


UTOPIC: Uncertainty-aware Overlap Prediction Network for Partial Point Cloud Registration

Zhilei Chen¹ , Honghua Chen^{1†}, Lina Gong¹, Xuefeng Yan¹, Jun Wang¹, Yanwen Guo³, Jing Qin⁴, Mingqiang Wei^{1,2†}

¹School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China

²Shenzhen Research Institute, Nanjing University of Aeronautics and Astronautics, Shenzhen, China

³Department of Computer Science, Nanjing University, Nanjing, China

⁴School of Nursing, Hong Kong Polytechnic University, Hong Kong, China

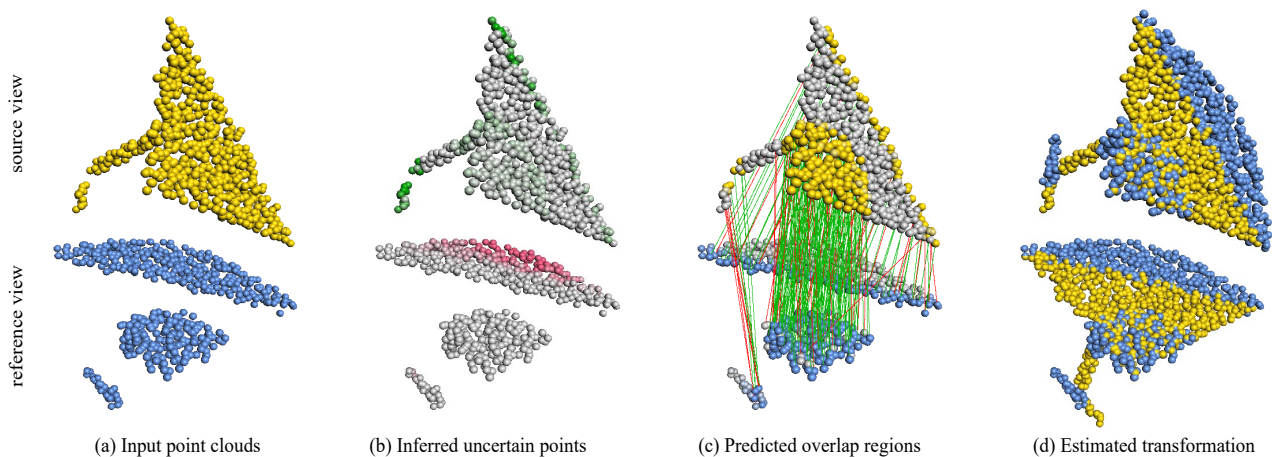


Figure 1: We present an uncertainty-aware overlap prediction network for partial point cloud registration, which is able to suppress confusing overlapping regions, thus obtaining more reliable overlap scores and more exact dense correspondences for robust registration. The redder or more green points in (b) are the points with higher uncertainties. The colorful points in (c) are points with higher overlap scores. The green lines denote the correct correspondences and the red lines represent the wrong correspondences.

Abstract

High-confidence overlap prediction and accurate correspondences are critical for cutting-edge models to align paired point clouds in a partial-to-partial manner. However, there inherently exists uncertainty between the overlapping and non-overlapping regions, which has always been neglected and significantly affects the registration performance. Beyond the current wisdom, we propose a novel uncertainty-aware overlap prediction network, dubbed UTOPIC, to tackle the ambiguous overlap prediction problem; to our knowledge, this is the first to explicitly introduce overlap uncertainty to point cloud registration. Moreover, we induce the feature extractor to implicitly perceive the shape knowledge through a completion decoder, and present a geometric relation embedding for Transformer to obtain transformation-invariant geometry-aware feature representations. With the merits of more reliable overlap scores and more precise dense correspondences, UTOPIC can achieve stable and accurate registration results, even for the inputs with limited overlapping areas. Extensive quantitative and qualitative experiments on synthetic and real benchmarks demonstrate the superiority of our approach over state-of-the-art methods.

CCS Concepts

• Computing methodologies → Point-based models;

1. Introduction

3D sensors are becoming increasingly available and affordable, including various types of 3D laser scanners, and RGB-D cameras

[†] Co-corresponding authors: H. Chen and M. Wei.

(such as Microsoft Kinect, Intel RealSense, Stereolabs ZED, and Apple Truth Depth Camera) [WYW*18, YLX*19]. This benefits to accurately represent scanned surfaces, detailing the shape, dimensions, and the size of various objects [CWL*22]. However, the raw point clouds directly captured by these sensors unavoidably require a registration step to synthesize a complete model or a large-scale scene from multiple partial scans [LXW*22].

