This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DV-Matcher: Deformation-based Non-Rigid Point Cloud Matching Guided by Pre-trained Visual Features

Zhangquan Chen[1#] Puhua Jiang[1,2#] Ruqi Huang[1*]
1. Tsinghua Shenzhen International Graduate School, China 2. Pengcheng Laboratory, China

Abstract

In this paper, we present DV-Matcher, a novel learning-based framework for estimating dense correspondences between non-rigidly deformable point clouds. Learning directly from unstructured point clouds without meshing or manual labelling*, our framework delivers high-quality dense correspondences, which is of significant practical utility in point cloud processing. Our key contributions are two-fold: First, we propose a scheme to inject prior knowledge from pre-trained vision models into geometric feature learning, which effectively complements the local nature of geometric features with global and semantic information; Second, we propose a novel deformation-based module to promote the extrinsic alignment induced by the learned correspondences, which effectively enhances the feature learning. Experimental results show that our method achieves state-of-the-art results in matching non-rigid point clouds in both near-isometric and heterogeneous shape collection as well as more realistic partial and noisy data. Our code is available at* https://github.com/rqhuang88/DV-Matcher.

1. Introduction

Point cloud data has long been the most prevailing form of 3D data acquisitions (e.g., laser scanning, photogrammetry). In this paper, we propose **DV-Matcher**, a learning-based framework for matching *non-rigidly deformable* point clouds – a task that is essential for reconstructing [63], understanding [4] and manipulating [52] *dynamic* objects in real world.

Apart from the primary pursuit of accurate dense correspondences, we have extended our effort towards the following goals for boosting the utility and potential of our framework in real-world applications: G1) **Robustness** with respect to significant deformations and partiality; G2) **Balance between Accuracy and Efficiency** for processing large-scale data; G3) **Correspondence Label-Free** for minimizing manual effort from users; G4) **Directly trainable on point clouds of unfixed resolution**, without introducing extra operations such as meshing and point re-sampling in training or inference. Though, as will be discussed soon, different combinations of the above goals have been achieved by the prior arts, DV-Matcher demonstrates high capacity along *all* axes, but also delivers state-of-the-art matching accuracy (see Fig. 1 for a challenging example).

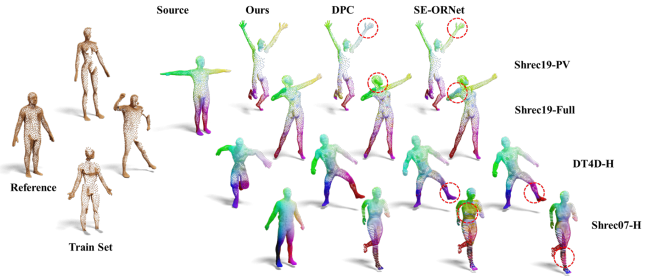


Figure 1. We train DV-Matcher and two baselines under a challenging setting. The training set consists of *only one full point cloud as reference and a set of* 516 *partial point clouds* sampled from shapes in SHREC'19 with significant pose/style deformations (see the yellow point clouds). Without any correspondence label, our framework not only manages to match accurately on SHREC'19 benchmark (*both* partial and full setting), but also generalizes well to unseen benchmarks. See Sec. 4.2 for more details.

In fact, estimating correspondences between non-rigid point clouds has long attracted research interests. Early approaches [3, 34, 39]are mostly axiomatic and featured by lightweight and efficient implementation, which fulfill G2-G4 above. When the deformations among input are small to moderate, which is typical in dynamic 3D reconstruction [47, 63], such algorithms are effective. On the other hand, due to their dependence on spatial proximities of input, they often struggle in the presence of large deformation in either pose/style [29]. Some recent advances aim for robustness regarding pose variation [66], which on the other hand falls short of handling style variation and partiality (see Sec. 4).

To deal with large deformation, a recent trend [9, 29, 30, 43] is to transplant the success of matching triangular meshes via *spectral method* [51] into the domain of point clouds. In essence, such approaches follow a self-supervised scheme

*Corresponding author: ruqihuang@sz.tsinghua.edu.cn

built on the bijective mapping between a mesh and its vertex set, and therefore passing the *intrinsic-geometry-aware* features to the point-based encoder. While achieving great performance, these methods all require triangular meshes during training. In fact, all the above methods are trained on existing mesh benchmarks, falling short of meeting G4.

In parallel, there exists another line of works [15, 21, 32, 64] learning to match without more advanced data representation. In particular, they leverage cross-reconstruction between point clouds induced by the estimated soft-maps as a proxy task for learning a feature extractor. While they satisfy G1-G4 to some extent, the reconstruction process does not physically align the input point clouds and suffers from mode collapsing, hindering achieving high-quality correspondences.

The above observations indeed reveal two critical points in designing a strong correspondence estimator – 1) **Easy-to-obtain matching cues other than spatial proximities** and 2) **A good proxy task for verifying and guiding correspondence learning**.

For the former, instead of endowing surface geometry to point clouds (*e.g.,* meshing), we take an easier path of projecting them into 2D images. Then we aggregate the regarding semantic meaningful features extracted from multi-view projections by pre-trained vision models back to 3D, which empirically serve as high-quality matching cues. In fact, the idea of applying pre-trained vision models in shape analysis has recently attracted considerable attention [1, 17, 46, 60]. Nevertheless, we emphasize that these methods essentially leverage pre-trained vision models to generate intermediate cues (*e.g.,* landmark correspondences) to assist 3D tasks, which can be error-prone. In contrast, our approach treats the aggregated features as visual encoding (similar to positional encoding [45]), and trains a strong feature extractor guided by the following deformation-based proxy task.

For the latter, we advocate correspondence-induced non-rigid shape deformation as the proxy task of choice. While the advantage of such proxy task is obvious, the prior arts [19, 29] based on deformation take an iterative optimization approach, which prevents efficient inference (G2). On the other hand, the existing end-to-end deformation frameworks [28, 58] require correspondence supervision (G3). In contrast, thanks to the visual encoding as well as tailored-for network design for exploit both visual and geometric information (see Sec. 3), our DV-Matcher achieve both accuracy and efficiency in deformation-based feature learning without correspondence label.

We conduct a rich set of experiments to verify the effectiveness of our pipeline, highlighting that it achieves state-of-the-art results in matching non-rigid point clouds under various settings including near-isometric, heterogeneous, full, partial and even realistic scanned point clouds from a range of categories. Remarkably, it generalizes well despite the distinctiveness between the training set and test set.

2. Related Works

2.1. Non-rigid Shape Matching

Non-rigid shape matching is a long-standing problem in computer vision and graphics. Unlike the rigid counterpart, non-rigidly aligning shapes is more challenging owing to the complexity inherent in deformation models.

Originating from the foundational work on functional maps [51], along with a series of follow-ups [5, 8, 16, 26, 27, 36, 41, 44, 48, 53, 56, 57], spectral methods have made significant progress in addressing the non-rigid shape matching problem, yielding state-of-the-art performance. However, because of the heavy dependence of Laplace-Beltrami operators, DFM can suffer notable performance drop when applied to point clouds without adaptation [9]. In fact, inspired by the success of DFM, several approaches [9, 29, 30] have been proposed to leverage intrinsic geometry information carried by meshes in the training of feature extractors tailored for non-structural point clouds. When it comes to pure point cloud matching, there is a line of works [15, 32, 64] leverage point cloud reconstruction as the proxy task to learn embeddings without correspondence labels. Since intrinsic information is not explicitly formulated in these methods, they can suffer from significant intrinsic deformations and often generalize poorly to unseen shapes.

