Unsupervised Template-assisted Point Cloud Shape Correspondence Network

Jiacheng Deng[1], Jiahao Lu[1], Tianzhu Zhang[1,2,†]
[1]University of Science and Technology of China, [2]Deep Space Exploration Lab
{dengjc, lujiahao }@mail.ustc.edu.cn, tzzhang@ustc.edu.cn

Abstract

Unsupervised point cloud shape correspondence aims to establish point-wise correspondences between source and target point clouds. Existing methods obtain correspondences directly by computing point-wise feature similarity between point clouds. However, non-rigid objects possess strong deformability and unusual shapes, making it a longstanding challenge to directly establish correspondences between point clouds with unconventional shapes. To address this challenge, we propose an unsupervised Template-Assisted point cloud shape correspondence Network, termed TANet, including a template generation module and a template assistance module. The proposed TANet enjoys several merits. Firstly, the template generation module establishes a set of learnable templates with explicit structures. Secondly, we introduce a template assistance module that extensively leverages the generated templates to establish more accurate shape correspondences from multiple perspectives. Extensive experiments on four human and animal datasets demonstrate that TANet achieves favorable performance against state-of-the-art methods.

1. Introduction

Point cloud shape correspondence is a challenging task that identifies and densely matches the source and target point clouds with deformable 3D shapes. The task has significant implications for various industries, including augmented reality [1], gaming [15], and robotics [21, 32]. However, the unrestricted mobility of humans and animals and their unusual postures have made direct correspondence between unconventional shapes a longstanding challenge in point cloud shape correspondence.

The shape correspondence problem has been thoroughly investigated for 3D mesh data. However, such spectral-based methods [2, 14, 19, 27, 35] are mainly limited by the time-consuming pre-processing step and the reliance on mesh vertice connectivity information, frequently absent in point cloud scenarios. Point-based methods [7, 8, 11, 13, 16, 24, 42] are directly applied to the raw point cloud data, relying solely on point coordinates without connectivity information. To mitigate the resource-intensive labeling in fully supervised methods [8, 11, 24], the unsupervised point cloud shape correspondence task garners significant attention and yields competitive results [7, 13, 16, 42]. DPC [16] designs construction losses to reduce outlier matches. HSTR [13] incorporates whitening losses to mitigate the impact of shape discrepancies. SE-ORNet [7] introduces an orientation estimation module to alleviate symmetry issues. The shapes of humans and animals are highly diverse and complex. For instance, the SURREAL dataset [11] comprises 230K human body shapes, while the SMAL dataset [45] contains 10K animal shapes, including numerous unconventional forms. Establishing accurate correspondences between two unconventional point cloud shapes remains a challenging problem in shape correspondence. Current methods [7, 13, 16, 42] rely on computing

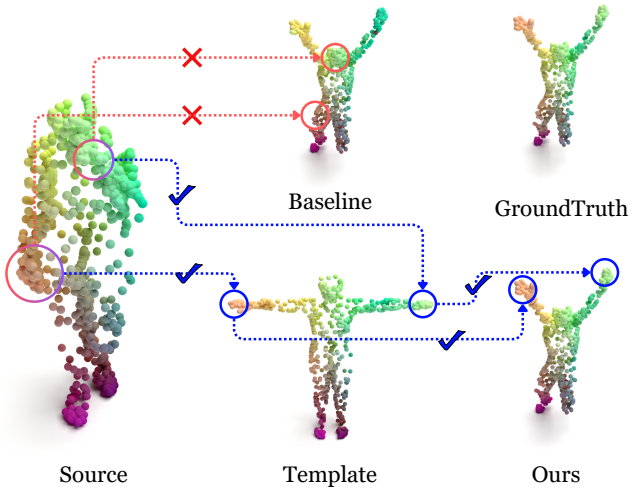


Figure 1. **Visualization of template-assisted shape correspondence results.** Correspondences are depicted by transferring colors from the source to the target based on matching results. The baseline incorrectly aligns the hands with the head and knees between unconventional shapes. In contrast, our method leverages the template to establish accurate correspondences for the hands.

[†]Corresponding Author

the similarity between point cloud features to establish correspondences. However, in the case of unconventional point cloud pairs, the complete structure is distorted or destroyed due to limb adhesion, bending, and other postures. Without explicit structural guidance, directly calculating similarity is often insufficient to guarantee accurate correspondences. Therefore, it is essential to consider how to construct template shapes as intermediaries to assist in establishing correct correspondences.

By analyzing prior shape correspondence methods, we have identified two pivotal issues that must be considered in shape correspondence. 1) *How to generate suitable templates for point cloud pairs?* Traditional point cloud template construction [11, 37, 41] mainly relies on manual selection or autoencoders [41]. While effective for shape-stable rigid objects in the same category, balancing template complexity becomes challenging for diverse non-rigid objects of different categories. Simple templates are clear but lack structural information, risking information loss during correspondence propagation. Complex templates offer richer information but are sensitive to noise, compromising accuracy. Despite attempts [26, 43] with learnable approaches, incomplete template structure remains a challenge. A more effective modeling approach is needed to generate templates suitable for intricate point cloud pairs. 2) *How to effectively leverage templates for better correspondences?* Establishing direct correspondences between unconventional shapes is challenging, but building correspondences between an unconventional shape and a suitable template is relatively straightforward. A correct mapping can be formed by finding corresponding template points for the source points and identifying corresponding points on the target point cloud based on the template. Direct similarity calculations between point clouds may introduce noise and yield ambiguous similarity weights. The template allows each point in the point cloud to compute a correlation vector, providing more stable information. Seeking consensus among these correlation vectors enhances appropriate point correspondences while mitigating erroneous ones.

