this work, we focus on jointly learning features via the functional map, and explicit features, i.e. directly from the feature extractor to establish correct correspondence. Nonetheless, learning to map explicit features is not easy since the geometric objects might potentially undergo arbitrary deformations. Therefore, we propose to employ optimal transport (OT), which is a well-known approach for linear assignment problems, to cast the feature alignment from 3D shapes as a probability measures matching problem.

The Wasserstein distance [41, 59] is widely acknowledged as an effective OT metric for comparing two probability measures, particularly when their supports are disjoint. However, it comes with the drawback of high computational complexity. Specifically, for discrete probability measures with at most m supports, the time and memory complexities are $\mathcal{O}(m^3 \log m)$ and $\mathcal{O}(m^2)$, respectively. This computational burden is exacerbated in 3D shape applications where each shape, represented as mesh, is treated as a distinct probability measure. To ameliorate the computational demands, entropic regularization coupled with the Sinkhorn algorithm [13] can yield an ϵ-approximation of the Wasserstein distance with a time complexity of $\mathcal{O}(m^2/\epsilon^2)$ [2, 29–31]. Nonetheless, this method does not alleviate the $\mathcal{O}(m^2)$ memory complexity due to the necessity of storing the cost matrix. Additionally, the entropic regularization fails to produce a valid metric between probability measures as it does not satisfy the triangle inequality. An alternative, more efficient method is the sliced Wasserstein distance (SWD) [8], which calculates the expectation of the Wasserstein distance between random one-dimensional push-forward measures derived from the original measures. SWD offers a time complexity of $\mathcal{O}(m \log m)$ and a linear memory complexity of $\mathcal{O}(m)$.

Motivated by the above discussion, we introduce a novel differentiable unsupervised OT-based loss derived from efficient sliced Wasserstein distance, which accounts for associating two extracted extrinsic features to align two meshes combined with functional map regularizers. Our proposed approach leverages a valid efficient OT metric to obtain highly discriminative local feature matching. Additionally, the integration of functional map regularizers promotes smoothness in the mapping process, allowing our method to achieve both precise and smooth correspondence.

Furthermore, we introduce an adaptive refinement process tailored for each pair of shapes, utilizing entropy regularized OT to enhance matching performance. The differentiable nature of entropic regularization in OT enables our refinement strategy to leverage the Sinkhorn algorithm. This approach yields a soft point-wise map, which is instrumental in calculating FM regularizers. These regularizers are then used to iteratively update features, thereby facilitating the retrieval of precise point-to-point correspondences.

Finally, we demonstrate our proposed approach on a diverse and extensive selection of datasets. Our contributions are as follows:

- We propose an unsupervised learning framework that employs efficient optimal transport to jointly learn with functional map in shape matching paradigm. Subsequently, we derive two novel unsupervised loss functions based on sliced Wasserstein distance, which is a valid fast optimal transport metric, to effectively align mesh features by interpreting them as probability measures, potentially offering a promising avenue for advancements in shape matching through efficient optimal transport.
- To enhance the quality of point mapping, we propose an adaptive refinement module that iteratively refines the optimal transport similarity matrix estimated via entropy regularization optimal transport.
- We outperform previous state-of-the-art works in various settings of non-rigid shape matching including near-isometric and non-isometric shape matching. Additionally, when applied to a downstream task such as segmentation transfer, our approach continues to outperform contemporary state-of-the-art methods in non-rigid shape matching. This success not only demonstrates the efficacy of our method in specific applications but also underlines its strong generalization capabilities across various use cases in shape matching.

2. Related work

Shape matching has been extensively explored for decades. For a comprehensive examination of this topic, we encourage readers to consult the detailed analyses presented in surveys [47, 56]. In this section, we focus specifically on the literature subset that directly relates to our research objectives.

2.1. Deep functional maps for shape correspondence.

Our methodology is founded on the functional map representation, initially introduced in [39] and substantially developed through subsequent research, e.g. [40]. The central concept of functional maps revolves around expressing shape correspondences as transformations between their respective spectral embeddings. This is efficiently achieved by utilizing compact matrices formulated from reduced eigenbases. The functional maps approach has seen considerable enhancements in terms of accuracy, efficiency, and robustness, as evidenced by a variety of recent contributions [22, 25, 45]. In contrast to axiomatic approaches that rely on manually engineered features [53], deep functional map methods aim to autonomously learn features from training data. The pioneering work in this domain was FMNet [32], which introduced a method to learn non-linear transformations of SHOT descriptors [48]. Subsequent developments [21, 46] facilitated the unsupervised training of FMNet by incorporating isometry losses in both spatial and spectral domains. This unsupervised approach has been further enhanced with the advent of robust mesh feature extractors [49], leading to the development of new frameworks [10, 12, 16, 27] that learn directly from geometric data, achieving top-tier performance.

2.2. Optimal transport for shape correspondence

Optimal transport has emerged as a powerful tool in the field of shape correspondence, offering innovative approaches to match and analyze complex shapes in computer graphics and computer vision. Starting with the axiomatic shape matching approach, [51] proposed an algorithm for probabilistic correspondence that optimizes an entropy-regularized Gromov-Wasserstein (GW) objective [36] to find the correspondence between two given shapes. The proposed framework is inefficient since solving entropy-regularized GW objective is relatively expensive and it does not perform well on non-isometric shape matching. To address the computational overhead of solving OT cost, [50] brought robust OT to the forefront, significantly enhancing the accuracy and efficiency of point cloud registration, but the framework is designed for point cloud that avoids the connectivity of the shape mesh. Perhaps the most relevant work to ours is Deep Shells [18], which is an improvement of [17]. Deep Shells demonstrated how OT can be seamlessly integrated into deep neural networks, offering a new perspective in shape matching with improved adaptability and precision. However, computing OT cost via Sinkhorn algorithm in Deel Shells [18] can be expensive since it has to store the cost matrix with quadratic memory cost and quadratic time complexity. In light of this, we propose to employ an efficient OT in learning shape correspondence. To be specific, we employ sliced Wasserstein distance, which calculates the expectation the Wasserstein distance between two random one-dimensional push-forward measures derived from original measures. Recently, sliced Wasserstein distance has been successfully applied in point cloud [38] and shape [26] deformation. However, to the best of our knowledge, we are the first to employ sliced Wasserstein distance on shape correspondence framework.

3. Background

In this section, we briefly recap functional map representation [39]. After that, we review the definition of Wasserstein distance and its closed-formed solution sliced Wasserstein distance.

3.1. (Deep) Functional Maps

Given a pair of smooth shapes $\mathcal{X}$ and $\mathcal{Y}$, which are discretized as triangular meshes with n_x and n_y vertices, respectively. The functional map method aims to obtain a dense correspondence between the two shapes by compactly representing the correspondence matrix as a smaller matrix. Specifically, the leading k eigenfunctions of the Laplace-Beltrami operator are computed on both shapes $\mathcal{X}$, $\mathcal{Y}$ and are presented as $\Phi_x \in \mathbb{R}^{n_x \times k}$ and $\Phi_y \in \mathbb{R}^{n_y \times k}$, respectively. The geometric features of the shape are either precomputed [48] or extracted from a neural network [49], represented as $\mathcal{F}_x \in \mathbb{R}^{n_x \times d}$ and $\mathcal{F}_y \in \mathbb{R}^{n_y \times d}$, where d is the feature dimension. The extracted features are then projected into the eigenbasis to get the corresponding coefficients $\mathbf{A} = \Phi_x^\dagger \mathcal{F}_x \in \mathbb{R}^{k \times d}$ and $\mathbf{B} = \Phi_y^\dagger \mathcal{F}_y \in \mathbb{R}^{k \times d}$, where $\dagger$ denotes the Moore-Penrose pseudo-inverse. After that, the bidirectional optimal functional map $\mathbf{C}^*_{xy}, \mathbf{C}^*_{yx} \in \mathbb{R}^{k \times k}$ is obtained by solving the linear system:

$$\mathbf{C}^*_{xy} = \arg\min_{\mathbf{C}} E_{data}(\mathbf{C}) + E_{reg}(\mathbf{C}), \tag{1}$$

where $E_{data}(\mathbf{C}) = \|\mathbf{CA} - \mathbf{B}\|^2$ promotes the descriptor preservation, whereas the E_{reg} is a regularization term imposing structural properties of $\mathbf{C}$ [39]. Finally, the dense correspondence can be reconstructed from estimated $\mathbf{C}^*$ by conducting nearest neighbor search between the rows of $\Phi_x \mathbf{C}_{yx}$ and that of Φ_y, with possible post-processing [19, 35, 42].

3.2. Efficient Optimal Transport

Wasserstein distance. For $p \geq 1$, given two probability measures $\mu \in \mathcal{P}_p(\mathbb{R}^d)$ and $\nu \in \mathcal{P}_p(\mathbb{R}^d)$, the Wasserstein distance [57] between μ and ν is :

$$\mathrm{W}_p^p(\mu,\nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x-y\|_p^p d\pi(x,y), \tag{2}$$

where $\Pi(\mu,\nu)$ are the set of all couplings between μ and ν i.e., joint probability measures that have marginals as μ and ν respectively. The Wasserstein distance is the optimal transportation cost between μ and ν since it is computed with the optimal coupling. As mentioned in the introduction section, the downside of Wasserstein distance is a high computational complexity in the discrete case i.e., $\mathcal{O}(m^3 \log m)$ in time and $\mathcal{O}(m^2)$ in space for m is the number of supports. To reduce the time complexity, entropic regularized optimal transport [13] is introduced.