2.2. Pre-trained Vision Model for Shape Analysis

Recently, Pre-trained Vision Models have become increasingly popular in due to their remarkable ability to understand data distributions from extensive image datasets. In the fields of shape analysis, [65] proposes an alternative to obtain superior 3D representations from 2D pre-trained models via Image-to-Point Masked Auto-encoders. [1] introduces a fully multi-stage method that exploits the exceptional reasoning capabilities of recent foundation models in language [49] and vision[35] to tackle difficult shape correspondence problems. In [46], before surface matching, the authors propose to use the features extracted from DINOv2 [50] of multi-view images of the shapes to perform co-alignment. In contrast to these approaches, which primarily utilize coarse patch features for sparse landmarks or semantic matching, our approach introduces an end-to-end method that aggregates pixel-level 2D features into point-wise 3D features.

2.3. Non-rigid Partial Shape Matching

While significant advancements have been made in full shape matching, there remains considerable room for improvement in estimating dense correspondences between shapes with partiality. Functional maps representation [5, 9, 54] has already been applied to partial shapes. However, both axiomatic and learning-based lines of work typically assume

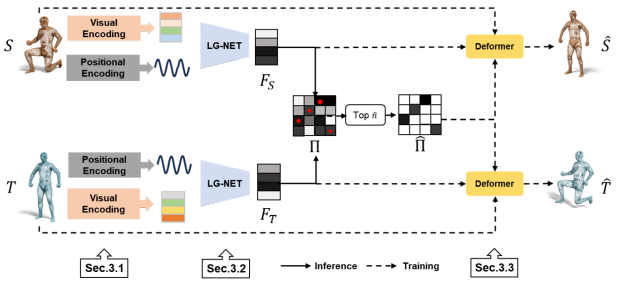


Figure 2. The schematic illustration of our pipeline.

the input to be a *connected mesh*, with the exception of [9], which relies on graph Laplacian construction [55] in its pre-processing. For partial point cloud matching, axiomatic registration approaches [3, 38, 62] assume the deformation of interest can be approximated by local, small-to-moderate, rigid deformations, therefore suffer from large intrinsic deformations. Simultaneously, there's a growing trend towards integrating deep learning techniques [6, 7, 25, 37]. However, these methods often focus on addressing the partial sequence point cloud registration problem.

3. Methodology

Fig. 2 shows the overall pipeline. We first introduce our visual encoding method in Sec.3.1. Then our global and local attention network will be discussed in Sec.3.2. The training losses are described in Sec.3.3.

3.1. Visual Encoding

Given a point cloud P consisting of N points, we denote the $i-$th point by $p_i = (x_i, y_i, z_i)$, our goal is to obtain a per-point feature carrying semantic information from pre-trained visual models. For more details, please refer to the Supp. Mat.

Depth aware projection: Following I2P-MAE [65], we project P on $xy-, yz-, xz-$plane to obtain three images. Taking $xy-$plane for example, we project point p_i onto pixel $(u_i, v_i) = (\lfloor \frac{x_i - x_{\min}}{\Delta} \times H \rfloor, \lfloor \frac{y_i - y_{\min}}{\Delta} \times W \rfloor)$, with pixel intensity $f(u_i, v_i) = sigmod(z_i)$. Here $\Delta = \max\{x_{\max} - x_{\min}, y_{\max} - y_{\min}\}$, and H, W are pre-determined image dimensions. For unprojected pixels, we simply set intensity to 0.

We note that the projected images often come with holes due to the discrete nature of point clouds, which are distinctive from the realistic training images used in DINO. To alleviate the discrepancy, we propose to 1) apply a 3×3 mean filter on the gray images and 2) assign pseudo color on the pixel values with the `PiYG` colormap in MATLAB. We now denote by $I_{\hat{z}}, I_{\hat{x}}, I_{\hat{y}}$ to resulting images, where $\hat{z}$ indicates projection onto $xy-$ plane (and similarly for $\hat{x}, \hat{y}$).

Lifting image features: Directly applying DINOv2 on the projected images results in coarse features of dimension $D \times D \times C$, where D is small (*e.g.,* 16). We further leverage FeatUp [20] to upsample DINOv2 features to match the dimension of the projected images

$$F^{img}_{\hat{z}} = \Theta(I_{\hat{z}}) \in \mathbb{R}^{H \times W \times C}, \tag{1}$$

where Θ is the per-pixel encoder of DinoV2-FeatUp [20] and C is the number of channels for each pixel. Via the one-to-one correspondences between p_i and (u_i, v_i), we obtain the point-wise feature of p_i via a simple pull-back:

$$f^i_{\hat{z}} = F_{\hat{z}}(u_i, v_i, :) \in \mathbb{R}^C. \tag{2}$$

We then have $F^{pt}_{\hat{z}} \in \mathbb{R}^{N \times C}$ by stacking $f^i_{\hat{z}}$ in order. We compute $F^{pt}_{\hat{x}}, F^{pt}_{\hat{y}}$ in the same manner. We emphasize that these computations are independent. In the end, we arrive at

$$F^{pt}(P) = [F^{pt}_{\hat{x}}, F^{pt}_{\hat{y}}, F^{pt}_{\hat{z}}] \in \mathbb{R}^{N \times 3C}. \tag{3}$$

The above procedure returns a set of per-point features for the input P, which essentially carry the semantic information extracted by the visual encoding.

3.2. Local and Global Attention Network

In this part, we describe our Local and Global Attention Network, which are depicted in Supp. Mat.

Input feature: In order to exploit both visual (image-based) and geometric (point-based) features, we perform early fusion at the input stage as follows:

$$F^{in}(P) = \text{LBR}(F^{pt}(P)) + \gamma(P), \tag{4}$$

where $\gamma(P) \in \mathbb{R}^{N \times 384}$ is the positional encoding [45] and LBR is a module proposed in PCT [23] for non-linearly converting $F^{pt}(P)$ into the same dimension of $\gamma(P)$.

Architecture design: We propose a dual-pathway architecture in parallel, comprising global attention [23] and local attention [61] blocks, which effectively draw matching cues from different hierarchical levels.

However, the receptive filed of local attention in [61] is pre-computed using point coordinates and fixed through training, which can be misleading in non-rigid shape matching. For instance, one's hand can be close to his head, but their intrinsic distance should always be large. Inspired by DGCNN [59], we lift the neighborhood search to the feature domain and keep updating it through learning.

In the end, we propose a fusion module consisting of LBR and a three-layer stacked N2P [61] attention, to merge features from both global and local paths, resulting in our output feature. We refer readers to the Supp. Mat. for more details.

3.3. Training Objectives and Matching Inference

In the following, we introduce our training losses, which consist of our novel deformation-based loss, arap loss, smoothness loss and geometrical similarity loss. As shown in Fig. 2,

our main model is a Siamese network. Given a pair of point clouds $\mathcal{S} \in \mathbb{R}^{N\times 3}, \mathcal{T} \in \mathbb{R}^{M\times 3}$, we compute C-dimension per-vertex features $F_{\mathcal{S}} \in \mathbb{R}^{N\times C}, F_{\mathcal{T}} \in \mathbb{R}^{M\times C}$ from the LG-Net (Sec. 3.2). We then estimate dense correspondences, $\Pi_{\mathcal{ST}} \in \mathbb{R}^{N\times M}$ and $\Pi_{\mathcal{TS}} \in \mathbb{R}^{N\times M}$, with respect to Euclidean distance [13] among rows of $F_{\mathcal{S}}$ and $F_{\mathcal{T}}$, followed by a softmax normalization. Then, we select the matches set with top $\hat{n} = 10$ matching scores and set the remaining elements to zero to get $\hat{\Pi}_{\mathcal{ST}}$ and $\hat{\Pi}_{\mathcal{TS}}$, which are used to define the following losses.