To achieve the above goal, we propose an *Unsupervised Template-Assisted Point Cloud Shape Correspondence Network* (TANet), including a template generation module and a template assistance module. In the template generation module, we incorporate a template bank with several learnable templates, effectively balancing template complexity in a data-driven manner. The space aligner enriches templates with comprehensive shape structures by establishing a mapping between the template feature space and the coordinate space. The space aligner is supervised by the coordinates and encoded features of the point cloud pairs. In the template assistance module, the adaptive selector chooses the most suitable template from the bank based on geometric and semantic attributes. The correlation fusion process enhances point features with the template correlation vector via attention, suppressing noise and ambiguity. Moreover, we devise a transitive consistency loss to ensure coherence between the similarity computed through the template and directly computed between the source and target point clouds.

To sum up, the contributions of this work can be summarized as follows: (i) We propose an unsupervised template-assisted point cloud shape correspondence network, achieved by jointly designing a template generation module and a template assistance module. (ii) We introduce a template generation module comprising a template bank with learnable templates and the space aligner for building explicit structures. The template assistance module adaptively selects a suitable template for robust point representations and maintains transitive consistency in shape correspondences. (iii) Extensive experiments on TOSCA [3] and SHREC'19 [25] benchmarks show the proposed TANet outperforming state-of-the-art methods. Cross-dataset experiments on SMAL [45] and SURREAL [11] demonstrate our method's desirable generalization capabilities.

2. Related Work

In this Section, we give a brief overview of related works on point cloud shape correspondence, including deep learning for point clouds, shape correspondence, and point cloud template.

Deep Learning for Point Clouds. PointNet [29] employs a symmetrical function to aggregate all point features, resulting in a permutation-invariant global feature representation. Building upon this, PointNet++ [30] takes the concept further by incorporating local information, enhancing the representation capabilities and efficiency. DGCNN [38] introduces EdgeConv [38] blocks, dynamically updating neighborhood information based on dynamic graphs, ultimately delivering improved performance in point cloud analysis. Similarly, KPConv [36] introduces point-wise convolution operators for point cloud feature learning. In the era of transformers, Point Cloud Transformer [12] stands out as a purely global Transformer network, pioneering self-attention layers instead of encoder layers within the PointNet framework. Point Transformer [44] operates within the local neighborhood of the target point cloud, enabling hierarchical extraction of geometric and semantic features.

Shape Correspondence. The goal of shape correspondence is to accurately find the point-to-point correspondence between two point clouds. Traditional spectral-based methods [2, 14, 27, 35] utilize mesh data to project features onto Laplace-Beltrami Operator (LBO) eigenbasis and then learn a transformation between the eigendecomposition of source and target shapes, which will be translated to a dense shape correspondence. Cao et al. [4] transfer rich structure and connectivity information in mesh data

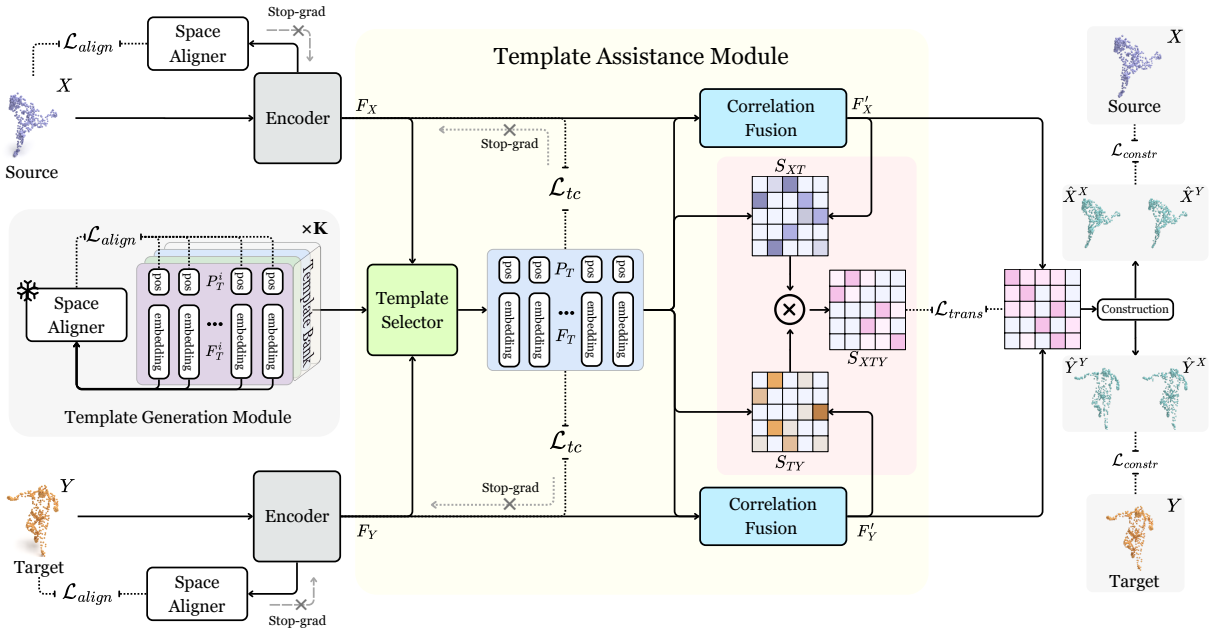


Figure 2. **Illustration of the TANet.** TANet comprises an encoder, a template generation module, and a template assistence module. The template generation module produces several learnable shape templates in the template bank with a space aligner. The template assistance module selects suitable templates for point cloud pairs and improve accuracy via correlation fusion and transitive consistency.

to point cloud shape for robust matching. However, these methods are constrained by complex preprocessing and a reliance on vertex connectivity information. In contrast, point-based methods [5, 7, 9, 13, 16, 18, 34, 42] process point clouds directly without depending on connectivity information. CorrNet3D [42] and DPC [16] harness the established DGCNN [38] network to extract point representations through shape reconstruction. HSTR [13] has introduced a multi-receptive-field transformer point cloud encoder that considers local point cloud structures and long-range contextual information. SE-ORNet [7] incorporates an orientation estimation module to align the orientations of point cloud pairs, achieving precise matching results, particularly for symmetrical parts. These methods directly calculate point-wise similarities, making them susceptible to matching errors caused by noise and incomplete representations. In contrast, our approach constructs and utilizes templates to establish stable correspondences.