The problem of partial point cloud registration has been extensively explored [WS19b, YL20, AHY*21, WLD*21, HGU*21, FLLW21, YL22, WYF*22, CWX*22], yet not well-solved. Generally, the central of accurate registration lies in two aspects: (i) *localizing reliable overlapping regions* and (ii) *establishing accurate correspondences*. Specifically, one can sample more matching points from the overlapping portions of two scans [HGU*21, WYF*22], or treat the overlap scores as the confidence of correspondences [YL22]. The larger the overlap ratio of a point cloud pair is, the easier the overlap region is to be exploited to stitch the point cloud pair for a precise alignment. Therefore, faithfully detecting the most reliable overlapping regions is essential for robust partial registration. On the other hand, given accurate overlapping regions, if the learned features are less distinctive, numerous outlier matches may still lead to a poor alignment [AHY*21, WLD*21]. Moreover, there always exists inherent uncertainty between overlapping and non-overlapping regions. For example, points near the boundary between overlapping and non-overlapping regions are quite ambiguous, especially when the overlap ratio is low or there exist noise and/or outliers. Such kind of uncertainty may confuse the prediction of overlapping points and correspondences, thus significantly hindering the performance of registration.

Unlike most of the existing registration methods, we propose to disentangle partial point cloud registration into two sub-goals: (i) to first enhance the feature representation with ample prior shape knowledge and transformation-invariant geometry relation embedding, and (ii) to introduce the uncertainty into point cloud registration to solve the ambiguous overlap prediction problem.

We follow this path and introduce *UTOPIC*, a neural model for partial point cloud registration with **UncerTainty-aware Overlap PredICtion**. First, based on the observation that if two partial point clouds can be recovered to their complete versions, it will be easier to align them, we employ a completion decoder to implicitly enrich the feature extractor with geometry shape priors. Second, we collaborate the geometric relation embedding of each point cloud with the Transformer model [VSP*17] to obtain more reliable and geometry-aware feature representations. Based on the above learned features, we propose an uncertainty quantification scheme to measure the overlap uncertainty of each point (see from Fig. 1 (b)), and then lower the importance of uncertain regions for subsequent overlap prediction and dense matching. Finally, quality rigid transformation parameters are computed, by considering both overlap scores and correspondences (see from Fig. 1 (c)) in the SVD solver. Experiments on both synthetic and real-scanned data and detailed analysis show that our approach achieves state-of-the-art performance compared with its competitors.

Our main contributions are as follows:

- We design a novel point cloud registration network, which is the

first to explore the overlap uncertainty to predict high-confidence overlap and accurate correspondences for robust registration.

- We employ a completion decoder to implicitly enrich the feature extractor of prior shape knowledge, which can be used to output more reliable deep features.
- We introduce a geometry transformer incorporated with geometric relation embedding, which better captures the global contextual information in both feature and geometry spaces and enhances cross-feature fusion.
- We propose an uncertainty quantification scheme, which is capable of alleviating the ambiguity of overlapping regions.

2. Related Work

2.1. Feature-based Methods

Feature-based point cloud registration methods usually contain two sub-stages: first extracting powerful feature descriptors and then recovering the transformation with robust pose estimators, e.g., RANSAC. Traditional feature-based methods [RBB09, STDS14] use handcrafted features to construct descriptors. With the prosperity of deep learning techniques, learning-based descriptors [CPK19, HGU*21, AHY*21, WLD*21] achieve more impressive and substantial improvements. FCGF [CPK19] designs a point descriptor based on fully convolutional network and sparse tensor, which is widely used for correspondence search and registration. [AHY*21] achieves rotation invariance by PCA on the neighborhood to find the normal axis. Predator [HGU*21] introduces an overlap attention module and uses probabilistic sampling mechanism for robust RANSAC, which significantly improves the performance of registration for low-overlap point cloud pairs. In contrast with the descriptors that rely on local reference frame, [WLD*21] achieves the rotation invariance by grouping equivariant feature learning. However, establishing correspondences based on feature descriptors becomes more challenging when the input scans have ambiguous geometry structures. Besides, these methods are time-consuming due to RANSAC-like estimators.

2.2. Direct Registration Methods

Direct registration methods estimate the transformation parameters with a neural network in an end-to-end manner. This kind of method can be further divided into two classes, according to whether depending on correspondences.

The correspondence-based methods first extract point-wise features, then calculate the correspondences and finally employ a SVD solver to obtain the transformation. DCP [WS19a] mimics the pipeline of ICP [BM92], and employs a transformer to compute soft correspondences and a differentiable SVD layer to extract the transformation. PRNet [WS19b] utilizes a keypoint detector to obtain keypoint-to-keypoint correspondences for partial point cloud registration. To overcome the disadvantage of matching by simply taking the inner product of feature vectors, [LZX*20] learns to match points based on joint information of the entire geometric features and Euclidean offset for each point pair. [ZSD*22] proposes to learn a partial permutation matching matrix, which does not assign corresponding points to outliers, and implements hard