Deformation-based loss aims to deform one point cloud to the other using the above predicted $\hat{\Pi}_{\mathcal{ST}}$ and $\hat{\Pi}_{\mathcal{TS}}$ as shown in Fig. 3.

Taking the direction from $\mathcal{S}$ to $\mathcal{T}$ for example, we start by constructing deformation graph on $\mathcal{S}$. Following [22, 29], we perform farthest point sampling (FPS) on $\mathcal{S}$ to obtain $\lfloor N/2 \rfloor$ points, $\mathcal{S}_{\mathcal{D}} \in \mathbb{R}^{\lfloor N/2 \rfloor \times 3}$, as nodes of the deformation graph. In particular, we encode the above sampling process as a binary matrix $\Pi_{\mathcal{D}} \in \mathbb{R}^{\lfloor N/2 \rfloor \times N}$ such that $\Pi_{\mathcal{D}}(i,j) = 1$ if and only if the j-th point of $\mathcal{S}$ is sampled in the $i-$th round of FPS, and $\Pi_{\mathcal{D}}(i,j) = 0$ otherwise. It is therefore evident that $\mathcal{S}_{\mathcal{D}} = \Pi_{\mathcal{D}}\mathcal{S}$.

We then assign with each node in $\mathcal{S}_{\mathcal{D}}$ a rigid transformation parametrized as $\{\theta, \delta\}$, where θ is a 6-dim representation of rotation matrix [67] and δ is a 3-dim translation vector. Stacking all together, we arrive at $\mathbf{X} = \{\Theta, \Delta\}$, where $\Theta \in \mathbb{R}^{\lfloor N/2 \rfloor \times 6}, \Delta \in \mathbb{R}^{\lfloor N/2 \rfloor \times 3}$. Given $\mathbf{X}$, we can then propagate the transformations from nodes in $\mathcal{S}_{\mathcal{D}}$ to each point in $\mathcal{S}$ via a distance-based weighting scheme, we refer readers to the Supp. Mat. for more details. Finally, we denote by $\hat{\mathcal{S}}$ the deformed version of $\mathcal{S}$ with respect to $\mathbf{X}$.

$$\hat{\mathcal{S}} = \mathcal{DG}\left(\mathbf{X}, \mathcal{S}\right). \tag{5}$$

In prior works [3, 29, 62], $\mathbf{X}$ is often optimized jointly with correspondences in alternative iterations. In contrast, we propose to train a neural network to predict $\mathbf{X}$ in a single feedforward. Specifically, we first construct a dimension-preserving graph convolution network $\mathbf{G} : \mathbb{R}^{\cdot\times C} \rightarrow \mathbb{R}^{\cdot\times C}$ for gathering information from $F_{\mathcal{S}}, F_{\mathcal{T}}$, namely, the features learned from LG-Net. Then we use the following MLP to predict $\mathbf{X}$:

$$\mathbf{X} = \mathbf{MLP}(\mathcal{S}_{\mathcal{D}}, \Pi_{\mathcal{D}}\mathbf{G}(F_{\mathcal{S}}), \Pi_{\mathcal{D}}\hat{\Pi}_{\mathcal{ST}}\mathcal{T}, \Pi_{\mathcal{D}}\hat{\Pi}_{\mathcal{ST}}\mathbf{G}(F_{\mathcal{T}})). \tag{6}$$

In the other words, we first pull back the point cloud $\mathcal{T}$ and regarding feature $\mathbf{G}(F_{\mathcal{T}})$ to the source shape via $\hat{\Pi}_{\mathcal{ST}}$, then we perform the sampling regarding deformation nodes, $\Pi_{\mathcal{D}}$, on both spatial positions and latent features regarding both source and (pull-back) target shape. Finally, we predict $\mathbf{X}$ with all the above information.

Now we define the deformation loss regarding $\mathbf{X}$ as the chamfer distance between deformed $\mathcal{S}$ and $\mathcal{T}$:

$$\mathcal{L}_{\text{deform}}^{(\mathcal{S},\mathcal{T})}(\mathbf{X}) = CD(\mathcal{DG}(\mathbf{X}, \mathcal{S}), \mathcal{T}), \tag{7}$$

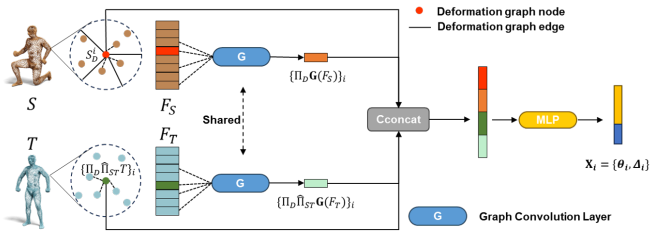


Figure 3. Illustration of our deformer, which predicts rigid transformation at each deformation graph node.

where $\mathbf{X}$ is learned via Eqn. 6.

ARAP loss: We adopt the classic As-Rigid-As-Possible (ARAP) regularization [22, 33] on $\mathbf{X}$. We postpone the exact form of $\mathcal{L}_{\text{arap}}^{(\mathcal{S},\mathcal{T})}$ to the Supp. Mat.

Smoothness loss: On the other hand, in order to encourage smoothness on the learned correspondences, we follow [32] to pose the following smoothness regularization:

$$\mathcal{L}_{\text{smooth}}^{(\mathcal{S},\mathcal{T})} = CD(\mathcal{T}, \hat{\Pi}_{\mathcal{ST}}\mathcal{T}) \tag{8}$$

Geometrical similarity: Since our predicted deformation graph comes from feature embedding $F_{\mathcal{S}}, F_{\mathcal{T}}$, we hope that the latter respects the local intrinsic geometry of the underlying surfaces. One high-level intuition is to enforce the Euclidean metric derived from the learned embeddings to approximate the underlying surface metric. Though this idea has been exploited in NIE [30], instead of minimizing the *global absolute residual* between the two metrics, we opt for maximizing the *local angular similarity* as follows.

We first adopt the heat method [14] to compute approximated geodesic matrix $\mathbf{M}_{\mathcal{S}}$ on $\mathcal{S}$. Then, treating $F_{\mathcal{S}} \in \mathbb{R}^{N\times C}$ as a $C-$dim embedding of $\mathcal{S}$, for each $x_i \in \mathcal{S}$, we compute its nearest neighbors in the embedded space and obtain $\text{NN}(i) = \{j_1, j_2, \cdots, j_k\}$ be the set of ordered indices. The regarding ascending list of distances is denoted by $d_{\mathcal{S}}^i \in \mathbb{R}^k$. On the other hand, we retrieve approximated geodesic distances from $\mathbf{M}_{\mathcal{S}}$ as $m_{\mathcal{S}}^i \in \mathbb{R}^k$, where $m_{\mathcal{S}}^i(t) = \mathbf{M}_{\mathcal{S}}(i, j_t), t = 1, 2, \cdots, k.$. The geometrical similarity loss can be denoted as:

$$\mathcal{L}_{\text{geo}}^{(\mathcal{S})} = \frac{1}{N}\sum_{i=1}^{N}(1 - \frac{d_{\mathcal{S}}^i \cdot m_{\mathcal{S}}^i}{\|d_{\mathcal{S}}^i\|\|m_{\mathcal{S}}^i\|}). \tag{9}$$

It is worth noting that the overhead of computing $\mathbf{M}_{\mathcal{S}}$ is only needed during training.