Point Cloud Template. Point cloud templates, such as shape templates, scene priors, and semantic priors, are pivotal in various point cloud tasks, including 6D pose estimation [17, 37], reconstruction [10, 20], scene synthesis [26], and segmentation [23, 33, 39]. Tian et al. [37] develop an autoencoder trained on a collection of object models, which decodes the mean latent embedding to create shape templates. For similar rigid objects, generative methods can yield robust and stable templates. ScenePrior [26] encodes scene priors into a latent space and utilizes sampling to synthesize plausible 3D scenes. SemAffiNet [39] and Mask3D [33] integrate a learnable implicit semantic prior via transformer blocks with learnable queries. However, the complex shape and category variations pose a challenge for non-rigid objects to obtain well-balanced explicit templates through generative methods. The learnable implicit modeling struggles to achieve explicit stuctures. Based on the above discussion, we propose TANet, which integrates the template generation and assistance modules within a unified framework. Through technical designs, the learnable templates not only balance shape complexity but also possess an explicitly structured shape.

3. Method

3.1. Overview

The unsupervised point cloud correspondence task strives to establish accurate one-to-one correspondences between source and target point clouds without annotations. As illustrated in Figure 2, TANet primarily comprises three key modules: a general point cloud encoder, a Template Generation Module, and a Template Assistance Module. The choice of point cloud encoder follows the previous study [13]. The detailed design and optimization process for the Template Generation Module will be expounded upon in Section 3.2. Moreover, Section 3.3 will provide further insights into the Template Assistance Module. Finally, the training and inference process are discussed in Section 3.4.

3.2. Template Generation Module

The template generation module consists primarily of two components: a template bank and a space aligner.

Template bank. In the template bank, we integrate K learnable explicit shape templates, each comprising learnable positions $P_T^i \in \mathbb{R}^{N\times 3}$ and learnable embeddings $F_T^i \in \mathbb{R}^{N\times d}$. During the training phase, the positions and embeddings of templates are updated concurrently with the network. In the inference phase, the positions and embeddings of template shapes remain unchanged, eliminating the need for additional updates based on input point clouds. The learnable embeddings F_T^i parameters are initialized using Gaussian random initialization. To expedite template training, we utilize random shapes in datasets to initialize the learnable positions P_T^i.

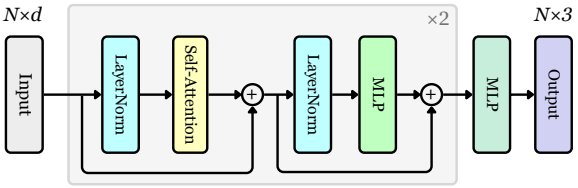


Figure 3. **Space aligner structure.** The predicted point positions in the output are constrained to match the true point positions.

Space aligner. Space aligner ϕ is crafted to predict point positions through point embeddings, aiding in template learning. The specific structure is illustrated in Figure 3. The predicted point positions undergo adjustment to align with true positions:

$$\mathcal{L}_{align} = \text{SmoothL}_1(\text{P}, \phi(\text{F})),$$

where P are true positions, and F are point embeddings. The space aligner is trained with the encoded feature and the coordinates from point cloud pairs. During the inference phase, the space aligner is not involved in the computation.

Additionally, the training of templates also includes the template construction loss $\mathcal{L}_{tc}$, which enforces consistency between the template shape and the cross-constructed template shape by point cloud pairs. Specifically, the cross-construction process is computed as follows:

$$\hat{t}_{x_i} = \sum_{j\in\mathcal{N}_T(x_i)} \frac{e^{s_{ij}}}{\sum_{l\in\mathcal{N}_T(x_i)} e^{s_{il}}} t_j, \tag{1}$$

where $x_i \in X$, $t_j \in P_T$ and $s_{ij} \in S_{XT}$. S_{XT} is the similarity matrix between F_X and F_T. $\mathcal{N}_T(x_i)$ represents the latent k-nearest neighbors of x_i in the target F_T. The cross-construction of P_T by the source point cloud X is denoted $\hat{T}_X \in \mathbb{R}^{N\times 3}$, where $\hat{T}_X^i = \hat{t}_{x_i}$. After the introduction of the cross-construction process, the template construction loss $\mathcal{L}_{tc}$ can be computed as follows:

$$\mathcal{L}_{tc} = \mathbf{CD}(P_T, \hat{T}_X) + \mathbf{CD}(P_T, \hat{T}_Y), \tag{2}$$

where $\mathbf{CD}$ denotes chamfer distance. It is essential to highlight that the point encoder remains unaltered during the backpropagation of $\mathcal{L}_{align}$ and $\mathcal{L}_{tc}$ gradients, preventing interference with the feature extraction of point cloud pairs.

3.3. Template Assistance Module

The template assistance module aims to enhance the correspondence accuracy in point cloud pairs through templates. It comprises three primary components: Template Selector, Correlation Fusion, and Transitive Consistency.