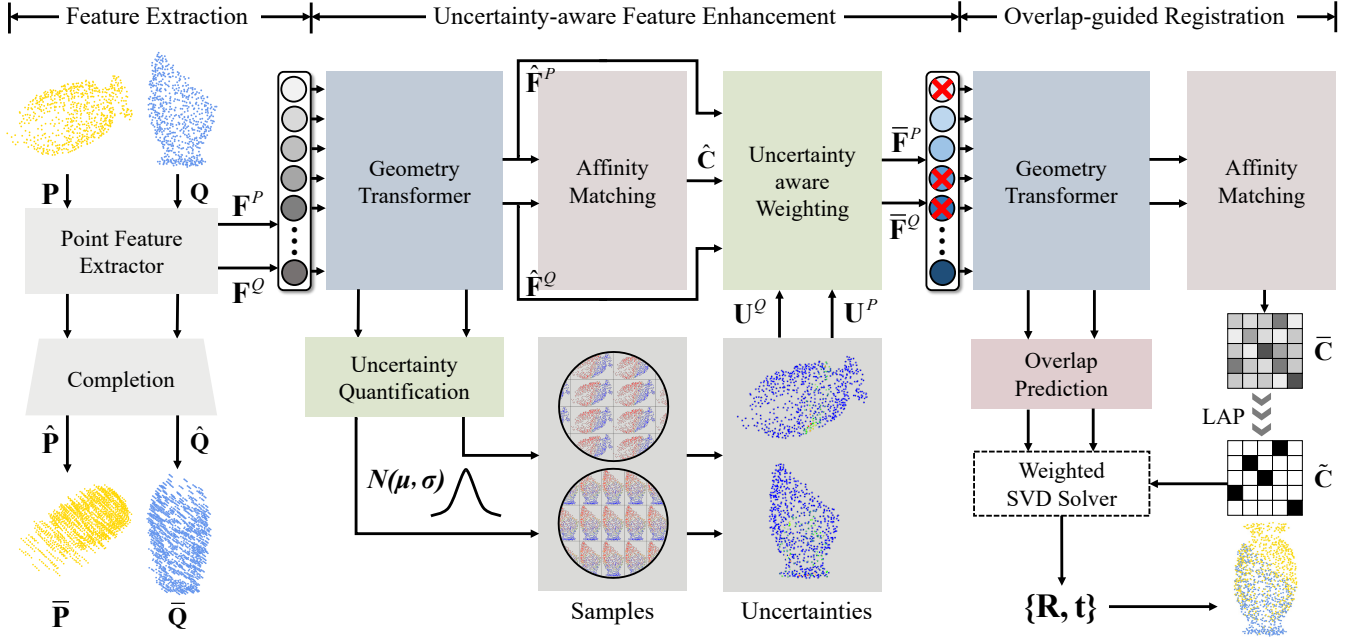


Figure 2: UTOPIC contains three parts: Feature Extraction, Uncertainty-aware Feature Enhancement and Overlap-guided Registration.

assignment to prevent ambiguity. RPM-Net [YL20] uses the differentiable Sinkhorn layer and annealing algorithm to get soft assignments of point correspondences and features. RGM [FLLW21] introduces a deep graph matching-based framework for registration, which takes both local geometry and graph topological structure into consideration. By contrast, some other methods such as [CDK20, PRG*20, BLZ*21, LKCP21, CSYT22] focus more on the removal of outlier correspondences. For example, DGR [CDK20] and 3DRegNet [PRG*20] utilize a classifier to predict a confidence value for each correspondence. [BLZ*21] incorporates spatial consistency for pruning outlier correspondences and [CSYT22] extends it with a new second-order spatial compatibility. Additionally, [LCL*21] and [QYW*22] adopt a coarse-to-fine strategy to better align indoor or large-scale outdoor point clouds. DeepPRO [LHF*21] directly predicts the point-wise location of the aligned point cloud to circumvent overlap prediction or keypoint detection. To deal with the low-overlap point clouds, [WYF*22] and [YL22] utilize overlap prediction module to improve the performance of partial registration, but they overlook the inherent uncertainty between overlapping and non-overlapping regions.

The correspondence-free methods extract global features for each point cloud and estimate the transformation with optimization algorithm or regression. PointNetLK [AGSL19] proposes to combine the deep feature networks and the conventional Lucas & Kanade optimization algorithm to tackle the registration problem. FMR [HMZ20] presents a feature-metric projection error to align point clouds, which does not need to search the correspondences so that the optimisation speed is fast. [YEK*20] leverages a probabilistic registration paradigm by formulating registration as the minimization of KL-divergence between two probability distribu-

tions modeled as mixtures of Gaussians. PCRNet [SLG*19] simply extracts the global features and regresses the transformation from the combined features. [YYH*21] and [LWF22] solve the point cloud registration and completion tasks simultaneously, which facilitate each other. In summary, the correspondence-free methods typically use the global features, weakening the importance of local structures. In addition, these methods require time-consuming iterative process to achieve promising results.