To summarize, we define the loss regarding the direction from $\mathcal{S}$ to $\mathcal{T}$ as

$$\begin{aligned}\mathcal{L}_{\text{total}}^{(\mathcal{S},\mathcal{T})} =& \lambda_{\text{deform}}\mathcal{L}_{\text{deform}}^{(\mathcal{S},\mathcal{T})} + \lambda_{\text{arap}}\mathcal{L}_{\text{arap}}^{(\mathcal{S},\mathcal{T})} \\ &+ \lambda_{\text{smooth}}\mathcal{L}_{\text{smooth}}^{(\mathcal{S},\mathcal{T})} + \lambda_{\text{geo}}\mathcal{L}_{\text{geo}}^{(\mathcal{S})}.\end{aligned} \tag{10}$$

Note that we perform training in a pairwise manner, we formulate all the above loss terms in both directions, that is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{total}}^{(\mathcal{S},\mathcal{T})} + \mathcal{L}_{\text{total}}^{(\mathcal{T},\mathcal{S})}.$$

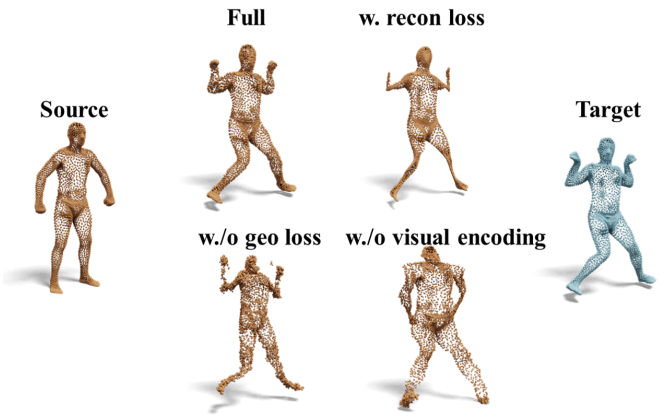


Figure 4. Ablation study on our training losses.

Partial matching loss: In the above, we entail the losses for training full-to-full non-rigid point cloud matching. Remarkably, our formulation can be easily extended to the challenging scenario of partial-to-full matching by modifying $\mathcal{L}_{\text{deform}}$ to a unilateral loss, i.e. only the partial-to-full chamfer distance is considered.

Inference: At inference time, we choose the nearest latent cross-neighborhood of $x_i \in \mathcal{S}$ to be its corresponding point by KNN [13], thus get the shape matching result between point cloud $\mathcal{S}$ and $\mathcal{T}$.

3.4. Remarks

1. While our exploitation of pre-trained vision models bears certain similarity to I2P-MAE [65], we highlight the key difference – I2P-MAE enforces point-based model to align with the vision model, as the features of vision model are also used as reconstruction target; on the other hand, our framework only use vision model for pre-encoding, the point model is essentially guided by geometry-based losses.
2. As mentioned in Sec. 1, the reconstruction-based proxy task used in [15, 32] can lead to mode collapse. We verify this in Fig. 4, where "w. recon loss" means incorporating the above proxy task in our framework. Evidently, our deformation-based approach is free of collapsing. We also qualitatively justify our other designs in Fig. 4.

4. Experiments

Dataset: We evaluate our method with several state-of-the-art methods on an array of different categories as follows, we defer the detailed descriptions of each benchmark to the Supp. Mat. for completeness.

1. Human: We consider the well-known benchmarks, including near-isometric ones – **SCAPE_r, FAUST_r, SHREC'19_r**, and more heterogenous ones – **DT4D-H, SHREC'07-H**. Regarding partial shape matching, we include the well-known **SHREC'16** benchmark as well as partial-view datasets we construct based on **SCAPE_r** and **FAUST_r**. Besides, we further consider large-scale dataset – **SURREAL** for training in the Supp. Mat..
2. Animal: We utilize **TOSCA** dataset, which comprises various animal species. We also consider the training of large-scale datasets from **SMAL** in the Supp. Mat..
3. Garment: We consider the **GarmCap** [40] dataset includes four different garments, each is presented by a sequence of point clouds in various poses.
4. Medical: We consider two datasets, **Spleen** and **Pancreas**, proposed in [2]. Both datasets are of limited data samples with considerable variabilities, which is typical in medical data processing.
5. Real-scans: The **Panoptic** dataset [31] consists of partial point clouds derived from multi-view RGB-D images. We randomly select a subset of these views to recover partial point clouds.

Baseline: We compare our method with a set of competitive baselines, including learning-based methods that can both train and test on point clouds – CorrNet3D[64], RMA-Net [19], SyNoRiM [25], DiffFMaps [43], DPC [32], SE-ORNet[15], HSTR [24] ; methods required mesh for geometry-based training but inference with point clouds – 3D-CODED [21], TransMatch [58], NIE[30], SSMSM[9], ConsistFMaps[8], DPFM[5], HCLV2S[28]; non-learning-based axiomatic methods used for registration – NDP [38] , AMM [62], PointSetReg [66]. In Sec. 4.2, we further consider classical baselines especially tailored for medical dataset – CAFE [11], ISR [10], Point2SSM [2]. Methods are marked according to if they need correspondence label ([S]) or not ([U]), as well as axiomatic ([A]).

Remark: In particular, we consider DPC [32] and SE-ORNet [15] as the primary competing methods because they can be trained purely on point clouds and without any correspondence annotation. We emphasize that meshing is generally non-trivial in real-world data (due to, *e.g.,* partiality and noise). The results of methods that require meshes during training are also included as *reference*.

Evaluation metric: Though we focus on the matching of point clouds, we primarily employ the widely-accepted geodesic error normalized by the square root of the total area of the mesh, to evaluate the performance of all methods.

Hyper-parameters: In Eqn. 10, the hyper-parameters $\lambda_{\text{deform}}, \lambda_{\text{arap}}, \lambda_{\text{smooth}}, \lambda_{\text{geo}}$ are uniformly set to 0.05, 0.005, 0.5, and 0.02 respectively. Model training utilizes the AdamW [42] optimizer with $\beta = (0.9, 0.99)$, learning rate of 2e-3, and batch size of 2. We provide more details of hyper-parameters in the Supp. Mat.

4.1. Standard non-rigid matching benchmarks

In the following, we denote by A/B the scheme of training on dataset A and test on B.

Near-isometric benchmarks: As illustrated in Tab. 1, our method consistently outperforms other purely point-

Table 1. Quantitative results on SCAPE_r (S_r), FAUST_r (F_r), SHREC'19_r (S19_r), DT4D-H and SHREC07-H (S07-H) in terms of mean geodesic errors ($\times 100$). The **best** results from the pure point cloud methods in each column are highlighted.