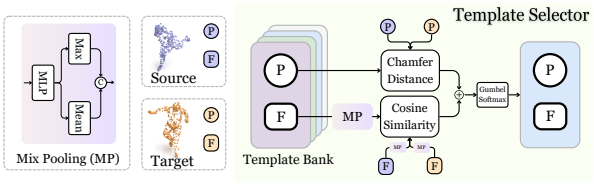


Figure 4. **Template selector structure.** The best suitable template is determined through geometric and semantic measures of similarity between templates and point cloud pairs.

Template selector. The input to the Template Selector comprises point cloud coordinates $X\&Y \in \mathbb{R}^{N\times 3}$ and features $F_X\&F_Y \in \mathbb{R}^{N\times d}$, along with template point cloud coordinates $\{P_T^i | i = 1, 2, \ldots, K\}$ and embeddings $\{F_T^i | i = 1, 2, \ldots, K\}$ from the template bank. Geometric and semantic similarities between the point cloud pair and templates guide the adaptive template selection. As illustrated in Figure 4, it involves calculating the chamfer distance as the geometric similarity between the point cloud pair and template point cloud. To get semantic similarity, the point cloud features and template embeddings undergo mix pooling and cosine similarity computation. The fusion of geometric and semantic similarities is employed in Gumbel-softmax to ascertain the most fitting template T for the given point cloud pair. The specific computation formula is as follows:

$$\begin{aligned} T = \mathbf{GS}\{&(\mathbf{MP}(F_T^i)\cdot\mathbf{MP}(F_X) + \mathbf{MP}(F_T^i)\cdot\mathbf{MP}(F_Y)) \\ &+ (\mathbf{CD}(P_T^i, X) + \mathbf{CD}(P_T^i, Y)) | i = 1, 2, \ldots, K\}, \end{aligned} \tag{3}$$

where $\mathbf{GS}$ represents Gumbel-Softmax, $\mathbf{MP}$ denotes Mix Pooing, and $\mathbf{CD}$ stands for chamfer distance.

Correlation fusion. As depicted in Figure 5, template embeddings F_T and point cloud features F_X/F_Y are input into the correlation fusion process to enhance point cloud features. Initially, a correlation between the point cloud and the template is established. Subsequently, an attention mechanism incorporates the correlation vectors into point cloud features, yielding $F'_X/F'_Y \in \mathbb{R}^{N\times d}$. The similarity in the attention mechanism is computed based on point cloud features to underscore the spatial distribution of shapes, enhancing perception to spatial relationships. The introduction of template correlation aids in refining point cloud features and mitigating ambiguous representations in certain noisy point clouds.

Transitive consistency. Utilizing the fused source features F'_X and template embeddings F_T, we compute the similarity matrix S_{XT}, and S_{TY} is obtained by calculating the similarity between F_T and F'_Y. The multiplication of S_{XT} and S_{TY} matrices yields the similarity matrix S_{XTY},

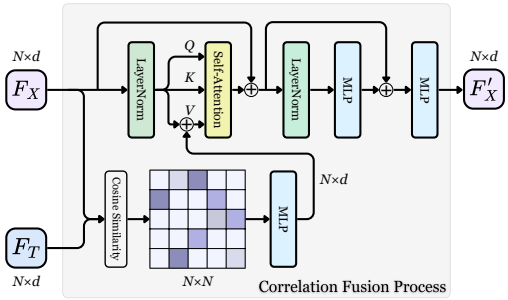


Figure 5. **The correlation fusion process.** Features of point cloud pairs and embeddings of templates compute correlation vectors, which are then fused using an attention mechanism.

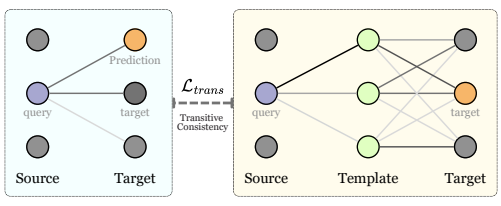
reflecting the similarity between the source and target point clouds transmitted through the template. As shown in Figure 6, (a) Direct similarity illustrates the process of directly calculating the similarity between the source point cloud and the target point cloud:

$$S_{XY} = F'_X {F'_Y}^{\top}. \tag{4}$$

In contrast, (b) Template-assisted similarity demonstrates the computation of S_{XTY}. The multiplication of matrices S_{XT} and S_{TY} essentially results in an indirect computation of the similarity relationship between the source and target point clouds. This process involves a weighted sum of the similarity calculated between the template point cloud and the target point cloud, providing a more stable and accurate similarity result for source and target points. The Template-assisted similarity method helps mitigate the impact of outliers and noise on the similarity calculation, leading to more reliable results. In order to guide the establishment of more stable and accurate similarity relationships between point clouds without compromising the inference speed, we introduce the Transitive Consistency loss ($\mathcal{L}_{trans}$), promoting the consistency between S_{XTY} and the similarity matrix S_{XY}. The Transitive Consistency loss is calculated as follows:

$$\mathcal{L}_{trans} = \mathbf{CE}(S_{XTY}, S_{XY}), \tag{5}$$

where $\mathbf{CE}$ means Cross-Entropy loss.



(a) Direct Similarity (b) Template-assisted Similarity

Figure 6. **Transitive Consistency.** (a) the direct similarity computation between the source and target point clouds. (b) a stable template is employed to transmit the similarity relation. L_{trans} ensures the consistency between these two similarity results.

3.4. Model Training & Inference

Following the previous work [7, 13, 16], we also incorporate the construction loss ($\mathcal{L}_{constr}$), encompassing both cross-construction and self-construction between the source shape X and the target shape Y. The construction loss promotes feature smoothness and unique correspondences. The computation process and formula for cross-construction and self-construction are similar to Equation 1. Specific formulas are detailed in the supplementary material. To sum up, the total loss of TANet is formulated as:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{trans} + \lambda_2 \mathcal{L}_{align} + \lambda_3 \mathcal{L}_{tc} + \lambda_4 \mathcal{L}_{constr}, \tag{6}$$

where λ_i are hyperparameters, balancing the contribution of different loss terms. The training of both the template and the network is jointly achieved, enabling end-to-end training without requiring additional prolonged iterations.