Sinkhorn divergence. For $p \geq 1$, given two probability measures $\mu \in \mathcal{P}_p(\mathbb{R}^d)$ and $\nu \in \mathcal{P}_p(\mathbb{R}^d)$, the Sinkhorn-p divergence [13] between μ and ν is :

$$\mathrm{S}_{\epsilon,p}^p(\mu,\nu) = \inf_{\pi \in \Pi_\epsilon(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} c d\pi(x,y) + \epsilon H(\pi), \tag{3}$$

where $\Pi_\epsilon(\mu,\nu) = \{\pi \in \Pi(\mu,\nu) | \mathrm{KL}(\pi, \mu \otimes \nu) \leq \epsilon\}$ with KL denotes the Kullback Leibler divergence. The cost $c : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$ is defined as $c_p(x,y) = \|x-y\|_p^p$ on $\mathbb{R}^d \times \mathbb{R}^d$. The entropy term $H(\pi)$ allows us to solve for the correspondence π via Sinkhorn-Knopp algorithm with $\mathcal{O}(m^2)$ in time complexity.

Sliced Wasserstein distance. The sliced Wasserstein (SW) distance [8] between two probability measures $\mu \in \mathcal{P}_p(\mathbb{R}^d)$ and $\nu \in \mathcal{P}_p(\mathbb{R}^d)$ is given by:

$$\mathrm{SW}_p^p(\mu,\nu) = \mathbb{E}_{\theta \sim \mathcal{U}(\mathbb{S}^{d-1})}[\mathrm{W}_p^p(\theta\sharp\mu, \theta\sharp\nu)], \tag{4}$$

where $\theta\sharp\nu$ denotes the push-forward measure of μ via function $f(x) = \theta^\top x$, and the one-dimensional Wasserstein distance appears in a closed form which is $\mathrm{W}_p^p(\theta\sharp\mu, \theta\sharp\nu) = \int_0^1 |F^{-1}_{\theta\sharp\mu}(z) - F^{-1}_{\theta\sharp\nu}(z)|^p dz$. Here, $F_{\theta\sharp\mu}$ and $F_{\theta\sharp\nu}$ are the cumulative distribution function (CDF) of $\theta\sharp\mu$ and $\theta\sharp\nu$ respectively. When μ and ν have at most n supports, the computation of the SW is only $\mathcal{O}(n \log n)$ in time and $\mathcal{O}(n)$ in space. The SW often is computed by using L Monte Carlo samples $\theta_1, \ldots, \theta_L$ from the unit sphere:

$$\widehat{\mathrm{SW}}_p^p(\mu,\nu;L) = \frac{1}{L}\sum_{l=1}^{L} \mathrm{W}_p^p(\theta_l\sharp\mu, \theta_l\sharp\nu). \tag{5}$$

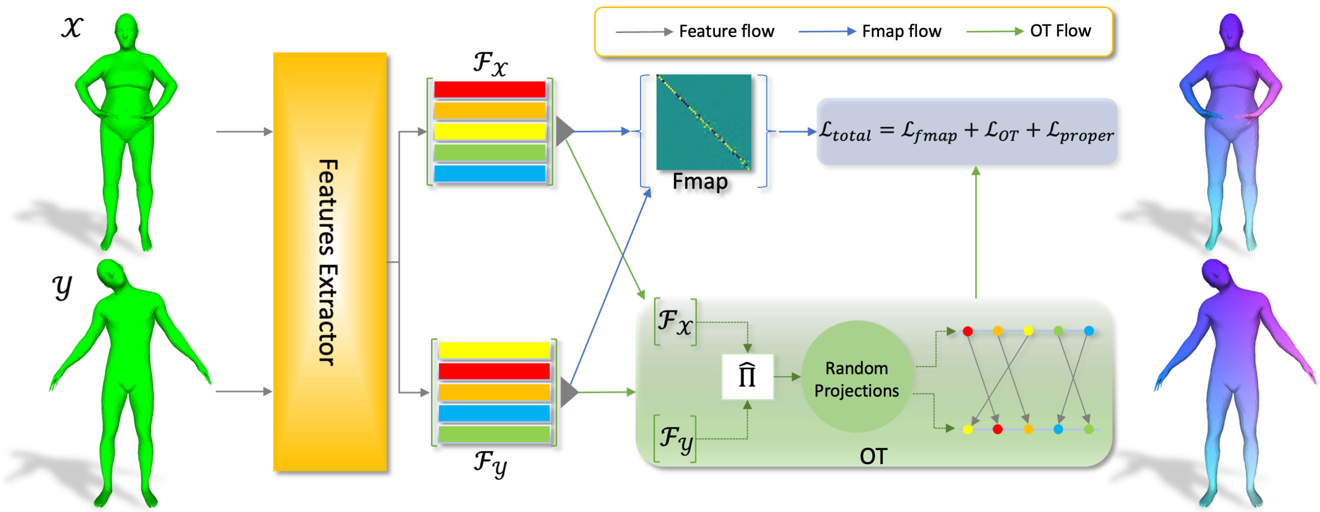

Figure 1. **Overview of unsupervised shape matching via efficient OT.** Our framework takes as input a pair of shapes $\mathcal{X}$ and $\mathcal{Y}$ and outputs point-to-point correspondence. Firstly, the features extractor tasks the pair input and extracts vertex-wise features $\mathcal{F}_x$ and $\mathcal{F}_y$. Subsequently, the differentiable functional map solver is used to compute functional map given pre-computed eigenfunctions and the extracted features. In parallel, our framework estimates a soft feature similarity matrix, derived from the same extracted features. After that, an OT cost is computed given soft feature similarity and extracted feature $\mathcal{F}_x$ and $\mathcal{F}_y$. Finally, a proper loss is optimized together with regularized functional map loss and OT loss.

Energy-based Sliced Wasserstein distance. Energy-based sliced Wasserstein (EBSW) is a more discriminative variant of the SW proposed in [37]. The definition of the EBSW is given as:

$$\text{EBSW}_p^p(\mu,\nu;f) = \mathbb{E}_{\theta\sim\sigma_{\mu,\nu}(\theta;f,p)}\left[\text{W}_p^p(\theta\sharp\mu,\theta\sharp\nu)\right], \quad (6)$$

where f is the energy function e.g., $f(x) = e^x$, and $\sigma_{\mu,\nu}(\theta;f,p) \propto f(\text{W}_p^p(\theta\sharp\mu,\theta\sharp\nu)) \in \mathcal{P}(\mathbb{S}^{d-1})$ is the energy-based slicing distribution. The EBSW can be computed based on importance sampling with L samples from proposal distribution $\sigma_0(\theta)$, e.g., $\mathcal{U}(\mathbb{S}^{d-1})$. For $\theta_1,\ldots,\theta_L \overset{i.i.d}{\sim} \sigma_0(\theta)$, we have:

$$\widehat{\text{IS-EBSW}}_p^p(\mu,\nu;f,L) = \sum_{l=1}^{L}\text{W}_p^p(\theta_l\sharp\mu,\theta_l\sharp\nu)\hat{w}_{\mu,\nu,\sigma_0,f,p}(\theta_l), \quad (7)$$

for $w_{\mu,\nu,\sigma_0,f,p}(\theta) = \frac{f(\text{W}_p^p(\theta\sharp\mu,\theta\sharp\nu))}{\sigma_0(\theta)}$ is the importance weighted function and $\hat{w}_{\mu,\nu,\sigma_0,f,p}(\theta_l) = \frac{w_{\mu,\nu,\sigma_0,f,p}(\theta_l)}{\sum_{l'=1}^{L} w_{\mu,\nu,\sigma_0,f,p}(\theta_{l'})}$ is the normalized importance weights.

4. Learning Shape Correspondence with Efficient Optimal Transport

In this section, we provide in-depth details of our proposed non-rigid shape matching framework. The whole framework is described in Fig. 1. Our pipeline starts by extracting features from the feature extractor as described in Sec. 4.1. Then we describe functional map in Sec. 4.2. Thirdly, we illustrate how efficient OT in Sec. 4.3 is applied to our framework and propose two novel loss functions for learning precise shape mapping. Thirdly, we summarize our unsupervised losses in Sec. 4.4. Finally, we propose an adaptive refinement process in Sec. 4.5.

4.1. Feature extractor

Our architecture is designed in the form of a Siamese network. Specifically, we utilize the same feature extractor with shared learning parameters to extract features from a pair of input shapes. We employ DiffusionNet [49] as our feature extractor since DiffusionNet is agnostic to discretization and resolution of the meshes, thereby ensuring robust shape correspondence. Consequently, from the pair of inputs, we extract two sets of features, denoted by $\mathcal{F}_x \in \mathbb{R}^{n_x\times d}$ and $\mathcal{F}_y \in \mathbb{R}^{n_y\times d}$ via DiffusionNet.