Method	Train / Test	S_r					F_r				
		S_r	F_r	S19_r	DT4D-H	S07-H	F_r	S_r	S19_r	DT4D-H	S07-H
3D-CODED[S] [21]	Mesh Required	31.0	33.0	\	\	\	2.5	31.0	\	\	\
TransMatch[S] [58]		18.6	18.3	38.8	25.3	31.2	2.7	33.6	21.0	26.7	25.3
NIE[U] [30]		11.0	8.7	15.6	12.1	13.4	5.5	15.0	15.1	13.3	15.3
SSMSM[U] [9]		4.1	8.5	7.3	8.0	37.7	2.4	11.0	9.0	11.8	42.2
NDP[A] [38]	PCD Only	16.2	\	\	\	\	20.4	\	\	\	\
AMM[A] [62]		13.1	\	\	\	\	14.2	\	\	\	\
PointSetReg[A] [66]		17.1	\	\	\	\	18.3	\	\	\	\
DiffFMaps[S] [43]		12.0	12.0	17.6	15.9	15.4	3.6	19.0	16.4	18.5	16.8
SyNoRiM[S] [18]		9.5	24.6	\	\	\	7.9	21.9	\	\	\
CorrNet3D[U] [64]		58.0	63.0	\	\	\	63.0	58.0	\	\	\
DPC[U] [32]		17.3	11.2	28.7	21.7	17.1	11.1	17.5	31.0	13.8	18.1
SE-ORNet[U] [15]		24.6	22.8	23.6	27.7	12.2	20.3	18.9	23.0	12.2	20.9
Ours[U]		**6.2**	**5.1**	**7.2**	**6.9**	**7.7**	**5.4**	**10.4**	**9.3**	**8.1**	**8.2**

based methods in all settings. Especially, our method achieves a promising performance improvement of over **59%** compared to the previous SOTA approaches (**7.2** vs. 23.6; **9.3** vs. 23.0) in SCAPE_r/SHREC19'_r case and FAUST_r/SHREC19'_r case. Many previous methods performs well in the standard seen datasets but generalizes poorly to unseen shapes. Remarkably, our method outperforms all baselines, including the STOA method leveraging meshes during training, SSMSM [9], in SCAPE_r/FAUST_r and FAUST_r/SCAPE_r (**5.1** vs. 8.5, **10.4** vs. 11.0).

Generalization to non-isometric benchmarks: We perform stress test on challenging non-isometric datasets including SHREC'07-H and DT4D-H. Our method achieves the best performance among *all* methods as shown in Tab. 1, which indicates the excellent generalization ability to some unseen challenging cases. Especially, SHREC'07-H dataset comprises 20 heterogeneous human shapes with vertex numbers ranging from 3000 to 15000 and includes topological noise. Our method achieves a performance improvement of over **42%** compared to the second best approach (**7.7** vs. 13.4; **8.2** vs. 15.3).

Partial matching benchmarks: As shown in Sec. 3.3, our framework can be easily adapted for unsupervised partial-to-full point matching. We evaluate our method in two types of partial shape matching, including the challenging SHREC'16[12] Cuts and Holes benchmark and two partial-view benchmarks built on SCAPE_r and FAUST_r datasets by ourselves, where we employ raycasting from the center of each face of a regular dodecahedron to observe the shapes, resulting 12 partial view point clouds.

As illustrated in Tab. 2, our method outperforms the recent unsupervised method SSMSM [9] in 3 out of 4 test cases, which requires meshes for training. Fig. 5a further shows qualitatively that our framework outperforms the competing methods including DPFM [5], which is based on mesh input as well. Moreover, recent axiomatic method PointSetReg [66] also struggles with partial cases.

Regarding purely point cloud-based baselines, we modify the reconstruction loss of [15, 32] to unilateral loss to adapt partiality in the same way. In the end, we achieve the SOTA compared to them, exhibiting a significant over **54%** superiority in partial view matching (**6.2** vs. 13.6; **5.3** vs. 13.9) and over **48%** superiority in cuts/holes (shown in Fig. 5c) setting (**16.9** vs. 32.9; **13.0** vs. 27.6).

Table 2. Quantitative results on partial cases including SCAPE-PV (S-PV), FAUST-PV(F-PV) and SHREC'16 (S16) in terms of mean geodesic errors ($\times 100$). * indicates its original checkpoint using SURREAL190K. The **best** results from the pure point cloud methods in each column are highlighted.

Method	Train / Test	S-PV		S16-CUTS	S16-HOLES
		S-PV	F-PV	CUTS	HOLES
ConsistFMaps unsup[U] [8]	Mesh Required	\	\	26.6	27.0
DPFM unsup[U] [5]		11.5	15.2	20.9	22.8
HCLV2S*[S] [28]		8.7	5.3	\	\
SSMSM[U] [9]		8.8	8.0	12.2	16.7
PointSetReg[A] [66]	PCD Only	15.7	\	36.1	24.0
DPC[U] [32]		13.6	14.5	32.9	32.5
SE-ORNet[U] [15]		15.4	13.9	40.5	27.6
Ours[U]		**6.2**	**5.3**	**16.9**	**13.0**

Table 3. Generalization performance of the checkpoint trained on sampled point cloud with fixed 1024 points of SHREC'19. We test this checkpoint on the more dense original point cloud. The **best** is highlighted. See more details of this evaluation in the Supp. Mat.

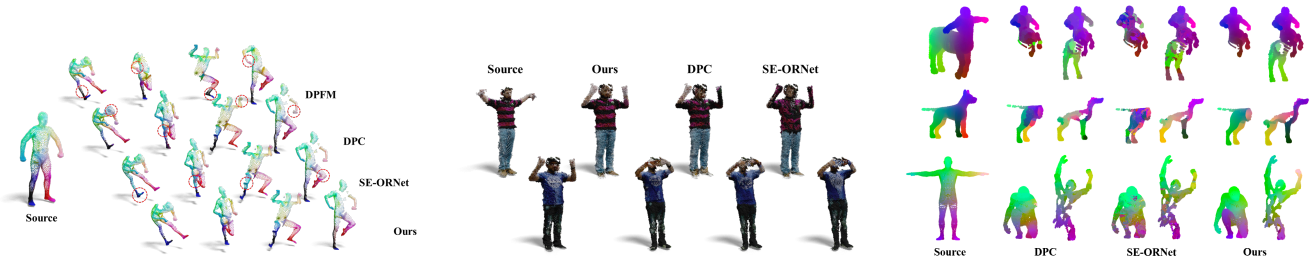
Method	SHREC'19 (1024)	SHREC'19 (Ori.)
DPC[U] [32]	5.6	6.1 (+8.93%)
SE-ORNet[U] [15]	5.1	5.9 (+15.69%)
Ours [U]	**4.31**	**4.33** (+0.46%)

Finally, we attribute the above success to that the per-point features aggregated from pre-trained vision model which carries rich semantic information, helps to identify correspondence at the coarse level. In addition to that, our final features are further boosted by the geometric losses, leading to strong performance.

4.2. Realworld Applications

In this part, we showcase the utility of our framework under more practical settings:

Learning from synthetic partial scans: Since in practice, raw point clouds are often acquired as partial-view scans, we synthesize a set of 516 partial point clouds with shapes



(a) Partial view of SCPAE. (b) Noisy partial real scans. (c) SHREC'16 – CUT and HOLE.

Figure 5. Qualitative results of SCAPE-PV, noisy real scans, and SHREC16.

Table 4. Generalization testing on partial, isometric, and non-isometric full shape in terms of mean geodesic errors (×100).