During inference, we set the closest point y_{j^*} in the feature space for each point x_i as its corresponding point. This selection rule can be formulated as:

$$f(x_i) = y_{j^*}, j^* = \underset{j}{\operatorname{argmax}}(s_{ij}). \tag{7}$$

4. Experiments

4.1. Experimental Setup

Datasets. We evaluate our method on the most popular benchmarks, including the TOSCA [3] dataset and the SHREC'19 [25] dataset. TOSCA comprises 41 different shapes of various animal species. Following previous methods [13, 16], we pair these shapes to create both the training and testing sets. SHREC'19 consists of 44 real human models paired into 430 shape correspondence samples. SURREAL [11] and SMAL [45] are two large-scale datasets that we employ to evaluate the generalization capabilities of our method. The SURREAL dataset comprises 230,000 training shapes, from which we randomly sample and pair 2,000 shape pairs for training. The SMAL dataset includes parameterized animal models for generating shape pairs to train shape correspondence models. Furthermore, robustness experiments are conducted on the real-scanned Owlii dataset [40] and the partial shape dataset SHREC'16 [6]. Besides, we maintain a consistent point count of $n = 1024$ in accordance with prior studies [13, 16, 42].

Evaluation Metrics. The primary evaluation metrics for the point cloud shape correspondence task include two key measures: average correspondence error and correspondence accuracy. The average correspondence error measures a source and target shape pair (X, Y) as follows:

$$err = \frac{1}{n} \sum_{x_i \in X} \left| f(x_i) - y_{gt} \right|_2, \tag{8}$$

where $y_{gt} \in Y$ represents the ground-truth matching point

to x_i. Correspondence accuracy is formulated as follows:

$$acc(\epsilon) = \frac{1}{n} \sum_{x_i \in X} \mathbb{1}\left(|f(x_i) - y_{gt}|_2 < \epsilon d\right), \quad (9)$$

where $\mathbb{1}(\cdot)$ denotes the indicator function, d is the maximum Euclidean distance between points in Y, and $\epsilon \in [0, 1]$ signifies the error tolerance.

Implementation details. The number of learnable templates (K) in the Template Bank is set to 4, with a template embedding dimension of 512. The Space Aligner incorporates two attention blocks, each with a dimensionality of 512. We employ a transformer encoder, akin to existing method [13], featuring four layers of attention blocks as our point cloud encoder. For the Correlation Fusion process, the dimensions of self-attention and MLP are set to 512. In Equation 6, λ_1, λ_2, λ_3, and λ_4 are set to 0.5, 0.5, 1, and 1. All experiments with our method on various datasets are conducted on a single GeForce RTX 3090 device using the PyTorch 1.10.1 framework [28]. Model training utilizes the AdamW [22] optimizer with a learning rate of 5e-4, a weight decay of 5e-4, and a batch size of 4.

Table 1. **Comparison on TOSCA and SHREC'19 benchmarks.** Acc signifies correspondence accuracy at 0.01 error tolerance, and err denotes average correspondence error. Higher accuracy and lower error reflect a better result, with the best and second-best outcomes highlighted in bold and underlined, respectively.

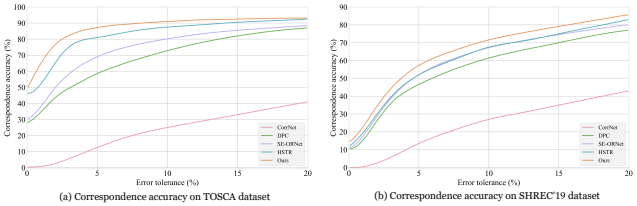
Method	Input	TOSCA		SHREC'19	
		acc ↑	err ↓	acc ↑	err ↓
SURFMNet[31]	Mesh	/	/	5.9%	0.2
CorrNet3D[42]	Point	0.3%	32.7	0.4%	33.8
DPC[16]	Point	34.7%	2.8	15.3%	5.6
SE-ORNet[7]	Point	38.3%	2.7	17.5%	5.1
HSTR[13]	Point	<u>52.3%</u>	<u>1.2</u>	<u>19.3%</u>	<u>4.9</u>
Ours	Point	**65.1%**	**0.7**	**21.5%**	**4.5**

4.2. Comparison with State-of-the-art Methods

We conducted comprehensive experimental comparisons with existing methods on the TOSCA animal dataset and the SHREC'19 human dataset. Furthermore, to ensure a fair comparison with existing methods, our approach is implemented without the use of post-processing or additional connectivity information.

Evaluation on TOSCA dataset. As shown in Table 1, our approach demonstrates significant improvement on the TOSCA dataset, achieving a new state-of-the-art performance with a 12.8% increase in accuracy and a minimal average error of 0.7cm. For a comprehensive depiction of accuracy across different error tolerances, we present correspondence accuracy in Figure 7(a). Similarly, our method outperforms others at different error tolerances.

Evaluation on SHREC'19 dataset. As indicated in Table 1, our method demonstrates significant improvement on the SHREC'19 dataset, achieving an impressive 2.2% increase in accuracy and a reduction of 0.4cm in error, consequently establishing a new state-of-the-art performance. Figure 7(b) illustrates the correspondence accuracy of various competitive methods at different error tolerances, showcasing our method's clear superiority across different levels of error tolerance.