4.2. Functional map module

As discussed in 3.1, we aim to employ deep functional map as a proxy to learn an intrinsic feature shape matching. Specifically, we employ regularized functional map [43], to compute optimal functional map $\mathbf{C}^*$ as mentioned in Sec. 3.1. During training, the network aims to minimize the structural regularization of functional map:

$$\mathcal{L}_{fmap} = \alpha_1\mathcal{L}_{bij} + \alpha_2\mathcal{L}_{othor}, \quad (8)$$

where $\mathcal{L}_{bij} = \|C_{xy}C_{yx} - I\|^2 + \|C_{yx}C_{xy} - I\|^2$ promotes identity mapping and $\mathcal{L}_{othor} = \|C_{xy}^T C_{yx} - I\|^2 + \|C_{yx}^T C_{xy} - I\|^2$ imposes locally area-preserving [43].

4.3. Feature extrinsic alignment via efficient optimal transport

We aim to integrate efficient OT into deep functional map to promote precise mesh feature alignment. Thanks to the fast computation and the closed-form solution of sliced Wasserstein (SW) distance, we derive a novel loss function based on SW distance.

Soft feature similarity. Firstly, from a pair of features $\mathcal{F}_x, \mathcal{F}_y$ extracted from shapes $\mathcal{X}, \mathcal{Y}$, respectively, we estimate a *soft feature similarity matrix* $\hat{\Pi}_{xy} \in \mathbb{R}^{n_x \times n_y}$ such that:

$$\hat{\Pi}_{xy}^{i,j} = \frac{\exp((\mathcal{F}_x^i \cdot \mathcal{F}_y^j)/\tau)}{\sum_{k=1}^{n_y} \exp((\mathcal{F}_x^i \cdot \mathcal{F}_y^k)/\tau))}, \tag{9}$$

where τ is scaling factor, and $\mathcal{F}_x^i, \mathcal{F}_y^j \in \mathbb{R}^d$ represent d-dimensional features of point i^{th} in shape $\mathcal{X}$ and j^{th} in shape $\mathcal{Y}$, respectively. Similarly, the $\hat{\Pi}_{yx}$ is constructed in the same fashion as in Eq. 9.

Feature alignment via OT. Finding precise point-to-point mapping based on feature similarity requires solving linear assignment problem in $\mathbb{R}^d$, which is expensive to integrate into a learning-based framework. Therefore, in this work, we relax the constraints to cast the feature-matching problem as a probability distribution matching problem. In other words, we represent the extracted features $\mathcal{F}_x, \mathcal{F}_y$ as probability distributions defined over $\mathbb{R}^d$. After that, we attempt to learn mappings that minimize the "distance" between the two distributions, i.e. probability measures. The OT cost [58] is a naturally fitted discrepancy between probability measures, thereby being employed in our framework.

SW distance as an efficient OT. Thanks to the fast computation and its closed-form solution of SW distance, we derive a novel loss function that jointly learns the mapping and minimizes the discrepancy between two feature probability measures as follows:

$$\mathcal{L}_{biSW} = (\mathbb{E}_{\theta \sim \mathcal{U}(\mathbb{S}^{d-1})}[\mathrm{W}_p^p(\theta \sharp \mathcal{F}_x, \theta \sharp \hat{\mathcal{F}}_y) + \mathrm{W}_p^p(\theta \sharp \mathcal{F}_y, \theta \sharp \hat{\mathcal{F}}_x)])^{\frac{1}{p}}, \tag{10}$$

where $\hat{\mathcal{F}}_x = \hat{\Pi}_{yx}\mathcal{F}_x$ and $\hat{\mathcal{F}}_y = \hat{\Pi}_{xy}\mathcal{F}_y$. The loss $\mathcal{L}_{biSW}$ minimizes the discrepancy between the feature probability measures in one shape and the softly permuted feature sets of its counterpart in a bidirectional manner. The loss converges toward zero when the soft feature similarity $\hat{\Pi}$ approaches a (partial) permutation matrix, indicating that the point-wise corresponding features are closely aligned. Moreover, the loss encourages the cycle consistency of the mapping. It is worth noting that our loss diverges from contrastive losses explored in prior works [11, 27, 60]. Where the contrastive loss only considers whether individual point correspondences are correct or not, our proposed loss introduces a more general and flexible matching by conceptualizing the point features as probability measures and employing OT cost as a metric of evaluation.

Bidirectional EBSW. It is worth noting that the proposed loss $\mathcal{L}_{biSW}$ in Eq. 10 employs the projecting directions sampled from uniform distribution over unit-hypersphere as the shared slicing distributions. Despite being easy to sample, the uniform distribution is not able to differentiate between informative and non-informative projecting features. Therefore, inspired by [37], we propose a bidirectional energy-based SW loss defined in the importance sampling form as:

$$\mathcal{L}_{biEBSW} = \left(\frac{\mathbb{E}_{\theta \sim \sigma_0(\theta)}[(\mathrm{W}_{\theta,\mathcal{X}} + \mathrm{W}_{\theta,\mathcal{Y}})w(\theta)]}{\mathbb{E}_{\theta \sim \sigma_0(\theta)}[w(\theta)]}\right)^{\frac{1}{p}}, \tag{11}$$

where we denote $\mathrm{W}_{\theta,\mathcal{X}} := \mathrm{W}_p^p(\theta \sharp \mathcal{F}_x, \theta \sharp \hat{\mathcal{F}}_y), \mathrm{W}_{\theta,\mathcal{Y}} := \mathrm{W}_p^p(\theta \sharp \mathcal{F}_y, \theta \sharp \hat{\mathcal{F}}_x)$, and $w(\theta) := \frac{\exp(\mathrm{W}_{\theta,\mathcal{X}} + \mathrm{W}_{\theta,\mathcal{Y}})}{\sigma_0(\theta)}$. The loss $\mathcal{L}_{biEBSW}$ shares the same properties for shape correspondence as the vanilla SW loss in Eq. 10. However, it imposes a more expressive mechanism for selecting projection directions in the computation of the SW distance. Moreover, the vanilla SW loss can be seen as a summation of two SW distances since the slicing distribution is fixed as uniform. In contrast, the bidirectional EBSW loss has the slicing distribution shared and affected by both one-dimensional Wasserstein distances. Hence, the bidirectional EBSW is considerably different from the original EBSW in [37].

We provide detailed computation and discussion of $\mathcal{L}_{biSW}$ and $\mathcal{L}_{biEBSW}$ at Sup. 9.

4.4. Loss functions

Proper functional maps. We employ the notion of proper functional map introduced by [44]: *The functional map C_{xy} is deemed "proper" if there exists a (partial) permutation matrix Π_{yx} so that $C_{xy} = \Phi_y^\dagger \Pi_{yx} \Phi_x$.* Drawing on this concept, we introduce a loss term that not only promotes the "properness" of the functional map but also concurrently regularizes the (OT) cost, namely:

$$\mathcal{L}_{proper} = \|C_{xy} - \Phi_y^\dagger \hat{\Pi}_{yx} \Phi_x\|^2 \tag{12}$$

It is worth noting that while our $\mathcal{L}_{proper}$ might bear resemblance to the coupling loss in [12], the proposed loss diverges by using soft feature similarity $\hat{\Pi}_{yx}$ jointly optimized with the feature extrinsic alignment through OT as discussed in Sec. 4.3. Therefore, it serves as a strong regularization for imposing structural smoothness of functional map and promoting precise mapping via OT.

Total loss. Our framework is trained end-to-end without annotation by minimizing the following unsupervised losses:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{fmap} + \lambda_2 \mathcal{L}_{OT} + \lambda_3 \mathcal{L}_{proper}, \tag{13}$$

where λ_i is the weight for each loss, and $\mathcal{L}_{OT}$ could be either $\mathcal{L}_{biSW}$ or $\mathcal{L}_{biEBSW}$.

4.5. Adaptive refinement via entropic optimal transport

Adaptive refinement. To provide a more precise correspondence, we propose an adaptive refinement module designed to incrementally improve the final match for each individual shape pairing. Specifically, we estimate the pseudo soft correspondence $\tilde{\Pi}$ via entropic regularized optimal transport [13] as mentioned in Eq. 3 is defined as:

$$\tilde{\Pi}_{xy} = \mathcal{Q}^{\mathcal{X}}(\mathcal{Q}^{\mathcal{Y}} \cdots (\mathcal{Q}^{\mathcal{X}}(p_\epsilon))), \tag{14}$$

where $\mathcal{Q}(\cdot)$ is the projection operator of a given probability density $p : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ defined as: $p_\epsilon(x, y) \propto \exp(-\frac{1}{\epsilon} c_2(x, y))$. Thanks to the differentiable property of the Sinkhorn algorithm, we can refine each individual pair by minimizing the $\mathcal{L}_{total}$ to update the features accordingly. In contrast to the axiomatic method [35] that often requires alternately updating the functional map and pointwise map, our method offers a differentiable process that facilitates simultaneous updates. Furthermore, it is noteworthy that our approach is orthogonal to [18] since we only employ entropic OT for refinement once during the inference, thereby reducing the computation and memory cost of the Sinkhorn algorithm. We provide detailed algorithms of adaptive refinement at Sup. 9.

Inference. During inference, our final mapping is obtained by nearest neighbor search on features extracted from the feature extractor module.

5. Experimental results

Datasets. We conduct a series of experiments across diverse shape-matching datasets and their application on a downstream task. Specifically, we perform experiment on human shape matching with near-isometric dataset such as FAUST [7] and SCAPE [3] as well as non-isometric dataset SHREC'19 [34]. Furthermore, our study extends to two non-isometric animal datasets: SMAL [62] and the more recent DeformingThings4D [28, 33]. Finally, we conclude our experiments by performing segmentation transfer on 3D semantic segmentation dataset introduced in [1].