2.3. Uncertainty Modeling

Generally, there are two typical kinds of uncertainties one can model [DKD09]. Aleatoric uncertainty captures inherent noise from observations, which cannot be reduced even if more data are observed. On the other hand, epistemic uncertainty exists in the model parameters which captures our ignorance about which model generates our collected data. This kind of uncertainty can be explained, if given enough data. Several recent work [ZFD*20, YZL*21, LZB22] try to model aleatoric uncertainty by learning the distribution instead of a single fixed uncertainty value. UC-Net [ZFD*20] proposes probabilistic RGB-D saliency detection network via conditional variational autoencoders to model human annotation uncertainty. [YZL*21] utilizes a probabilistic representational model in combination with a transformer for camouflaged object detection. [LZB22] proposes a confidence estimation network to model the uncertainty, which is used to dynamically supervise the predicted camouflage maps. Inspired by these works, we propose an uncertainty quantification scheme to capture overlap uncertainty. To the best of our knowledge, this is the first attempt to explicitly introduce overlap uncertainty to point cloud registration.

3. Method

3.1. Problem Formulation and Overview

Given two partial point clouds $\mathbf{P} = \{\mathbf{p}_i \in \mathbb{R}^3 \mid i = 1, \dots, N\}$ and $\mathbf{Q} = \{\mathbf{q}_j \in \mathbb{R}^3 \mid j = 1, \dots, M\}$, point cloud registration aims at estimating a rigid transformation $\mathbf{T} = \{\mathbf{R}, \mathbf{t}\}$, where $\mathbf{R} \in \mathbf{SO}(3)$ and $\mathbf{t} \in \mathbb{R}^3$. For the simple cases with strict one-to-one correspondences between \mathbf{P} and \mathbf{Q} , the transformation parameters can be solved by

$$\min_{\mathbf{R}, \mathbf{t}} \sum_i^N \sum_j^M \mathbf{C}_{i,j} \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{q}_j\|_2^2, \quad (1)$$

where \mathbf{C} is the correspondence matrix with $\sum_j^M \mathbf{C}_{i,j} = 1, \forall i$, $\sum_i^N \mathbf{C}_{i,j} = 1, \forall j$, and $\mathbf{C}_{i,j} \in \{0, 1\}^{N \times M}, \forall i, j$. However, for most of the partial overlap scenes, there are no strict one-to-one correspondences. As a result, the correspondence matrix \mathbf{C} cannot satisfy the above constraints. To solve this problem, we also introduce slack variables in \mathbf{C} as [YL20] to convert inequality constraints back into equality constraints. Specifically, we add an additional row and column to \mathbf{C} for the points that have no correspondences, thus the new matrix $\tilde{\mathbf{C}} \in \mathbb{R}^{(N+1) \times (M+1)}$ can meet above three constraints.

Considering that the high-confidence overlap region is prone to provide more reliable correspondences, we further reformulate the problem of partial point cloud registration as

$$\min_{\mathbf{R}, \mathbf{t}} \sum_i^N \sum_j^M \tilde{\mathbf{C}}_{i,j} w_{i,j} \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{q}_j\|_2^2, \quad (2)$$

where $w_{i,j}$ is the weight computed based on the overlap scores, and $\tilde{\mathbf{C}}$ is the predicted correspondence matrix. We then propose an end-to-end neural network to predict both the reliable overlap scores and dense correspondences, based on which the rigid transformation is computed via the SVD solver.

Fig. 2 shows the overall pipeline of *UTOPIC*, including three key parts: feature extraction, uncertainty-aware feature enhancement, and overlap-guided registration. We first utilize the shared point feature extractor to encode each point in \mathbf{P} and \mathbf{Q} , followed by a completion decoder to implicitly enrich the learned feature with the prior shape knowledge. Next, we enhance the feature representation with two sub-modules: a geometry transformer is designed to enhance the global contextual information and promote the cross-feature fusion, and an uncertainty quantification scheme is applied to measure the overlap uncertainty for solving the ambiguous overlap prediction problem. Finally, the geometry transformer and affinity matching are adopted again to detect high-fidelity overlapping regions and accurate dense correspondences, which are then both leveraged to guide the recovery of the alignment transformation.

3.2. Feature Extraction

For each input point cloud, we extract multi-level point-wise feature, by using four AdaptConv [ZFF*21] layers (64, 64, 128, 256) followed by an MLP, as shown in Fig. 3. The learned point-wise feature of \mathbf{P} and \mathbf{Q} are represented as $\mathbf{F}^P \in \mathbb{R}^{|\mathbf{P}| \times V}$ and $\mathbf{F}^Q \in \mathbb{R}^{|\mathbf{Q}| \times V}$. V is the dimension of the output feature.

It is intuitive that if two partial point clouds can be recovered

to their complete versions, it will be easier to align them. Hence, we detach the completion decoder from PCN [YKH*18], for involving rich geometric shape knowledge into the point-wise feature extractor. The completion decoder produces coarse results $\hat{\mathbf{P}}$, $\hat{\mathbf{Q}}$ and fine results $\bar{\mathbf{P}}$, $\bar{\mathbf{Q}}$. During our early experiments, we find that it is not necessary to carefully design a completion model, like [WXH*21, XWL*21]. Even though the completion results are not perfect (see from the left bottom of Fig. 2), it still benefits our registration accuracy (see from the ablation study in Sec. 4.3). We only use the completion decoder to enhance the feature representation. Note also that this completion decoder is only used during the training stage, while removed during testing.