Method	SHREC19-PV	SHREC19_r	DT4D-H	SHREC07-H
DPC [32]	32.77	32.65	23.71	18.94
SE-ORNET [15]	29.18	28.06	18.61	29.99
Ours	**9.40**	**8.97**	**8.61**	**9.50**

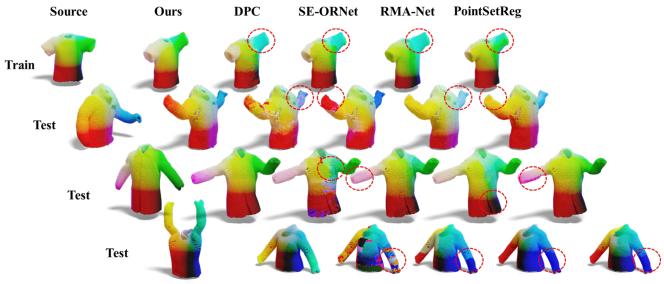


Figure 6. Qualitative results of GarmCap garment dataset.

from the SHREC'19 dataset. For simplicity, we add one template full point cloud as the reference and train DV-Matcher, DPC and SE-ORNet on the 516 pairs (*i.e.,* full vs. partial). Note here we do not use any correspondence annotation. Then we evaluate the trained models on 1) test set of the above synthetic partial data, including pairs formed by a full and a partial point cloud (both randomly sampled from SHREC'19); 2) test set of SHREC'19 dataset, which are all full point clouds; 3) test set of DT4D-H and SHREC'07-H. As shown in Tab. 4, our framework delivers consistently the best scores in all test cases, outperforming the baselines by a significant margin (see Fig. 1 as well).

Matching real scans: As shown in Fig. 5b, we transfer texture from the source shape (left-most) to target via maps from ours, DPC [32], SE-ORNet [15]. Our method demonstrates smoother texture transfer compared to baselines (see particularly facial details and strips in the T-shirt).

Statistical shape models (SSM) for medical data: Following Point2SSM [2], we evaluate our method on the anatomical SSM tasks. We stick to the regarding experimental setting and report our score in the spleen subset. As shown in Tab. 5, our method outperforms the second best over **29%** relative error reduction (**2.3** vs. 3.4; **1.9** vs. 2.7).

Garment dataset: We in particular choose one of the four sequences of garment dataset [40], T-shirt, to train

Table 5. Statistical shape analysis on medical dataset in terms of chamfer distance (CD) and earth mover's distance (EMD). The **best** is highlighted.

Dataset	**Spleen**		**Pancreas**	
Point Accuracy Metrics (mm)	CD	EMD	CD	EMD
CPAE[U] [11]	61.3	4.0	18.8	2.6
ISR[U] [10]	17.6	2.9	7.4	1.8
DPC[U] [32]	10.6	2.1	6.1	1.7
Point2SSM[U] [2]	3.4	1.5	2.7	1.4
Ours[U]	**2.3**	**1.1**	**1.9**	**1.0**

DV-Matcher and baselines. Then we directly evaluate on all test sets. As shown in Fig. 6, our method achieves the best accuracy as well as remarkable generalization performance, superior to the baselines, whether recent non-learning-based axiomatic methods [66] or learning-based methods [32], [15], [19]. We defer the quantitative results in the Supp. Mat.

4.3. Robustness and ablation analysis:

Robustness analysis: Reconstruction-based methods [15, 24, 32] typically perform down-sampling to $n = 1024$ points for *both* training and testing. However, point clouds scanned in reality typically consist of tens of thousands of points, which is much denser. To evaluate the robustness regarding point density, we use the checkpoint trained on down-sampled data released by the regarding authors to evaluate performance in both down-sampled test data (1024 points) and original test data ($\sim$ 5000 points). As shown in Tab. 3, DPC and SE-ORNet [15, 32] both experience a degradation more than **8%**. On the other hand, our method only yields a 0.46% drop and achieves the best performance in both cases. Beyond the quantitative results, we also report qualitatively the generalization performance on TOSCA benchmark following the same setting as above, see Fig. 7 for more details.

We attribute the above robustness to our introduction of pre-trained vision model in point feature learning, which effectively compensates for the discrepancy of low-resolution geometry. Besides, our framework is trained on 3D set of interest, which is more robust to the errors made by 2D pre-trained models. To further support this, we consider the tasks in SCAPE_r and spleen dataset. We compare the results using only visual feature aggregation (Eqn. 3) and our full model, and achieve 19.6 vs. **6.2** and 15.2 vs. **2.3** –

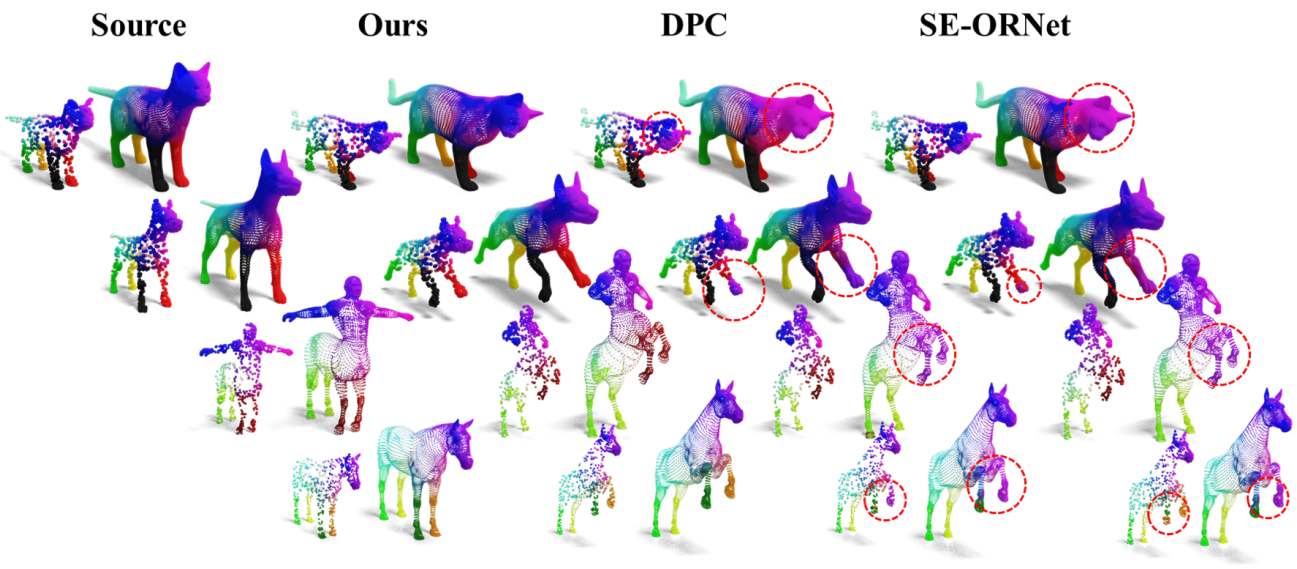


Figure 7. Qualitative results of TOSCA. Our method demonstrates enhanced generalization capabilities when transitioning from sparse point clouds in training to dense point clouds in testing.

Table 6. Mean geodesic errors ($\times 100$) on different ablated settings, the models are all trained on SCAPE_r and test on SCAPE_r.