Baselines. We conduct extensive comparisons with a wide range of non-rigid shape matching methods: (1) Axiomatic methods including ZoomOut [35], BCICP [42], Smooth Shells [17]; (2) Supervised methods including FMNet [32], GeomFMaps [15], TransMatch [55]; (3) Unsupervised methods including SURFMNet [46], Deep Shells [18], AFMap [27], SSLMSM [11], UDMSM [10], ULRSSM [12]. While there are numerous non-rigid shape-matching methods in the literature, we decided to choose the most recent and relevant to our works for comparison.

Metrics. Regarding shape matching metric, similar to all of our competing methods, we employ mean geodesic errors ($\times$100) [24]. For segmentation transfer, we use semantic segmentation mIOU as in [23].

5.1. Near-isometric Shape Matching

Datasets. We employ a more challenging remeshed version of FAUST [7] and SCAPE [3], as proposed in [15, 42]. The remeshed FAUST dataset includes 100 shapes, representing 10 individuals in 10 different poses, with the evaluation focusing on the final 20 shapes. Similarly, the remeshed SCAPE dataset comprises 71 poses of a single individual, where again, the last 20 shapes are used for evaluation purposes. Additionally, the SHREC'19 dataset presents a more complex challenge due to its significant variations in mesh connectivity, encompassing 44 shapes and 430 pairs for evaluation.

Results. We conduct experiments on FAUST, SCAPE, and the combination of both datasets. Quantitative results in Tab. 1 show that supervised methods tend to overfit the trained dataset. On the other hand, unsupervised methods typically can achieve a better generalization on new datasets. Compared to Deep Shells, an OT-based method, we outperform in most settings as shown in Tab. 1 and Fig. 2. Compared to state-of-the-art ULRSSM, our method indicates a slightly better mapping demonstrated in Fig. 2.

5.2. Non-isometric Shape Matching

Datasets. We consider SMAL [62] and DeformingThings4D [28, 33] for evaluating non-isometric shape matching. For the SMAL dataset, we adopt the data split in [16] that uses five species for training and three unseen species for testing, resulting in a 29/20 split of the dataset. Regarding DeformingThings4D, denoted as DT4D-H, we follow the split also presented in [16] comprising 198 samples for training and 95 for testing.

Results. To measure the performance on non-isometric datasets, i.e. SMAL and DT4D-H, we compare our method with previous state-of-the-art baselines as shown in Tab. 2. Regarding the DT4D-H dataset, we only perform comparisons on the challenging intra-class scenario. Our proposed method outperforms previous methods in both dataset as shown in Tab. 2. Visualization in Fig. 3 shows that AFMap often fails to retrieve a non-isometric mapping. In addition, ULRSSM demonstrates better mapping despite some ambiguity. On the other hand, our method obtains a precise and

Table 1. **Quantitative results on near-isometric shape matching.** The color denotes the best and second -best result. Our method outperforms various methods including axiomatic, supervised and unsupervised methods in most settings.

Method	FAUST			SCAPE			FAUST + SCAPE		
	FAUST	SCAPE	SHREC'19	FAUST	SCAPE	SHREC'19	FAUST	SCAPE	SHREC'19
				Axiomatic					
ZoomOut [35]	6.1	\	\	\	7.5	\	\	\	\
BCICP [42]	6.1	\	\	\	11.0	\	\	\	\
Smooth Shells [17]	2.5	\	\	\	4.7	\	\	\	\
				Supervised					
FMNet [32]	11.0	30.0	\	33.0	17.0	\	\	\	\
GeomFMaps [15]	2.6	3.3	9.9	3.0	3.0	12.2	2.6	3.0	7.9
TransMatch [55]	1.8	32.8	19.0	18.5	16.0	39.5	1.7	13.5	12.9
				Unsupervised					
SURFMNet [46]	15.0	32.0	\	32.0	12.0	\	33.0	29.0	\
Deep Shells [18]	1.7	5.4	27.4	2.7	2.5	23.4	1.6	2.4	21.1
AFMap [27]	1.9	2.6	6.4	2.2	2.2	9.9	1.9	2.3	5.8
SSLMSM [11]	2.0	7.0	9.1	2.7	3.1	8.4	1.9	4.3	6.2
UDMSM [10]	1.5	7.5	20.1	3.2	2.0	28.3	1.7	7.6	28.7
ULRSSM [12]	1.6	3.6	7.2	1.9	1.9	7.6	1.7	3.2	4.6
Ours	1.5	3.4	5.5	1.6	1.8	7.0	1.6	2.2	4.7

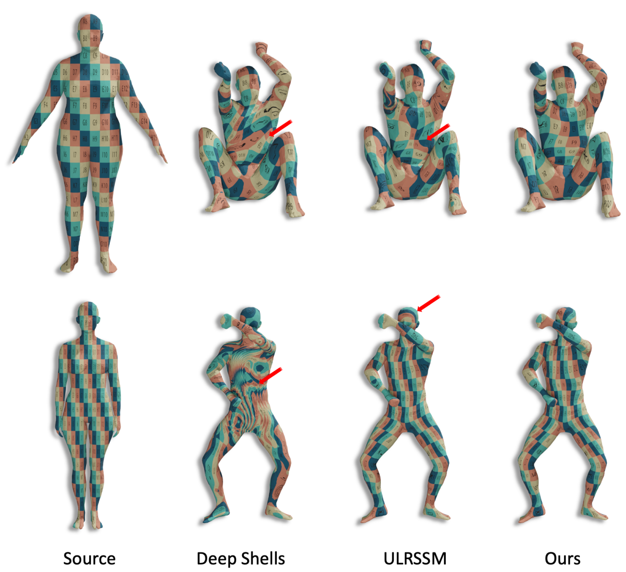

Figure 2. **Qualitative results** of different methods evaluated on SHREC'19 datasets. Correspondence is visualized by texture transfer. The red arrow indicates poor mappings.

Table 2. **Quantitative results for non-isometric matching on SMAL and DT4D-H.** Our method surpass state-of-the-art methods on challenging non-isometric dataset such as SMAL and DT4D-H.

Method	SMAL	DT4D-H
Deep Shells [18]	29.3	31.1
GeoFMaps [15]	7.6	22.6
AFMap [27]	5.4	11.6
ULRSSM [12]	4.2	4.5
Ours	4.0	4.2

Table 3. **Quantitative results for 3D shape segmentation transfer.** Our method is effectively applied to semantic segmentation transfer on 3D shapes, establishing a new benchmark for state-of-the-art performance in this domain.

Method	Coarse	Fine-grained
AFMaps [27]	81.3	43.2
UDMSM [10]	85.3	45.2
ULRSSM [12]	84.2	58.2
Ours	87.8	60.5

smooth mapping, thus visually better than the two state-of-the-art methods.

5.3. Segmentation transfer

Datasets. We illustrate the performance of our proposed method on the task of segmentation transfer on 3D semantic segmentation dataset proposed in [1]. Specifically, the dataset is derived from FAUST [7], which is manually annotated into two types of label: coarse annotations include 4 classes and fine-grained annotations comprise 17 cate-

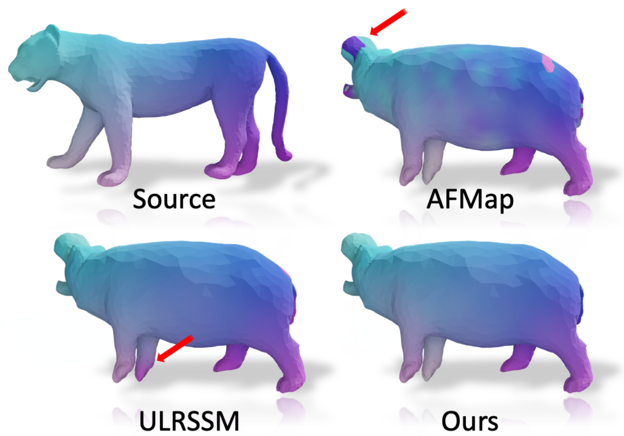

Figure 3. **Qualitative results** of various methods on challenging non-isometric SMAL dataset. Our method demonstrates superior point mapping capabilities compared to previous works. More visualization is provided in Sup. 12.

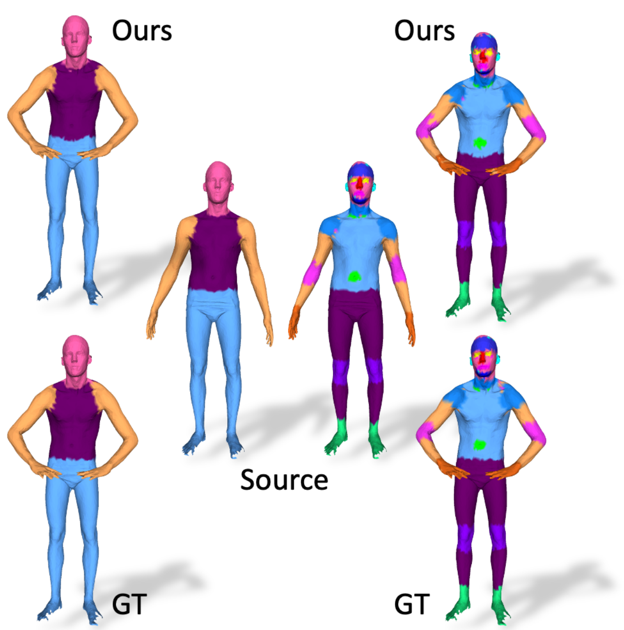

Figure 4. **Qualitative results of segmentation transfer.** Our method exhibits a high-quality segmentation map via computed correspondence. More visualization is provided in Sup. 12.

gories. After excluding non-connected meshes, we test our method on 79 meshes by computing correspondence among the collection and then transferring annotation from one single mesh to the others.