(a) Correspondence accuracy on TOSCA dataset (b) Correspondence accuracy on SHREC'19 dataset

Figure 7. **Correspondence accuracy at various error tolerances.** (a) Corresponding accuracy of different methods on the TOSCA dataset under various error tolerances. (b) Experiments similar to (a) conducted on the SHREC'19 dataset.

Table 2. **Evaluation of the model with different designs on TOSCA dataset.** TGM denotes the template generation module, L_{tc} stands for template construction loss, TS represents template selector, CF denotes correlation fusion process, and L_{trans} is the transitive consistency loss.

	TGM	$\mathcal{L}_{tc}$	TS	CF	$\mathcal{L}_{trans}$	TOSCA	
						acc ↑	err ↓
[A]	✗	✗	✗	✗	✗	52.3%	1.2
[B]	✓	✗	✗	✓	✗	54.6%	1.2
[C]	✓	✗	✗	✗	✓	55.8%	1.1
[D]	✓	✓	✗	✓	✗	59.3%	1.0
[E]	✓	✓	✗	✗	✓	58.8%	1.0
[F]	✓	✓	✓	✓	✗	61.9%	0.9
[G]	✓	✓	✓	✗	✓	61.3%	0.9
[H]	✓	✓	✓	✓	✓	**65.1%**	**0.7**

4.3. Ablation Study

Evaluation of the model with different designs. In Table 2, we conduct extensive ablation studies on the TOSCA dataset to validate the effectiveness of our model designs. Specifically, [A] presents a baseline method without any template-related design. [B] and [C] demonstrate the employment of the template generation module, assisted respectively through correlation fusion and transitive consistency loss, resulting in accuracy improvements of 2.3% and 3.5%. The limited improvement is attributed to the difficulty in structural learning of templates without appropriate loss constraints. [D] and [E] illustrate the introduction of template construction loss to aid in the learning and optimization of template structures, leading to advancements in accuracy and error compared to settings [B] and [C]. Building upon this, [F] and [G] further employ a template selector to identify the most suitable template for aiding the establishment of correspondence for specific shapes. The further improvements in accuracy and error indicate that adaptive

template selection significantly contributes to accurate correspondence establishment. Finally, in [H], we present the performance of the complete model, highlighting the essential roles played by different modules in collectively achieving accurate point cloud shape correspondences.

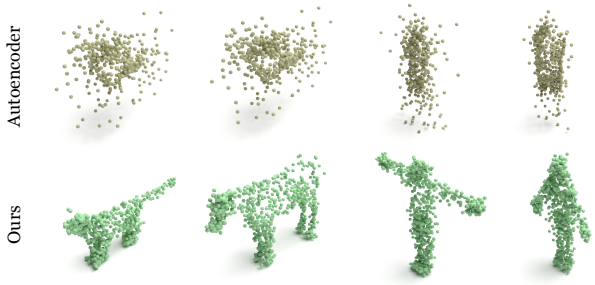


Figure 8. **Visual comparison of template generation methods.** The templates generated by the template generation module exhibit clearer structures in both animal and human templates.

Effectiveness of the template generation module. Figure 8 compares templates generated by our template generation module and the common autoencoder-based approach [37]. Our designed learnable templates, constrained by the space aligner and construction loss, provide more structural templates to aid point cloud shape correspondence. However, due to the significant shape variations of non-rigid bodies, it remains challenging to generate perfect templates without direct shape supervision. The existing templates are merely a sub-optimal form. Additionally, we carefully consider the number of templates in Table 3. A single or dual template falls short of capturing the shape variations, resulting in a significant decrease in accuracy. While eight templates can slightly enhance accuracy, they lead to a substantial increase in model parameters. Hence, we offer a trade-off to set up four learnable templates.

Table 3. **Ablation studies on the number of templates.**

#Template	Param.	acc ↑
1	7.3M	56.8%
2	7.8M	61.4%
4	8.8M	65.1%
8	10.9M	66.2%

Table 4. **Comparison of similarity computation methods.**

Settings	FLOPs(G)	TOSCA	
		acc ↑	err ↓
Direct	1.07	65.1%	0.73
Transitive	4.29	66.3%	0.67

Effectiveness of the template assistance module. The effectiveness of correlation fusion has been validated in Table 2. Here, we further delve into the discussion regarding the computation of similarities between point cloud pairs. The purpose of transitive consistency loss is to constrain the direct similarity computation between the source and target point clouds to align with the similarity computed through template transitivity. As demonstrated in Table 4, similarity computation through template propagation has been proven to be a more stable and accurate approach. However, considering the increased computational complexity associated with the large number of point clouds, we adopt the direct similarity computation method during inference to enhance model efficiency. The transitive consistency loss only constrains the learning of point cloud representations through similarity-guided learning during training.

Table 5. **Cross-dataset generalization evaluation on SMAL and SURREAL benchmarks.** Acc signifies correspondence accuracy at 0.01 error tolerance, and err denotes average correspondence error. The best and second-best outcomes highlighted in bold and underlined, respectively.