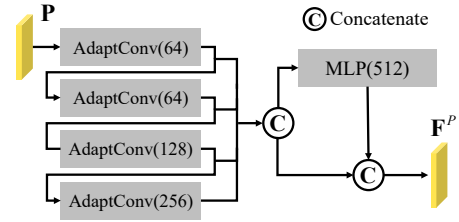


Figure 3: Detailed architecture of the point feature extractor.

3.3. Uncertainty-aware Feature Enhancement

3.3.1. Geometry Transformer

Transformer has been proved to be effective for capturing contextual information within a single point cloud or promoting the cross-feature fusion of point cloud pairs [WS19a, YLS*21, WYF*22, YL22]. Nevertheless, they only feed transformer with deep features and do not take the geometric information into consideration, which makes the learned features less discriminative. [LH22] introduces a position encoding method but merely using the coordinates of points. [QYW*22] proposes a geometric relative position embedding, yet with the large memory consumption. To this end, we elaborately design a geometry transformer that not only encodes sufficient geometry information but also occupies relatively low memory (see from Fig. 4). The proposed transformer consists of a geometry self-attention module and a feature cross-attention module. The two modules are alternately iterated for N times to obtain hybrid features $\hat{\mathbf{F}}^P \in \mathbb{R}^{|\mathbf{P}| \times V}$ and $\hat{\mathbf{F}}^Q \in \mathbb{R}^{|\mathbf{Q}| \times V}$.

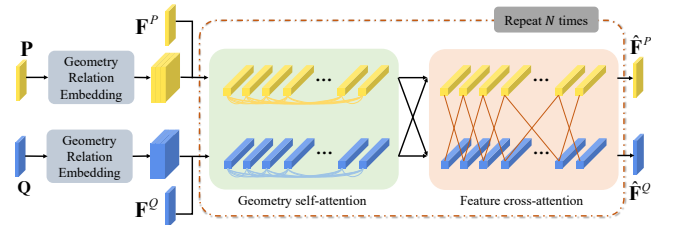


Figure 4: Detailed architecture of the geometry transformer.

Geometry self-attention. The geometry self-attention is designed to learn the global contextual information in both feature

and geometry spaces for each point cloud. We describe the computation for \mathbf{P} (the same for \mathbf{Q}). Given the features $\mathbf{F} \in \mathbb{R}^{|\mathbf{P}| \times d_t}$, where d_t is the dimension of the hidden feature, the output features $\mathbf{Z} \in \mathbb{R}^{|\mathbf{P}| \times d_t}$ are the weighted sum of all projected input features as

$$\mathbf{z}_i = \sum_{j=1}^{|\mathbf{P}|} a_{i,j} (\mathbf{f}_j \mathbf{W}^V), \quad (3)$$

where $a_{i,j}$ is the weight coefficient and computed by a row-wise softmax on the attention scores $e_{i,j}$, which is computed as

$$e_{i,j} = \frac{(\mathbf{f}_i \mathbf{W}^Q) (\mathbf{f}_j \mathbf{W}^K)^T + \mathbf{g}_{i,j} \mathbf{W}^G}{\sqrt{d_t}}, \quad (4)$$

where $\mathbf{g}_{i,j}$ is the geometric relation embedding, $\mathbf{W}^Q, \mathbf{W}^K$, and $\mathbf{W}^V \in \mathbb{R}^{d_t \times d_t}$ are the projection weights for queries, keys and values. $\mathbf{W}^G \in \mathbb{R}^{3 \times 1}$ is the projection weights for geometric relation embedding. The right part of Fig. 5 shows the computation of the geometry self-attention. Note that [QYW*22] also embeds geometry structure knowledge into the transformer, but with the extra space complexity of $O(N^2 \times d_t)$. By contrast, our geometry self-attention only costs $O(N^2 \times 3)$. Usually, the feature dimension d_t is far larger than 3, our solution therefore has less space complexity.

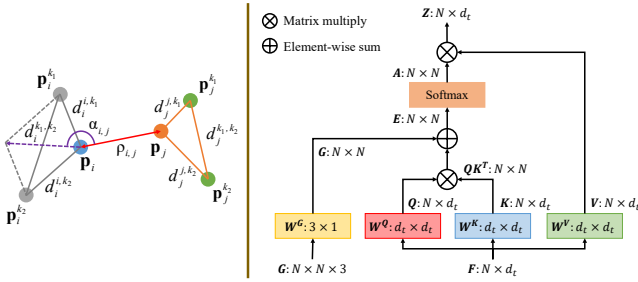


Figure 5: Left: An illustration of the geometric relation embedding. Right: Computation flow of the geometry self-attention.

Geometric relation embedding. We first define the point triplet as the group of a target point associated with its two nearest neighbors. Inspired by the handcrafted features, we also define a novel geometric relation embedding with point-wise distance, triplet-wise angle and the difference between the perimeter of the local triangles. In detail, given two points \mathbf{p}_i and \mathbf{p}_j , their geometric relation can be described as

(1) *Point-wise Distance.* Point-wise distance is the distance $\rho_{i,j} = \|\mathbf{p}_i - \mathbf{p}_j\|_2$ between two points in the Euclidean space.