Results. To further demonstrate the robustness, we apply our methods on co-segmentation, also known as segmentation transfer task. We train all methods on the remeshed FAUST_r mentioned in Sec. 5.1. It is worth noting while the FAUST_r is remeshed to around 10K faces, the segmentation dataset in [1] is remeshed to 20K triangular faces. Therefore, it showcases the generalization of our method that does not depend on the discretization and resolution of mesh. Tab. 3 indicates that our method sets a new state-of-the-art on the segmentation-transfer task on FAUST [1] dataset in both coarse and fine-grained annotation. Fig. 4 shows that our method is very closed to ground truth without the need for training semantic segmentation models.

6. Ablation study

Table 4. **Ablation study on SHREC'19.** In the first setting, we replace $\mathcal{L}_{OT}$ with $\mathcal{L}_{MSE}$ in Eq. 13. In the second row, we substitute $\mathcal{L}_{OT}$ with $\mathcal{L}_{uniSW}$. The third row indicates the $\mathcal{L}_{OT}$ being $\mathcal{L}_{biSW}$ as in Eq. 10. The fourth row indicates not using adaptive refinement at the end of the training process.

Ablation Setting	SHREC'19
w. $\mathcal{L}_{MSE}$	34.3
w. $\mathcal{L}_{uniSW}$	4.9
w. $\mathcal{L}_{biSW}$	4.8
w.o. adaptive refinement	7.2
Ours	**4.7**

Settings. We conduct an ablation study to validate our contribution. We train our model on FAUST+SCAPE dataset and evaluate it on SHREC'19 dataset. Firstly, we evaluate the effectiveness of different losses in the feature alignment component. Furthermore, we also investigate the importance of the adaptive refinement module.

Results. Our results are summarized in Tab. 4. First of all, by comparing the first row with the last row, we conclude that $\mathcal{L}_{MSE}$ can not learn to align features for retrieving point-to-point correspondence. Secondly, we observe that by using bidirectional SW, we can gain a slightly better performance. Finally, the last row indicates that by employing importance sampling energy-based SW, we can even gain better performance.

7. Conclusion

In conclusion, we introduce an innovative framework that integrates functional maps with an efficient optimal transport method, notably the sliced Wasserstein distance, to address computational challenges and enhance feature alignment. Our approach significantly outperforms existing methods in non-rigid shape matching across various scenarios, including both near-isometric and non-isometric forms. This advancement, confirmed through successful applications in tasks like segmentation transfer, highlights our method's efficacy and strong generalization potential in shape matching.

References

[1] Ahmed Abdelreheem, Ivan Skorokhodov, Maks Ovsjanikov, and Peter Wonka. Satr: Zero-shot semantic segmentation of 3d shapes. *arXiv preprint arXiv:2304.04909*, 2023. 6, 7, 8

[2] Jason Altschuler, Jonathan Niles-Weed, and Philippe Rigollet. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. In *Advances in Neural Information Processing Systems*, pages 1964–1974, 2017. 2

[3] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM SIGGRAPH*. 2005. 6

[4] Souhaib Attaiki and Maks Ovsjanikov. Ncp: Neural correspondence prior for effective unsupervised shape matching. In *NeurIPS*, 2022. 1

[5] Souhaib Attaiki, Gautam Pai, and Maks Ovsjanikov. Dpfm: Deep partial functional maps. In *International Conference on 3D Vision (3DV)*, 2021. 1

[6] Mathieu Aubry, Ulrich Schlickewei, and Daniel Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *ICCV*, 2011. 1

[7] Federica Bogo, Javier Romero, Matthew Loper, and Michael J Black. Faust: Dataset and evaluation for 3d mesh registration. In *CVPR*, 2014. 1, 6, 7

[8] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51:22–45, 2015. 2, 3

[9] Michael M Bronstein and Iasonas Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *CVPR*, 2010. 1

[10] Dongliang Cao and Florian Bernard. Unsupervised deep multi-shape matching. In *ECCV*, 2022. 1, 2, 6, 7

[11] Dongliang Cao and Florian Bernard. Self-supervised learning for multimodal non-rigid 3d shape matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17735–17744, 2023. 1, 5, 6, 7

[12] Dongliang Cao, Paul Roetzer, and Florian Bernard. Unsupervised learning of robust spectral shape matching. *arXiv preprint arXiv:2304.14419*, 2023. 1, 2, 5, 6, 7

[13] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013. 2, 3, 6

[14] Huong Quynh Dinh, Anthony Yezzi, and Greg Turk. Texture transfer during shape transformation. *ACM Transactions on Graphics (ToG)*, 24(2):289–310, 2005. 1

[15] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep geometric functional maps: Robust feature learning for shape correspondence. In *CVPR*, 2020. 6, 7

[16] Nicolas Donati, Etienne Corman, and Maks Ovsjanikov. Deep orientation-aware functional maps: Tackling symmetry issues in shape matching. In *CVPR*, 2022. 1, 2, 6

[17] Marvin Eisenberger, Zorah Lahner, and Daniel Cremers. Smooth shells: Multi-scale shape registration with functional maps. In *CVPR*, 2020. 3, 6, 7

[18] Marvin Eisenberger, Aysim Toker, Laura Leal-Taixé, and Daniel Cremers. Deep shells: Unsupervised shape correspondence with optimal transport. *NIPS*, 2020. 1, 3, 6, 7

[19] Danielle Ezuz and Mirela Ben-Chen. Deblurring and denoising of maps between shapes. In *Computer Graphics Forum*. Wiley Online Library, 2017. 3

[20] Dvir Ginzburg and Dan Raviv. Cyclic functional mapping: Self-supervised correspondence between non-isometric deformable shapes. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 36–52. Springer, 2020. 1

[21] Oshri Halimi, Or Litany, Emanuele Rodola, Alex M Bronstein, and Ron Kimmel. Unsupervised learning of dense shape correspondence. In *CVPR*, 2019. 1, 2

[22] Qixing Huang, Fan Wang, and Leonidas Guibas. Functional map networks for analyzing and exploring large shape collections. *ACM Transactions on Graphics (ToG)*, 33(4):1–11, 2014. 2

[23] Sangpil Kim, Hyung-gun Chi, Xiao Hu, Qixing Huang, and Karthik Ramani. A large-scale annotated mechanical components benchmark for classification and retrieval tasks with deep neural networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 175–191. Springer, 2020. 6

[24] Vladimir G Kim, Yaron Lipman, and Thomas Funkhouser. Blended intrinsic maps. *ACM Transactions on Graphics (ToG)*, 30(4):1–12, 2011. 6

[25] Artiom Kovnatsky, Michael M Bronstein, Alexander M Bronstein, Klaus Glashoff, and Ron Kimmel. Coupled quasi-harmonic bases. In *Computer Graphics Forum*. Wiley Online Library, 2013. 2

[26] Tung Le, Khai Nguyen, Shanlin Sun, Kun Han, Nhat Ho, and Xiaohui Xie. Diffeomorphic deformation via sliced wasserstein distance optimization for cortical surface reconstruction. *arXiv preprint arXiv:2305.17555*, 2023. 3

[27] Lei Li, Nicolas Donati, and Maks Ovsjanikov. Learning multi-resolution functional maps with spectral attention for robust shape matching. *NIPS*, 2022. 1, 2, 5, 6, 7

[28] Yang Li, Hikari Takehara, Takafumi Taketomi, Bo Zheng, and Matthias Nießner. 4dcomplete: Non-rigid motion estimation beyond the observable surface. In *ICCV*, 2021. 6

[29] Tianyi Lin, Nhat Ho, and Michael Jordan. On efficient optimal transport: An analysis of greedy and accelerated mirror descent algorithms. In *International Conference on Machine Learning*, pages 3982–3991, 2019. 2

[30] Tianyi Lin, Nhat Ho, Xi Chen, Marco Cuturi, and Michael I. Jordan. Fixed-support Wasserstein barycenters: Computational hardness and fast algorithm. In *NeurIPS*, pages 5368–5380, 2020.