Method	Input	SMAL		SURREAL	
		acc ↑	err ↓	acc ↑	err ↓
SURFMNet[31]	Mesh	5.9%	0.2	4.3%	0.3
GeoFMNet[9]	Mesh	/	/	8.2%	0.2
Diff-FMaps[24]	Point	/	/	4.0%	7.1
3D-CODED[11]	Point	0.5%	19.2	2.1%	8.1
Elementary[8]	Point	0.5%	13.7	2.3%	7.6
CorrNet3D[42]	Point	5.3%	9.8	6.0%	6.9
DPC[16]	Point	33.2%	5.8	17.7%	6.1
SE-ORNet[7]	Point	<u>36.4%</u>	<u>3.9</u>	**21.5%**	**4.6**
HSTR[13]	Point	33.9%	5.6	19.4%	5.6
Ours	Point	**37.1%**	**3.7**	<u>20.6%</u>	<u>4.8</u>

4.4. Cross-dataset Generalization Performance

The cross-dataset generalization ability is a crucial evaluation metric for the point cloud shape correspondence task. Following the conventional setup [7, 13, 16], generalization evaluation involves training on one dataset and testing on another. Specifically, training on the SMAL dataset and testing on the TOSCA dataset, as well as training on the SURREAL dataset and testing on the SHREC'19 dataset. As shown in Table 5, our method surpasses existing state-of-the-art approaches, establishing a new state-of-the-art performance by overcoming domain differences on the SMAL dataset. Regarding generalization to the SURREAL dataset, our results are also competitive. Confronting challenges posed by human body symmetry and directional rotations in the SURREAL and SHREC'19 datasets, our approach slightly lags behind the SE-ORNet [7], specifically designed for this issue. However, TANet significantly outperforms other point cloud shape correspondence methods.

4.5. Visual Comparison

In Figure 9, we present visual comparisons of our method with competitive approaches. When the source point cloud forms an unconventional shape, such as a distorted standing posture, other methods struggle to distinguish between the hand and facial structures. Our method more accurately establishes the correspondence for hands and heads by leveraging learned templates as an intermediary.

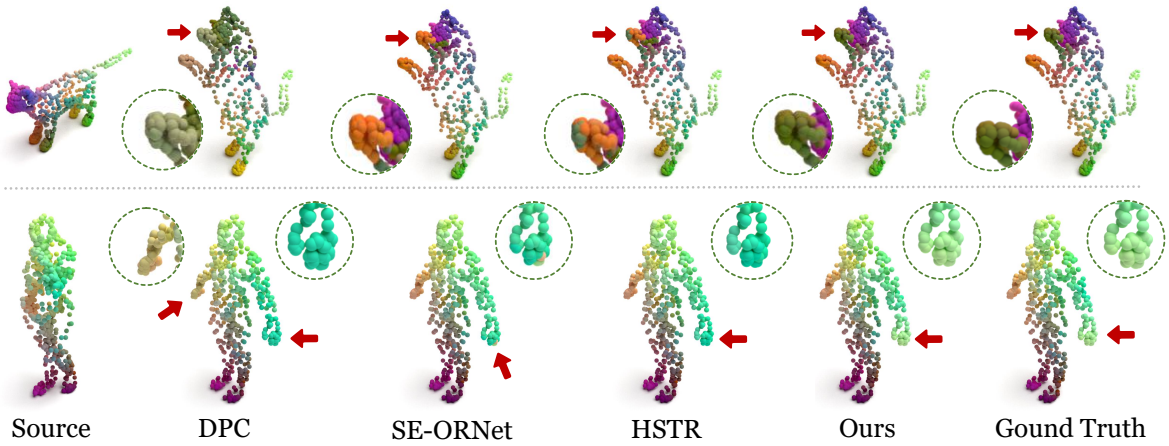


Figure 9. **Visual comparison on TOSCA and SHREC'19.** We visually compared the correspondence results of different methods on unconventional shape pairs, providing intuitive validation of the proposed approach's superiority in establishing correspondences.

Figure 10. **Visualization of the correspondence results on partial-shape SHREC'16 dataset.** The partial shapes contain four species, including horses, cats, dogs, and humans. The generation of partial shapes in the SHREC'16 dataset primarily involves two methods: cutting and holing to wipe off body parts.

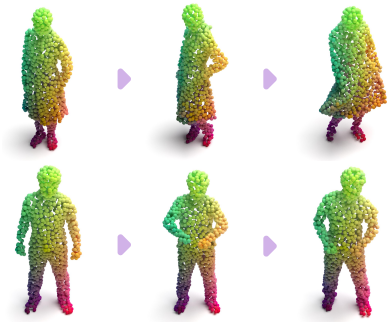


Figure 11. **Visualization of the correspondence results on real scanned Owlii dataset.** The results contain two consecutive action sequences involving individuals wearing different attire.

4.6. Robustness Analysis

To further validate the robustness and generalization capabilities of our method, we directly evaluate our trained model on the SHREC'16 [6] and Owlii [40] datasets, presenting visual results. As shown in Figure 10, the SHREC'16 dataset comprises point cloud pairs of partial shapes, simulating challenges encountered in real-world applications such as occlusion, missing parts, and noise. Although only the upper body of the cat is present, and the dog is represented by a sparse point cloud with minimal parts of its body, our method discriminatively models local structures and accurately establishes shape correspondences. Moreover, the correspondence results of the partial human shape are predictions of the model trained on the animal dataset, providing further validation of the generalization capability. Illustrated in Figure 11, the Owlii dataset includes real-scanned human motion sequences, introducing challenges with scan noise and variations in attire and accessories. Nevertheless, TANet accurately establishes the shape correspondences for these sequence point clouds, including details such as dresses worn by the individuals. The visual results on SHREC'16 and Owlii further substantiate TANet's superior robustness and generalization.

5. Conclusion

In this paper, we propose an unsupervised template-assisted point cloud shape correspondence network, including a template generation module and a template assistance module. Specifically, the template generation module is introduced to learn a set of learnable templates that balance shape complexity and possess an explicitly structured shape. The template assistance module is designed to leverage the generated templates adaptively to achieve more accurate shape correspondences from template selection, feature learning, and transitive consistency perspectives. Extensive experiments on the SHREC'19 and TOSCA benchmarks demonstrate the superiority of TANet. Cross-dataset experiments on SURREAL and SMAL showcase our method's desirable generalization capabilities.