(2) *Triplet-wise Angle.* We search two nearest neighbors $\mathbf{p}_i^{k_1}, \mathbf{p}_i^{k_2}$ for \mathbf{p}_i , and form a triplet. Then, we calculate the sum of vector $\mathbf{p}_i^{k_1} - \mathbf{p}_i$ and $\mathbf{p}_i^{k_2} - \mathbf{p}_i$, denoted as $\mathbf{p}_i^k - \mathbf{p}_i$. The triplet-wise angle is computed as $\alpha_{i,j} = \angle(\mathbf{p}_i^k - \mathbf{p}_i, \mathbf{p}_j - \mathbf{p}_i)$.

(3) *Triangle Perimeter Difference.* We also denote the two nearest neighbors of \mathbf{p}_j as $\mathbf{p}_j^{k_1}$ and $\mathbf{p}_j^{k_2}$, which form a local triangle. As is shown in the left part of Fig. 5, we compute the perimeter dif-

ference between local triangles as $\eta_{i,j} = (d_i^{i,k_1} + d_i^{i,k_2} + d_i^{k_1,k_2}) - (d_j^{j,k_1} + d_j^{j,k_2} + d_j^{k_1,k_2})$, where $d_i^{i,k_1} = \|\mathbf{p}_i - \mathbf{p}_i^{k_1}\|_2$.

Finally, the geometric relation embedding $\mathbf{g}_{i,j}$ is computed by aggregating the point-wise distance, triplet-wise angle and the triangle perimeter difference as

$$\mathbf{g}_{i,j} = \text{CAT}[\rho_{i,j}, \alpha_{i,j}, \eta_{i,j}], \quad (5)$$

where $\text{CAT}[\cdot, \cdot]$ represents the concatenation operation.

Feature cross-attention. Cross-attention is essential for the feature interaction between two point clouds. Given the features $\mathbf{F}^P, \mathbf{F}^Q$ of \mathbf{P}, \mathbf{Q} , the cross-attention features \mathbf{Z}^P of \mathbf{P} is computed as

$$\mathbf{z}_i^P = \sum_{j=1}^{|\mathbf{Q}|} a_{i,j} (\mathbf{f}_j^Q \mathbf{W}^V). \quad (6)$$

Similarly, $a_{i,j}$ is computed by a row-wise softmax on the cross-attention scores $e_{i,j}$ as

$$e_{i,j} = \frac{(\mathbf{f}_i^P \mathbf{W}^Q) (\mathbf{f}_j^Q \mathbf{W}^K)^T}{\sqrt{d_t}}. \quad (7)$$

The cross-attention feature of \mathbf{Q} is computed in the same way. The geometry self-attention module encodes the transformation-invariant geometric relation embedding for each point cloud, while the feature cross-attention module encourages the feature interaction conditioned on each other. Therefore, the final feature representation is more robust and discriminative.

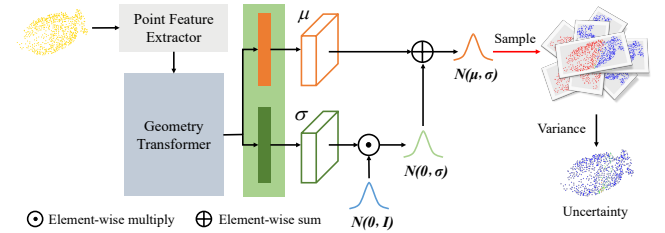


Figure 6: Overview of the uncertainty quantification module. This module decomposes the sample operation into a trainable part and a random part for end-to-end learning.

3.3.2. Uncertainty Quantification

In the partial overlap scenes, it is not trivial to find the one-to-one correspondence. Outlier points may lead to many mismatches. To resolve this problem, recent literature [HGU*21, XLW*21, WYF*22, YL22] is devoted to better feature representation and reliable overlap predictions. However, these approaches neglect the uncertainty that inherently exists between overlapping and non-overlapping regions and in some other challenging regions, especially when the overlap ratio is low or there exist noise and outliers.

Motivated by this issue, we propose a probability-based overlap uncertainty quantification scheme to generate more discriminative features. In detail, for the input point cloud \mathbf{P} , we assign each point

a Gaussian distribution parameterized by the mean μ_i and the variance σ_i , which are predicted by different MLP layers from $\hat{\mathbf{F}}^P$. The overlap score of \mathbf{p}_i can be then sampled from the learned distribution, $o_i \sim \mathcal{N}(\mu_i, \sigma_i)$. Following [YZL*21], we decompose the sampling operation into two steps (see from Fig. 6). First, one random sample ε_i is obtained from the standard Gaussian distribution $\mathcal{N}(0, I)$, i.e., $\varepsilon_i \sim \mathcal{N}(0, I)$, and the desired sample is computed by $\mu_i + \varepsilon_i \sigma_i$. To measure point-wise overlap uncertainty, we then re-sample K overlap scores for each point from the learned distribution, denoted as $O^P = \{o^{(1)}, \dots, o^{(K)}\}$. According to Bayesian probability theory, we can treat O^P as empirical samples from an approximate predictive distribution and measure the uncertainty by calculating the variance as

$$\mathbf{U}^P = \text{norm}(\text{var}(O^P)), \quad (8)$$

where $\mathbf{U}^P \in \mathbb{R}^{|\mathbf{P}| \times 1}$ represents the overlap uncertainty, $\text{norm}(\cdot)$ is the min-max normalization and $\text{var}(\cdot)$ is the variance computation. Also, the overlap uncertainty \mathbf{U}^Q can be obtained by the same way.