[31] Tianyi Lin, Nhat Ho, and Michael I. Jordan. On the efficiency of entropic regularized algorithms for optimal transport. *Journal of Machine Learning Research (JMLR)*, 23: 1–42, 2022. 2

[32] Or Litany, Tal Remez, Emanuele Rodola, Alex Bronstein, and Michael Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. In *ICCV*, 2017. 1, 2, 6, 7

[33] Robin Magnet, Jing Ren, Olga Sorkine-Hornung, and Maks Ovsjanikov. Smooth non-rigid shape matching via effective dirichlet energy optimization. In *International Conference on 3D Vision (3DV)*, 2022. 6

[34] Simone Melzi, Riccardo Marin, Emanuele Rodolà, Umberto Castellani, Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. Shrec 2019: Matching humans with different connectivity. In *Eurographics Workshop on 3D Object Retrieval*, 2019. 6

[35] Simone Melzi, Jing Ren, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. Zoomout: spectral upsampling for efficient shape correspondence. *ACM Transactions on Graphics (ToG)*, 38(6):1–14, 2019. 3, 6, 7

[36] Facundo Mémoli. Gromov–wasserstein distances and the metric approach to object matching. *Foundations of computational mathematics*, 11:417–487, 2011. 2

[37] Khai Nguyen and Nhat Ho. Energy-based sliced wasserstein distance. *Advances in Neural Information Processing Systems*, 2023. 4, 5

[38] Trung Nguyen, Quang-Hieu Pham, Tam Le, Tung Pham, Nhat Ho, and Binh-Son Hua. Point-set distances for learning representations of 3d point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10478–10487, 2021. 3

[39] Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (ToG)*, 31(4):1–11, 2012. 1, 2, 3

[40] Maks Ovsjanikov, Etienne Corman, Michael Bronstein, Emanuele Rodolà, Mirela Ben-Chen, Leonidas Guibas, Frederic Chazal, and Alex Bronstein. Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses*, pages 1–60. 2016. 2

[41] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019. 1

[42] Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. Continuous and orientation-preserving correspondences via functional maps. *ACM Transactions on Graphics (ToG)*, 37:1–16, 2018. 3, 6, 7

[43] Jing Ren, Mikhail Panine, Peter Wonka, and Maks Ovsjanikov. Structured regularization of functional map computations. In *Computer Graphics Forum*. Wiley Online Library, 2019. 4, 5

[44] Jing Ren, Simone Melzi, Peter Wonka, and Maks Ovsjanikov. Discrete optimization for shape matching. In *Computer Graphics Forum*. Wiley Online Library, 2021. 5

[45] Emanuele Rodolà, Luca Cosmo, Michael M Bronstein, Andrea Torsello, and Daniel Cremers. Partial functional correspondence. In *Computer Graphics Forum*. Wiley Online Library, 2017. 2

[46] Jean-Michel Roufosse, Abhishek Sharma, and Maks Ovsjanikov. Unsupervised deep learning for structured shape matching. In *ICCV*, 2019. 1, 2, 6, 7

[47] Yusuf Sahillioğlu. Recent advances in shape correspondence. *The Visual Computer*, 36(8):1705–1721, 2020. 2

[48] Samuele Salti, Federico Tombari, and Luigi Di Stefano. Shot: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, 125:251–264, 2014. 1, 2, 3

[49] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *arXiv preprint arXiv:2012.00888*, 2020. 2, 3, 4

[50] Zhengyang Shen, Jean Feydy, Peirong Liu, Ariel H Curiale, Ruben San Jose Estepar, Raul San Jose Estepar, and Marc Niethammer. Accurate point cloud registration with robust optimal transport. *Advances in Neural Information Processing Systems*, 34:5373–5389, 2021. 2

[51] Justin Solomon, Gabriel Peyré, Vladimir G Kim, and Suvrit Sra. Entropic metric alignment for correspondence problems. *ACM Transactions on Graphics (ToG)*, 35(4):1–13, 2016. 2

[52] Robert W Sumner and Jovan Popović. Deformation transfer for triangle meshes. *ACM Transactions on Graphics (ToG)*, 23(3):399–405, 2004. 1

[53] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer graphics forum*, pages 1383–1392. Wiley Online Library, 2009. 2

[54] Mingze Sun, Shiwei Mao, Puhua Jiang, Maks Ovsjanikov, and Ruqi Huang. Spatially and spectrally consistent deep functional maps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14497–14507, 2023. 1

[55] Giovanni Trappolini, Luca Cosmo, Luca Moschella, Riccardo Marin, Simone Melzi, and Emanuele Rodolà. Shape registration in the time of transformers. *NIPS*, 2021. 6, 7

[56] Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. In *Computer graphics forum*, pages 1681–1707. Wiley Online Library, 2011. 2

[57] Cédric Villani. *Topics in optimal transportation*. Number 58. American Mathematical Soc., 2003. 3

[58] Cédric Villani. *Topics in optimal transportation*. American Mathematical Soc., 2021. 5

[59] Cédric Villani et al. *Optimal transport: old and new*. Springer, 2009. 1

[60] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pretraining for 3d point cloud understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 574–591. Springer, 2020. 5

[61] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 766–782. Springer, 2016. 1

[62] Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *CVPR*, 2017. 6

Integrating Efficient Optimal Transport and Functional Maps For Unsupervised Shape Correspondence Learning

Supplementary Material

In this supplementary, we first define some notations that are used in our main paper and supplementary in Sec. 8. We then discuss some limitations of our work and potential future directions to address them in Sec. 9. In Sec. 10, we provide detailed computation and algorithm to compute the proposed loss functions. Furthermore, we delineate the implementation details and hyperparameters used in our training process in Sec. 11. Finally, we provide additional qualitative results of our proposed approach in Sec. 12.

8. Notations

For any $d \geq 2$, we denote $\mathbb{S}^{d-1} := \{\theta \in \mathbb{R}^d \mid ||\theta||_2^2 = 1\}$ and $\mathcal{U}(\mathbb{S}^{d-1})$ as the unit hyper-sphere and its corresponding uniform distribution. We denote $\theta\sharp\mu$ as the push-forward measures of μ through the function $f : \mathbb{R}^d \to \mathbb{R}$ that is $f(x) = \theta^\top x$. Furthermore, we denote $\mathcal{P}(\mathcal{X})$ as the set of all probability measures on the set $\mathcal{X}$. For $p \geq 1$, $\mathcal{P}_p(\mathcal{X})$ is the set of all probability measures on the set $\mathcal{X}$ that have finite p-moments.

9. Limitations and discussion

Our work is the first to integrate an efficient optimal transport to functional map framework for shape correspondence, yet it is not without limitations, potentially opening new research directions. First of all, our algorithm is designed for use with clean and complete meshes. An intriguing avenue for future research would be to extend the applicability of our method to more diverse scenarios, such as dealing with partial meshes, noisy point clouds, and other forms of data representation. This expansion would enhance the versatility of our approach in handling a wider range of practical applications. Secondly, our adaptive refinement module, which utilizes an entropic regularized optimal transport for estimating the soft-feature similarity matrix, shows promise in achieving more precise refinement. However, this method is not without its drawbacks, notably a quadratic increase in memory complexity and computational demand. This presents a challenge that future research could address by developing more computationally efficient approximations, thereby making the process more feasible for larger datasets or more resource-constrained environments. Overall, these potential research directions could significantly contribute to the evolution of shape correspondence methodologies.

10. Detailed algorithms and discussion

Sliced Wasserstein distance. The unidirectional sliced Wasserstein distance version of Eq. 10 is given by:

$$\mathcal{L}_{uniSW} = (\mathbb{E}_{\theta\sim\mathcal{U}(\mathbb{S}^{d-1})} \mathrm{W}_p^p(\theta\sharp\mathcal{F}_x, \theta\sharp\hat{\mathcal{F}}_y))^{\frac{1}{p}}, \tag{15}$$

where $\hat{\mathcal{F}}_y = \hat{\Pi}_{xy}\mathcal{F}_y$. The unidirectional sliced Wasserstein distance given in Eq. 15 is computed by using L Monte Carlo samples $\theta_1, ..., \theta_L$ from the unit sphere:

$$\widehat{\mathcal{L}_{uniSW}} = \left(\frac{1}{L}\sum_{l=1}^{L} \mathrm{W}_p^p(\theta_l\sharp\mathcal{F}_x, \theta_l\sharp\hat{\mathcal{F}}_y)\right)^{\frac{1}{p}}, \tag{16}$$

where $\mathrm{W}_p^p(\theta\sharp\mathcal{F}_x, \theta\sharp\hat{\mathcal{F}}_y) = \int_0^1 |F^{-1}_{\theta\sharp\mathcal{F}_x}(z) - F^{-1}_{\theta\sharp\hat{\mathcal{F}}_y}(z)|^p dz$ denotes the closed form solution one-dimensional Wasserstein distance of two probability measures $\mathcal{F}_x$ and $\hat{\mathcal{F}}_y$. Here, $F_{\theta\sharp\mathcal{F}_x}$ and $F_{\theta\sharp\hat{\mathcal{F}}_y}$ are the cumulative distribution function (CDF) of $\theta\sharp\mathcal{F}_x$ and $\theta\sharp\hat{\mathcal{F}}_y$ respectively.

Similarly, the bidirectional sliced Wasserstein distance in Eq. 10 is also estimated by using L Monte Carlo samples $\theta_1, ..., \theta_L$ from the unit sphere:

$$\begin{aligned}\widehat{\mathcal{L}_{biSW}} = (\frac{1}{L}\sum_{l=1}^{L}[&\mathrm{W}_p^p(\theta_l\sharp\mathcal{F}_x, \theta_l\sharp\hat{\mathcal{F}}_y) \\ &+ \mathrm{W}_p^p(\theta_l\sharp\mathcal{F}_y, \theta_l\sharp\hat{\mathcal{F}}_x)])^{\frac{1}{p}},\end{aligned} \tag{17}$$

where $\hat{\mathcal{F}}_x = \hat{\Pi}_{yx}\mathcal{F}_x$ and $\hat{\mathcal{F}}_y = \hat{\Pi}_{xy}\mathcal{F}_y$. We provide a pseudo-code for computing the unidirectional and bidirectional sliced Wasserstein distance in Algorithm 1 and Algorithm 2, respectively.