6. Acknowledgements

This work was partially supported by the National Defense Basic Scientific Research program (JCKY2022911B002) and the National Nature Science Foundation of China (NSFC 12150007, NSFC 62121002).

References

[1] Fabio Arena, Mario Collotta, Giovanni Pau, and Francesco Termine. An overview of augmented reality. *Computers*, 11 (2):28, 2022. 1

[2] Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proceedings of the National Academy of Sciences*, 103(5):1168–1172, 2006. 1, 2

[3] Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. *Numerical geometry of non-rigid shapes*. Springer Science & Business Media, 2008. 2, 5

[4] 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. 2

[5] Etienne Corman, Maks Ovsjanikov, and Antonin Chambolle. Supervised descriptor learning for non-rigid shape matching. In *European conference on computer vision*, pages 283–298. Springer, 2014. 3

[6] Luca Cosmo, Emanuele Rodola, Michael M Bronstein, Andrea Torsello, Daniel Cremers, Y Sahillioglu, et al. Shrec'16: Partial matching of deformable shapes. *Proc. 3DOR*, 2(9): 12, 2016. 5, 8

[7] 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. 1, 3, 5, 6, 7

[8] Theo Deprelle, Thibault Groueix, Matthew Fisher, Vladimir Kim, Bryan Russell, and Mathieu Aubry. Learning elementary structures for 3d shape generation and matching. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 7

[9] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep geometric functional maps: Robust feature learning for shape correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8592–8601, 2020. 3, 7

[10] Georgia Gkioxari, Nikhila Ravi, and Justin Johnson. Learning 3d object shape and layout without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1695–1704, 2022. 3

[11] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 3d-coded: 3d correspondences by deep deformation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 230–246, 2018. 1, 2, 5, 7

[12] 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(2):187–199, 2021. 2

[13] 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. 1, 3, 5, 6, 7

[14] Qi-Xing Huang, Bart Adams, Martin Wicke, and Leonidas J Guibas. Non-rigid registration under isometric deformations. In *Computer Graphics Forum*, pages 1449–1457. Wiley Online Library, 2008. 1, 2

[15] Dany Laksono and Trias Aditya. Utilizing a game engine for interactive 3d topographic data visualization. *ISPRS International Journal of Geo-Information*, 8(8):361, 2019. 1

[16] 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. 1, 3, 5, 6, 7

[17] Jiehong Lin, Zewei Wei, Changxing Ding, and Kui Jia. Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In *European Conference on Computer Vision*, pages 19–34. Springer, 2022. 3

[18] Or Litany, Tal Remez, Emanuele Rodola, Alex Bronstein, and Michael Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. In *Proceedings of the IEEE international conference on computer vision*, pages 5659–5667, 2017. 3

[19] Roee Litman and Alexander M Bronstein. Learning spectral descriptors for deformable shape correspondence. *IEEE transactions on pattern analysis and machine intelligence*, 36(1):171–180, 2013. 1

[20] Haolin Liu, Yujian Zheng, Guanying Chen, Shuguang Cui, and Xiaoguang Han. Towards high-fidelity single-view holistic reconstruction of indoor scenes. In *European Conference on Computer Vision*, pages 429–446. Springer, 2022. 3

[21] Ming Liu. Robotic online path planning on point cloud. *IEEE transactions on cybernetics*, 46(5):1217–1228, 2015. 1

[22] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[23] Jiahao Lu, Jiacheng Deng, Chuxin Wang, Jianfeng He, and Tianzhu Zhang. Query refinement transformer for 3d instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18516–18526, 2023. 3

[24] 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, 7

[25] 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*, page 3, 2019. 2, 5

[26] Yinyu Nie, Angela Dai, Xiaoguang Han, and Matthias Nießner. Learning 3d scene priors with 2d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 792–802, 2023. 2, 3

[27] 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

[28] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 6

[29] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 2

[30] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 2

[31] Jean-Michel Roufosse, Abhishek Sharma, and Maks Ovsjanikov. Unsupervised deep learning for structured shape matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1617–1627, 2019. 6, 7

[32] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, Mihai Dolha, and Michael Beetz. Towards 3d point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11):927–941, 2008. 1

[33] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d: Mask transformer for 3d semantic instance segmentation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8216–8223. IEEE, 2023. 3

[34] Ramana Sundararaman, Gautam Pai, and Maks Ovsjanikov. Implicit field supervision for robust non-rigid shape matching. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 344–362. Springer, 2022. 3

[35] Art Tevs, Alexander Berner, Michael Wand, Ivo Ihrke, and H-P Seidel. Intrinsic shape matching by planned landmark sampling. In *Computer graphics forum*, pages 543–552. Wiley Online Library, 2011. 1, 2

[36] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6411–6420, 2019. 2

[37] Meng Tian, Marcelo H Ang, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 530–546. Springer, 2020. 2, 3, 7

[38] 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)*, 38(5):1–12, 2019. 2, 3

[39] Ziyi Wang, Yongming Rao, Xumin Yu, Jie Zhou, and Jiwen Lu. Semaffinet: Semantic-affine transformation for point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11819–11829, 2022. 3

[40] Yi Xu, Yao Lu, and Ziyu Wen. Owlii dynamic human mesh sequence dataset. In *ISO/IEC JTC1/SC29/WG11 m41658, 120th MPEG Meeting*, 2017. 5, 8

[41] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 206–215, 2018. 2

[42] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6052–6061, 2021. 1, 3, 5, 6, 7

[43] Cheng Zhang, Zhaopeng Cui, Yinda Zhang, Bing Zeng, Marc Pollefeys, and Shuaicheng Liu. Holistic 3d scene understanding from a single image with implicit representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8833–8842, 2021. 2

[44] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021. 2

[45] Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6365–6373, 2017. 1, 2, 5