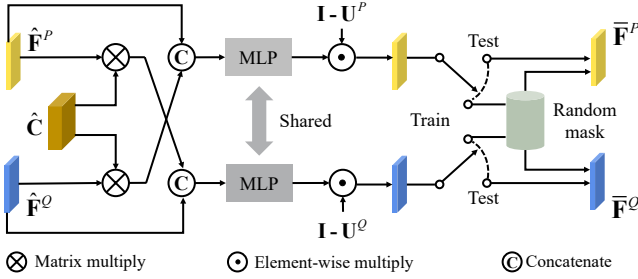


Figure 7: The details of the uncertainty-aware weighting module.

3.3.3. Uncertainty-aware Weighting

Once obtaining the overlap uncertainty, the learned point-wise features are re-weighted to highlight the regions of low uncertainties. As illustrated in Fig. 7, we take the features $\hat{\mathbf{F}}^P, \hat{\mathbf{F}}^Q$, the correspondence $\hat{\mathbf{C}}$, and the overlap uncertainty $\mathbf{U}^P (\mathbf{U}^Q)$ as inputs to compute the uncertainty-aware features $\tilde{\mathbf{F}}^P (\tilde{\mathbf{F}}^Q)$ as

$$\tilde{\mathbf{F}}^P = \text{MLP}(\text{cat}[\hat{\mathbf{F}}^P, \hat{\mathbf{C}} \otimes \hat{\mathbf{F}}^Q]), \quad (9)$$

$$\tilde{\mathbf{F}}^P = \tilde{\mathbf{F}}^P \odot (\mathbf{I} - \mathbf{U}^P), \quad (10)$$

where $\text{MLP}(\cdot)$ denotes a multi-layer perception. Although this module is designed to lower the importance of features from the uncertain regions, the network are expected to pay more attention to these regions. We take an uncertainty-based random masking strategy, as demonstrated in the right-most part of Fig. 7. During training stage, we first assign a random number to each point, and then mask out those features whose associated overlap uncertainty is larger than the random value. The uncertainty-based random masking strategy increases the difficulty and diversity of training samples. As a result, the uncertainty information can be incorporated into the training procedure. This motivates the network to focus more on the learning of these uncertain points. Note that the random masking strategy is removed during testing stage.

Affinity matching. Notably, in the process of the above uncertainty-aware weighting, a soft correspondence $\tilde{\mathbf{C}}$ is required. An affinity matching module is used to generate it. We calculate the similarity matrix of features $\tilde{\mathbf{F}}^P, \tilde{\mathbf{F}}^Q$ in a learnable way instead of simple dot-product of feature vector, which can be represented as

$$\mathbf{A}_{i,j} = (\tilde{\mathbf{f}}_i^P)^T \mathbf{W} (\tilde{\mathbf{f}}_j^Q), \quad (11)$$

where $\mathbf{W} \in \mathbb{R}^{V \times V}$ is the learnable parameter. For handling outliers and recovering equality constraints of the correspondences matrix, we add a slack variable to row and column of the similarity matrix, and then leverage Sinkhorn [Sin64] algorithm to calculate the soft correspondence matrix $\tilde{\mathbf{C}}$. More details please refer to [YL20].

3.4. Overlap-guided Registration

As previously investigated, approaches [HGU*21, XLW*21, WYF*22, YL22] present different strategies to improve the performance of registration with overlap scores. For example, Predator [HGU*21] samples more interest points from overlapping regions for RANSAC. [WYF*22] applies gumbel softmax to sample the overlap correspondences. OMNet [XLW*21] utilizes overlap masks to highlight the points within the overlap regions. We argue that only using the predicted overlap points to estimate the rigid transformation is unreliable for the reason that sparse correspondences from these points may be insufficient and confused due to repetitive structures, noise and outliers [HMZA21]. Hence, we propose an overlap-guided registration module to tackle this problem.