Energy-based sliced Wasserstein distance. The unidirectional sliced Wasserstein distance version of Eq. 11 is defined as:

$$\mathcal{L}_{uniEBSW} = \left(\frac{\mathbb{E}_{\theta\sim\sigma_0(\theta)}[\mathrm{W}_{\theta,\mathcal{X}}w(\theta)]}{\mathbb{E}_{\theta\sim\sigma_0(\theta)}[w(\theta)]}\right)^{\frac{1}{p}}, \tag{18}$$

where we denote $\mathrm{W}_{\theta,\mathcal{X}} := \mathrm{W}_p^p(\theta\sharp\mathcal{F}_x, \theta\sharp\hat{\mathcal{F}}_y)$, $w(\theta) := \frac{\exp(\mathrm{W}_{\theta,\mathcal{X}})}{\sigma_0(\theta)}$, and $\sigma_0(\theta) \in \mathcal{P}(\mathbb{S}^{d-1})$ denotes the proposed distribution. The unidirectional energy-based sliced Wasserstein distance given in Eq. 18 can be computed via importance sampling estimator L Monte Carlo $\theta_1, ..., \theta_L$ sampled from $\sigma_0(\theta)$:

$$\widehat{\mathcal{L}_{uniEBSW}} = \left(\frac{1}{L}\sum_{l=1}^{L}[\mathrm{W}_{\theta_l,\mathcal{X}}\tilde{w}(\theta_l)]\right)^{\frac{1}{p}}, \tag{19}$$

Algorithm 1 Computational algorithm of the unidirectional SW distance

Input: Features extracted from feature extractor module $\mathcal{F}_x, \mathcal{F}_y$; $p \geq 1$; soft features similarity $\hat{\Pi}$ from Eq. 9; and the number of projections L.

Compute $\hat{\mathcal{F}}_y = \hat{\Pi}_{xy}\mathcal{F}_y$
for $l = 1$ to L **do**
  Sample $\theta_l \sim \mathcal{U}(\mathbb{S}^{d-1})$
  Compute $v_l = \mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_x, \theta_l \sharp \hat{\mathcal{F}}_y)$
end for
Compute $\widehat{\mathcal{L}_{uniSW}} = \left(\frac{1}{L}\sum_{l=1}^{L} v_l\right)^{\frac{1}{p}}$
Return: $\widehat{\mathcal{L}_{uniSW}}$

Algorithm 2 Computational algorithm of the bidirectional SW distance

Input: Features extracted from feature extractor module $\mathcal{F}_x, \mathcal{F}_y$; $p \geq 1$; soft features similarity $\hat{\Pi}$ from Eq. 9; and the number of projections L.

Compute $\hat{\mathcal{F}}_x = \hat{\Pi}_{yx}\mathcal{F}_x$ and $\hat{\mathcal{F}}_y = \hat{\Pi}_{xy}\mathcal{F}_y$
for $l = 1$ to L **do**
  Sample $\theta_l \sim \mathcal{U}(\mathbb{S}^{d-1})$
  Compute $v_l = \mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_x, \theta_l \sharp \hat{\mathcal{F}}_y) + \mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_y, \theta_l \sharp \hat{\mathcal{F}}_x)$
end for
Compute $\widehat{\mathcal{L}_{biSW}} = \left(\frac{1}{L}\sum_{l=1}^{L} v_l\right)^{\frac{1}{p}}$
Return: $\widehat{\mathcal{L}_{biSW}}$

Algorithm 3 Computational algorithm of the unidirectional EBSW distance

Input: Features extracted from feature extractor module $\mathcal{F}_x, \mathcal{F}_y$; $p \geq 1$; soft features similarity $\hat{\Pi}$ from Eq. 9; and the number of projections L.

Compute $\hat{\mathcal{F}}_y = \hat{\Pi}_{xy}\mathcal{F}_y$
for $l = 1$ to L **do**
  Sample $\theta_l \sim \mathcal{U}(\mathbb{S}^{d-1})$
  Compute $v_l = \mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_x, \theta_l \sharp \hat{\mathcal{F}}_y)$
  Compute $w_l = f(\mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_x, \theta_l \sharp \hat{\mathcal{F}}_y))$
end for
Compute $\widehat{\mathcal{L}_{uniEBSW}} = \left(\frac{1}{L}\sum_{l=1}^{L} v_l \frac{w_l}{\sum_{i=1}^{L} w_i}\right)^{\frac{1}{p}}$
Return: $\widehat{\mathcal{L}_{uniEBSW}}$

where $\tilde{w}(\theta_l) := \frac{w(\theta_l)}{\sum_{l'=1}^{L} w(\theta_{l'})}$. When $\sigma_0(\theta) = \mathcal{U}(\mathbb{S}^{d-1}) = \frac{\Gamma(d/2)}{2\pi^{d/2}}$ (a constant of θ) [37], we substitute $w(\theta_l)$ with $f(\mathrm{W}_{\theta_l,\mathcal{X}})$. We can choose the energy function $f(x) = e^x$, then the normalized importance weights become the Softmax function of $\mathrm{W}_{\theta,\mathcal{X}}$ as follows:

$$\tilde{w}(\theta_l) = \mathrm{Softmax}(\mathrm{W}_{\theta_l,\mathcal{X}}) = \frac{\exp(\mathrm{W}_{\theta_l,\mathcal{X}})}{\sum_{l'=1}^{L}\exp(\mathrm{W}_{\theta_{l'},\mathcal{X}})}$$

Based on the computation of unidirectional energy-based sliced Wasserstein distance, we can compute the bidirectional energy-based sliced Wasserstein distance, i.e. $\mathcal{L}_{biEBSW}$, in Eq. 11 as follows:

$$\widehat{\mathcal{L}_{biEBSW}} = \left(\frac{1}{L}\sum_{l=1}^{L}[(\mathrm{W}_{\theta_l,\mathcal{X}} + \mathrm{W}_{\theta_l,\mathcal{Y}})\hat{w}(\theta_l)]\right)^{\frac{1}{p}}, \quad (20)$$

where we denote $\mathrm{W}_{\theta,\mathcal{Y}} := \mathrm{W}_p^p(\theta \sharp \mathcal{F}_y, \theta \sharp \hat{\mathcal{F}}_x)$, and $\hat{w}(\theta_l) := \frac{\exp(\mathrm{W}_{\theta_l,\mathcal{X}} + \mathrm{W}_{\theta_l,\mathcal{Y}})}{\sum_{l'=1}^{L}\exp(\mathrm{W}_{\theta_{l'},\mathcal{X}} + \mathrm{W}_{\theta_{l'},\mathcal{Y}})}$. It is worth noting that the importance weights of $\widehat{\mathcal{L}_{biEBSW}}$ in Eq. 20 are different from that

Algorithm 4 Computational algorithm of the bidirectional EBSW distance

Input: Features extracted from feature extractor module $\mathcal{F}_x, \mathcal{F}_y$; $p \geq 1$; soft features similarity $\hat{\Pi}$ from Eq. 9; and the number of projections L.

Compute $\hat{\mathcal{F}}_x = \hat{\Pi}_{yx}\mathcal{F}_x$ and $\hat{\mathcal{F}}_y = \hat{\Pi}_{xy}\mathcal{F}_y$
for $l = 1$ to L **do**
  Sample $\theta_l \sim \mathcal{U}(\mathbb{S}^{d-1})$
  Compute $v_l = \mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_x, \theta_l \sharp \hat{\mathcal{F}}_y) + \mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_y, \theta_l \sharp \hat{\mathcal{F}}_x)$
  Compute $w_l = f(\mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_x, \theta_l \sharp \hat{\mathcal{F}}_y) + \mathrm{W}_p^p(\theta_l \sharp \mathcal{F}_y, \theta_l \sharp \hat{\mathcal{F}}_x))$
end for
Compute $\widehat{\mathcal{L}_{biEBSW}} = \left(\frac{1}{L}\sum_{l=1}^{L} v_l \frac{w_l}{\sum_{i=1}^{L} w_i}\right)^{\frac{1}{p}}$
Return: $\widehat{\mathcal{L}_{biEBSW}}$

Algorithm 5 Algorithm of the adaptive refinement

Input: Pair shapes $\mathcal{X}, \mathcal{Y}$ with their Laplace-Beltrami operators Φ_x, Φ_y . Trained model with parameter $\mathcal{G}_\Theta$. Number of refinement steps T.

while reach T **do**
  Compute $\mathcal{F}_x = \mathcal{G}_\Theta(\mathcal{X}, \Phi_x)$ and $\mathcal{F}_y = \mathcal{G}_\Theta(\mathcal{Y}, \Phi_y)$. ▷ Extract features.
  Compute $C_{xy}, C_{yx} = \text{FMSolver}(\mathcal{F}_x, \mathcal{F}_y, \Phi_x, \Phi_y)$. ▷ Find functional map via FM solver.
  Compute $\tilde{\Pi}_{xy}, \tilde{\Pi}_{yx} = \text{Sinkhorn}(\mathcal{F}_x, \mathcal{F}_y)$. ▷ Estimate pseudo similarity matrix via Sinkhorn.
  Compute unsupervised losses $\mathcal{L}_{total}(\mathcal{F}_x, \mathcal{F}_y, C_{xy}, C_{yx}, \tilde{\Pi}_{xy}, \tilde{\Pi}_{yx})$.
  Update features and soft similarity matrix by minimizing $\mathcal{L}_{total}$.
end while
Compute $P = NN(\mathcal{F}_x, \mathcal{F}_y)$ ▷ Compute point-to-point correspondence via nearest neighbor search.
Return: P

of $\widehat{\mathcal{L}_{uniEBSW}}$ in Eq. 19, since the slicing distribution here is shared and affected by both one-dimensional Wasserstein distances, thus providing a more expressive projecting features for computing sliced Wasserstein distance. We provide a pseudo-code for computing the unidirectional and bidirectional energy-based sliced Wasserstein distance in Algorithm 3 and Algorithm 4, respectively.