Ours-Recon	w/o visual enc.	w/o $\mathcal{L}_{\text{deform}}$	w/o $\mathcal{L}_{\text{geo}}$	w/o $\mathcal{L}_{\text{smooth}}$	w/o LG-Net
7.6	15.1	11.2	12.8	6.9	17.3
w/o FeatUp	w/o LA-Net	w/o GA-Net	w/o PE	w/o Fusion	Ours-Full
7.8	8.0	8.1	7.1	7.9	6.2

Table 7. Average time cost for each shape regarding SCAPE_r.

Method	PointSetReg	DFR	DPC	SE-ORNET	Ours
Time cost(s)	62.1[CPU]	13.4	1.41	0.85	3.22

our scheme shows significant gain in both cases. Especially, the relative improvement is more significant in the spleen dataset, which is less familiar to 2D pre-trained models.

Finally, we highlight that we have also performed robustness evaluation regarding noisy data and rotation perturbations in Supp. Mat.

Ablation study: We first justify our overall design in Tab. 6, where we sequentially remove each building block from our pipeline and train/test model on SCAPE_r, including visual encoding (visual enc.) for pre-trained semantic priors, positional encoding (PE) for fine-grained positional information of each point, local and global attention network (LG-Net) for feature refining, etc. Specifically, the deformation-based loss we proposed plays a crucial role in efficient registration as illustrated in Fig.4, as [with recon loss] would cause shape collapse, which delivers the difference between the methods simply using reconstruction loss like DPC [32], SE-ORNET [15], etc. Besides, our geometric loss ensures the preservation of local isometry as the [w/o geo loss] indicates. Furthermore, [w/o visual enc.] shows that our method is in synergy with the coarse-grained semantic information from pre-trained vision models. We have also assessed the necessity of each specific module within the LG-Net network in Tab. 6, i.e., local attention blocks branch (LA-Net), global attention blocks branch (GA-Net), fusion module (Fusion) after the dual-pathway network.

5. Conclusion and Limitation

In this paper, we propose DV-Matcher for non-rigid point cloud matching that can be trained purely on point clouds without any correspondence annotation and extends naturally to partial-to-full matching. By incorporating semantic features from pre-trained vision models and a deformation-based proxy task, DV-Matcher demonstrates strong matching performance, promising generalizability and various robustness regarding partiality, point density, input orientation. Last but not least, DV-Matcher also performs well in real-world data, including 3D medical scans, textured garment scans, and noisy dynamic human scans.

Limitation & Future Work While demonstrating superior performance and robustness over the baselines in a wide range of tests. We recognize the following limitations, which naturally give rise to future directions: 1) As shown in Tab. 7, our method takes on average 3.22 seconds to match input of around 5000 points, which is far from real-time; 2) Though our method has shown robustness regarding certain rotation perturbations on inputs (see the Supp. Mat. for more details), it does not guarantee robustness in general orientation. In the future, we plan to further explore the ability of pre-trained visual models to address this limitation; 3) Though our method performs the best in the stress test, it is still difficult to directly train on all raw scans, which requires overlapping region prediction. We believe it would be interesting to explore for higher practical impact.

Acknowledgment This work was supported by the National Natural Science Foundation of China under contract No. 62171256.

References

[1] Ahmed Abdelreheem, Abdelrahman Eldesokey, Maks Ovsjanikov, and Peter Wonka. Zero-shot 3d shape correspondence. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 2

[2] Jadie Adams and Shireen Elhabian. Point2ssm: Learning morphological variations of anatomies from point cloud. *arXiv preprint arXiv:2305.14486*, 2023. 5, 7

[3] Brian Amberg, Sami Romdhani, and Thomas Vetter. Optimal step nonrigid icp algorithms for surface registration. 2007. 1, 3, 4

[4] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. SCAPE: Shape Completion and Animation of People. 2005. 1

[5] Souhaib Attaiki, Gautam Pai, and Maks Ovsjanikov. Dpfm: Deep partial functional maps. In *2021 International Conference on 3D Vision (3DV)*, pages 175–185. IEEE, 2021. 2, 5, 6

[6] Aljaz Bozic, Pablo Palafox, Michael Zollhöfer, Angela Dai, Justus Thies, and Matthias Nießner. Neural non-rigid tracking. In *NeurIPS*, pages 18727–18737, 2020. 3

[7] Aljaz Bozic, Michael Zollhofer, Christian Theobalt, and Matthias Nießner. Deepdeform: Learning non-rigid rgb-d reconstruction with semi-supervised data. In *CVPR*, pages 7002–7012, 2020. 3

[8] Dongliang Cao and Florian Bernard. Unsupervised deep multi-shape matching. In *ECCV*, 2022. 2, 5, 6

[9] Dongliang Cao and Florian Bernard. Self-supervised learning for multimodal non-rigid shape matching. In *CVPR*, 2023. 1, 2, 3, 5, 6

[10] Nenglun Chen, Lingjie Liu, Zhiming Cui, Runnan Chen, Duygu Ceylan, Changhe Tu, and Wenping Wang. Unsupervised learning of intrinsic structural representation points. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9121–9130, 2020. 5, 7

[11] An-Chieh Cheng, Xueting Li, Min Sun, Ming-Hsuan Yang, and Sifei Liu. Learning 3d dense correspondence via canonical point autoencoder. *Advances in Neural Information Processing Systems*, 34:6608–6620, 2021. 5, 7

[12] Luca Cosmo, Emanuele Rodola, Michael M Bronstein, Andrea Torsello, Daniel Cremers, Y Sahillioğlu, et al. Shrec'16: Partial matching of deformable shapes. In *Eurographics Workshop on 3D Object Retrieval, EG 3DOR*, pages 61–67. Eurographics Association, 2016. 6

[13] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13 (1):21–27, 1967. 4, 5

[14] Keenan Crane, Clarisse Weischedel, and Max Wardetzky. The heat method for distance computation. *Commun. ACM*, 60 (11):90–99, 2017. 4

[15] Jiacheng Deng, Chuxin Wang, Jiahao Lu, Jianfeng He, Tianzhu Zhang, Jiyang Yu, and Zhe Zhang. Se-ornet: Self-ensembling orientation-aware network for unsupervised point cloud shape correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5364–5373, 2023. 2, 5, 6, 7, 8

[16] Nicolas Donati, Etienne Corman, and Maks Ovsjanikov. Deep orientation-aware functional maps: Tackling symmetry issues in shape matching. In *CVPR*, pages 742–751, 2022. 2

[17] Niladri Shekhar Dutt, Sanjeev Muralikrishnan, and Niloy J Mitra. Diffusion 3d features (diff3f): Decorating untextured shapes with distilled semantic features. *arXiv preprint arXiv:2311.17024*, 2023. 2

[18] Jiahui Huang et. al. Multiway non-rigid point cloud registration via learned functional map synchronization, 2022. 6

[19] Wanquan Feng, Juyong Zhang, Hongrui Cai, Haofei Xu, Junhui Hou, and Hujun Bao. Recurrent multi-view alignment network for unsupervised surface registration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 5, 7

[20] Stephanie Fu, Mark Hamilton, Laura E. Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. Featup: A model-agnostic framework for features at any resolution. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[21] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 3d-coded: 3d correspondences by deep deformation. In *ECCV*, 2018. 2, 5, 6

[22] Chen Guo, Xu Chen, Jie Song, and Otmar Hilliges. Human performance capture from monocular video in the wild. In *3DV*, 2021. 4

[23] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7:187–199, 2021. 3

[24] Jianfeng He, Jiacheng Deng, Tianzhu Zhang, Zhe Zhang, and Yongdong Zhang. Hierarchical shape-consistent transformer for unsupervised point cloud shape correspondence. *IEEE Transactions on Image Processing*, 2023. 5, 7

[25] Jiahui Huang, Tolga Birdal, Zan Gojcic, Leonidas J. Guibas, and Shi-Min Hu. Multiway Non-rigid Point Cloud Registration via Learned Functional Map Synchronization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2022. 3, 5

[26] Ruqi Huang and Maks Ovsjanikov. Adjoint map representation for shape analysis and matching. In *Computer Graphics Forum*, pages 151–163. Wiley Online Library, 2017. 2

[27] Ruqi Huang, Jing Ren, Peter Wonka, and Maks Ovsjanikov. Consistent zoomout: Efficient spectral map synchronization. In *Computer Graphics Forum*, pages 265–278. Wiley Online Library, 2020. 2

[28] Xiangru Huang, Haitao Yang, Etienne Vouga, and Qixing Huang. Dense correspondences between human bodies via learning transformation synchronization on graphs. In *NeurIPS*, 2020. 2, 5, 6

[29] Puhua Jiang, Mingze Sun, and Ruqi Huang. Non-rigid shape registration via deep functional maps prior. In *NeurIPS*, 2023. 1, 2, 4

[30] Puhua Jiang, Mingze Sun, and Ruqi Huang. Neural intrinsic embedding for non-rigid point matching. In *CVPR*, 2023. 1, 2, 4, 5, 6

[31] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser

Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3334–3342, 2015. 5

[32] Itai Lang, Dvir Ginzburg, Shai Avidan, and Dan Raviv. Dpc: Unsupervised deep point correspondence via cross and self construction. In *2021 International Conference on 3D Vision (3DV)*, pages 1442–1451. IEEE, 2021. 2, 4, 5, 6, 7, 8

[33] Zohar Levi and Craig Gotsman. Smooth rotation enhanced as-rigid-as-possible mesh animation. *IEEE transactions on visualization and computer graphics*, 21(2):264–277, 2014. 4

[34] Hao Li, Robert W Sumner, and Mark Pauly. Global correspondence optimization for non-rigid registration of depth scans. In *Computer graphics forum*, pages 1421–1430. Wiley Online Library, 2008. 1

[35] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 2

[36] Lei Li, Nicolas Donati, and Maks Ovsjanikov. Learning multi-resolution functional maps with spectral attention for robust shape matching. In *NeurIPS*, 2022. 2

[37] Yang Li and Tatsuya Harada. Lepard: Learning partial point cloud matching in rigid and deformable scenes. In *CVPR*, 2022. 3

[38] Yang Li and Tatsuya Harada. Non-rigid point cloud registration with neural deformation pyramid. *Advances in Neural Information Processing Systems*, 35:27757–27768, 2022. 3, 5, 6

[39] Miao Liao, Qing Zhang, Huamin Wang, Ruigang Yang, and Minglun Gong. Modeling deformable objects from a single depth camera. In *2009 IEEE 12th International Conference on Computer Vision*, pages 167–174. IEEE, 2009. 1

[40] Siyou Lin, Boyao Zhou, Zerong Zheng, Hongwen Zhang, and Yebin Liu. Leveraging intrinsic properties for non-rigid garment alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14485–14496, 2023. 5, 7

[41] Or Litany, Tal Remez, Emanuele Rodolà, Alexander M. Bronstein, and Michael M. Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. In *ICCV*, 2017. 2

[42] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[43] Riccardo Marin, Marie-Julie Rakotosaona, Simone Melzi, and Maks Ovsjanikov. Correspondence learning via linearly-invariant embedding. *Advances in Neural Information Processing Systems*, 33:1608–1620, 2020. 1, 5, 6

[44] Simone Melzi, Jing Ren, Emanuele Rodolà, Peter Wonka, and Maks Ovsjanikov. Zoomout: Spectral upsampling for efficient shape correspondence. *Proc. SIGGRAPH Asia*, 2019. 2

[45] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3

[46] Luca Morreale, Noam Aigerman, Vladimir G. Kim, and Niloy J. Mitra. Semantic neural surface maps. In *Eurographics*, 2024. 2

[47] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136, 2011. 1

[48] Dorian Nogneng and Maks Ovsjanikov. Informative descriptor preservation via commutativity for shape matching. *Computer Graphics Forum*, 36(2):259–267, 2017. 2

[49] OpenAI. Gpt-3.5 language model. https://www.openai.com/research/gpt-3, 2021. Accessed: May 21, 2023. 2

[50] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 2

[51] 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):30, 2012. 1, 2

[52] Gianluca Paravati, Fabrizio Lamberti, Valentina Gatteschi, Claudio Demartini, and Paolo Montuschi. Point cloud-based automatic assessment of 3d computer animation courseworks. *IEEE Transactions on Learning Technologies*, 10(4):532–543, 2016. 1

[53] Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. Continuous and orientation-preserving correspondences via functional maps. *ACM Trans. Graph.*, 37(6):248:1–248:16, 2018. 2

[54] Emanuele Rodolà, Luca Cosmo, Michael M Bronstein, Andrea Torsello, and Daniel Cremers. Partial Functional Correspondence. In *Computer Graphics Forum*, 2016. 2

[55] Nicholas Sharp and Keenan Crane. A laplacian for nonmanifold triangle meshes. *Computer Graphics Forum*, 2020. 3

[56] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics*, 2022. 2

[57] Mingze Sun, Shiwei Mao, Puhua Jiang, Maks Ovsjanikov, and Ruqi Huang. Spatially and spectrally consistent deep functional maps. In *ICCV*, 2023. 2

[58] Giovanni Trappolini, Luca Cosmo, Luca Moschella, Riccardo Marin, Simone Melzi, and Emanuele Rodolà. Shape registration in the time of transformers. *Advances in Neural Information Processing Systems*, 34:5731–5744, 2021. 2, 5, 6

[59] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019. 3

[60] Thomas Wimmer, Peter Wonka, and Maks Ovsjanikov. Back to 3d: Few-shot 3d keypoint detection with back-projected

2d features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 2

[61] Chengzhi Wu, Junwei Zheng, Julius Pfrommer, and Jürgen Beyerer. Attention-based point cloud edge sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5333–5343, 2023. 3

[62] Yuxin Yao, Bailin Deng, Weiwei Xu, and Juyong Zhang. Fast and robust non-rigid registration using accelerated majorization-minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 3, 4, 5, 6

[63] Tao Yu, Zerong Zheng, Kaiwen Guo, Jianhui Zhao, Qionghai Dai, Hao Li, Gerard Pons-Moll, and Yebin Liu. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. In *CVPR*, 2018. 1

[64] Yiming Zeng, Yue Qian, Zhiyu Zhu, Junhui Hou, Hui Yuan, and Ying He. Corrnet3d: Unsupervised end-to-end learning of dense correspondence for 3d point clouds. In *CVPR*, pages 6052–6061, 2021. 2, 5, 6

[65] Renrui Zhang, Liuhui Wang, Yu Qiao, Peng Gao, and Hongsheng Li. Learning 3d representations from 2d pre-trained models via image-to-point masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21769–21780, 2023. 2, 3, 5

[66] Mingyang Zhao, Jingen Jiang, Lei Ma, Shiqing Xin, Gaofeng Meng, and Dong-Ming Yan. Correspondence-free non-rigid point set registration using unsupervised clustering analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21199–21208, 2024. 1, 5, 6, 7

[67] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5745–5753, 2019. 4