Adaptive refinement. As discussed in Sec. 4.5, we refine our correspondence result by estimating the pseudo-soft correspondence via entropic regularized optimal transport. The pseudo-code for our adaptive refinement is given in Algorithm 5.

11. Implementation details

All experiments are implemented using Pytorch 2.0, and executed on a system equipped with an NVIDIA GeForce RTX GPU 2080 Ti and an Intel Xeon(R) Gold 5218 CPU. We employ DiffusionNet [49] as the feature extraction mechanism, with wave kernel signatures (WKS) [6] serving as the input features. The dimension of the WKS is set to 128 for all of our experiments. Regarding spectral resolution, we opt for the first 200 eigenfunctions derived from the Laplacian matrices to form the spectral embedding. The output features of the feature extractor are set to 256. During training, the value of the learning rate is set to $1e-3$ with cosine annealing to the minimum learning rate of $1e-4$. The network is optimized with Adam optimizer with batch size 1. About adaptive refinement, the number of refinement iterations is empirically set to 12.

Regarding the loss functions, as stated in Eq. 13, we empirically set $\lambda_1 = \lambda_3 = 1.0, \lambda_2 = 100.0$. About the weight for each component of $\mathcal{L}_{fmap}$ in Eq. 8, we set $\alpha_1 = \alpha_2 = 1.0$. Regarding Sliced Wasserstein distance and energy-based sliced Wasserstein distance, we set $p = 2, L = 200$ for all of our experiments.

12. Additional visualizations

In this section, we provide additional visualizations of our proposed approach on multiple datasets.

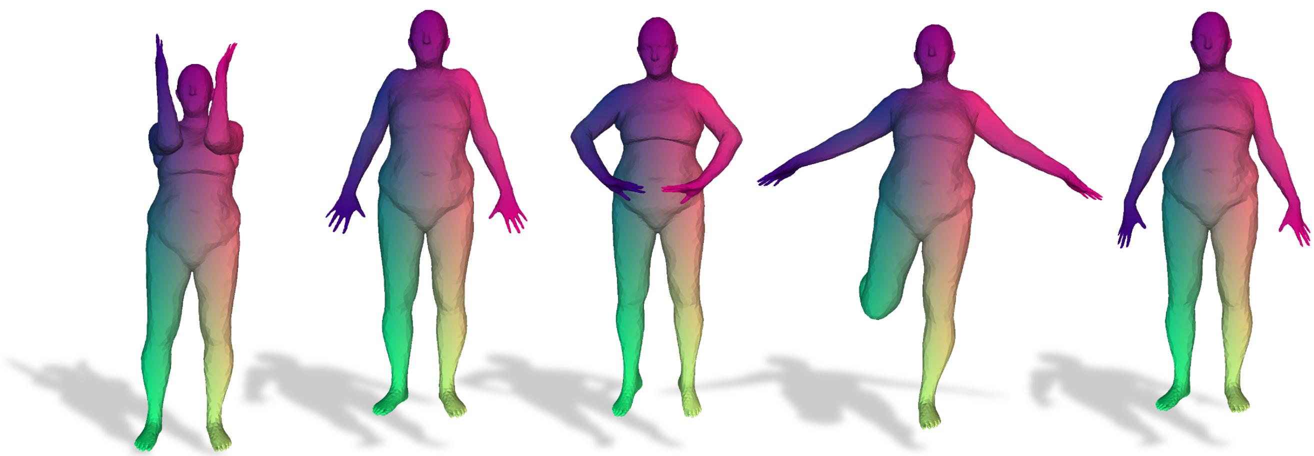

Figure 5. Qualitative results of our method on FAUST dataset.

Figure 6. Qualitative results of our method on SCAPE dataset.

Figure 7. Qualitative results of our method on SHREC dataset.

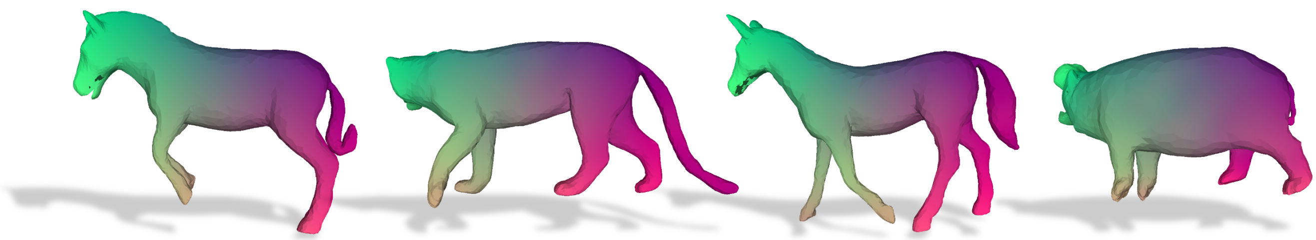

Figure 8. Qualitative results of our method on SMAL dataset.

Figure 9. Qualitative results of our method on DT4D-H dataset.

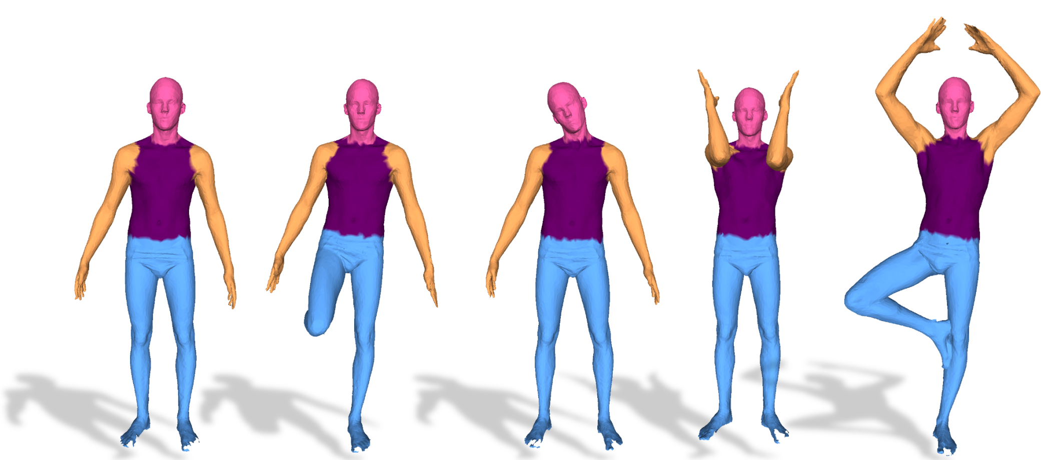

Figure 10. Qualitative results of our method on segmentation transfer coarse FAUST dataset.

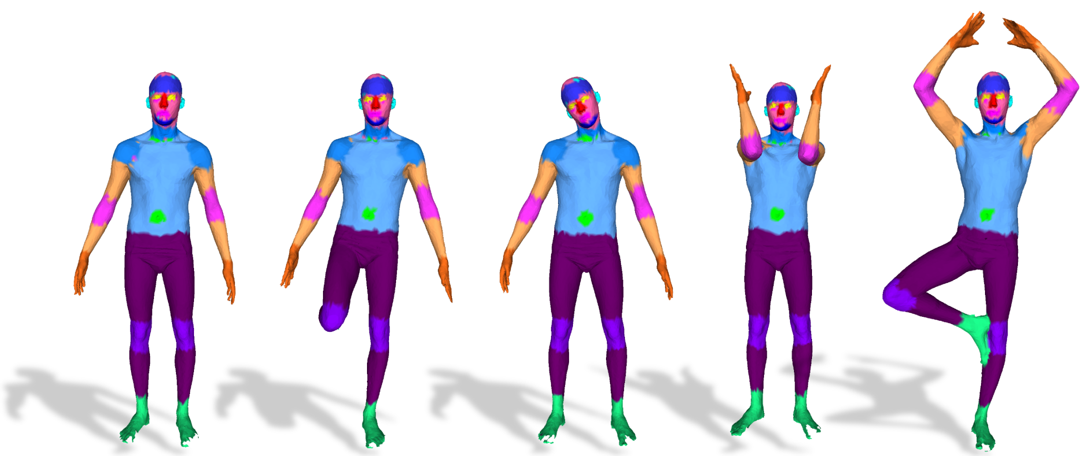

Figure 11. Qualitative results of our method on segmentation transfer fine-grained FAUST dataset.