With the uncertainty-aware features $\tilde{\mathbf{F}}^P, \tilde{\mathbf{F}}^Q$, another geometry transformer is adopted to predict the reliable overlap scores, $\hat{\mathbf{O}}^P$ and $\hat{\mathbf{O}}^Q$. Meanwhile, the final soft correspondence $\tilde{\mathbf{C}}$ is also obtained by an affinity matching module. We then leverage the linear assignment problem (LAP) solver [JV87] to compute the binary hard correspondence $\tilde{\mathbf{C}}$ from $\tilde{\mathbf{C}}$. Based on observation that the correspondence is more likely to be accurate if a pair of corresponding points also both lie in the overlapping region, we add this weight for each correspondence as:

$$w_{i,j} = \begin{cases} \frac{\hat{o}_i^P \hat{o}_j^Q}{\sum_i \sum_j \tilde{\mathbf{C}}_{i,j} \hat{o}_i^P \hat{o}_j^Q}, & \tilde{\mathbf{C}}_{i,j} = 1 \\ 0, & \tilde{\mathbf{C}}_{i,j} = 0 \end{cases}. \quad (12)$$

Finally, the rigid transformation is easily computed by SVD, according to Eq. 2.

3.5. Loss Function

The proposed *UTOPIC* is trained in an end-to-end way without iterative steps, using four loss terms $\mathcal{L} = \mathcal{L}_r + \mathcal{L}_o + \mathcal{L}_u + \mathcal{L}_c$.

Registration loss. We adopt the cross entropy loss between the predicted soft correspondence $\tilde{\mathbf{C}}$ and the ground truth correspondence \mathbf{C} to train our model. The formula is as

$$\mathcal{L}_r = - \sum_i \sum_j (C_{i,j} \log \tilde{\mathbf{C}}_{i,j} + (1 - C_{i,j}) \log (1 - \tilde{\mathbf{C}}_{i,j})). \quad (13)$$

The ground-truth correspondence \mathbf{C} is computed by mutual near-

Table 1: Quantitative comparison of different methods on ModelNet, ModelLoNet and ScanObjectNN. The best performance is highlighted in **bold**, and the sub-optimal performance is marked by underline.

Dataset	Method	RMSE(R)	MAE(R)	RMSE(t)	MAE(t)	Error(R)	Error(t)
ModelNet [WSK*15]	IDAM [LZX*20]	10.8991	7.4032	0.14258	0.09009	14.2849	0.19260
	RPM-Net [YL20]	2.5491	0.8849	0.01984	0.00865	1.6914	0.01830
	OMNet [XLW*21]	8.2406	5.8538	0.09341	0.06161	11.4084	0.12982
	RGM [FLLW21]	3.0068	0.8072	0.02556	0.00699	1.5610	0.01458
	Predator [HGU*21]	<u>0.6505</u>	0.4609	0.01148	0.00447	0.8643	0.00904
	REGTR [YL22]	1.1690	0.1894	<u>0.00434</u>	<u>0.00153</u>	<u>0.3493</u>	<u>0.00306</u>
	Ours	0.2238	0.1322	0.00212	0.00131	0.2464	0.00264
ModelLoNet [WSK*15, HGU*21]	IDAM [LZX*20]	16.9098	11.7129	0.23770	0.15980	22.6078	0.34240
	RPM-Net [YL20]	6.6189	2.0453	0.04813	0.01827	3.8514	0.03835
	OMNet [XLW*21]	10.2776	7.7237	0.12428	0.08294	15.0794	0.17329
	RGM [FLLW21]	17.8005	5.5261	0.10970	0.04025	9.8361	0.08408
	Predator [HGU*21]	16.0141	3.3206	0.12510	0.03281	5.8900	0.07673
	REGTR [YL22]	3.4594	<u>1.3427</u>	<u>0.03216</u>	<u>0.01254</u>	<u>2.5753</u>	<u>0.02617</u>
	Ours	2.5463	0.8588	0.02098	0.00765	1.5934	0.01555
ScanObjectNN [UPH*19]	IDAM [LZX*20]	15.9602	8.0429	0.13187	0.07255	15.2581	0.15084
	RPM-Net [YL20]	8.1251	2.8829	0.08984	0.03314	5.4576	0.07384
	OMNet [XLW*21]	10.5122	7.9538	0.11367	0.07984	15.2959	0.16384
	RGM [FLLW21]	12.6069	3.7596	0.08729	0.02740	7.0376	0.05634
	Predator [HGU*21]	12.6050	4.8842	0.12480	0.04899	9.4936	0.10070
	REGTR [YL22]	2.1477	<u>0.7448</u>	<u>0.02268</u>	<u>0.00614</u>	<u>1.3842</u>	<u>0.01247</u>
	Ours	<u>3.6790</u>	0.2633	0.01394	0.00179	0.5601	0.00366

est neighbour searching as

$$\mathbf{C}_{i,j} = \begin{cases} 1, & \mathcal{NN}(\mathbf{q}_j, \mathcal{T}(\mathbf{P})) = \mathbf{p}_i, \mathcal{NN}(\mathcal{T}(\mathbf{p}_i), \mathbf{Q}) = \mathbf{q}_j \\ & \text{and } \|\mathcal{T}(\mathbf{p}_i) - \mathbf{q}_j\|_2 < r \\ 0, & \text{otherwise} \end{cases}, \quad (14)$$

where $\mathcal{NN}(\cdot)$ denotes the spatial nearest neighbor, $\mathcal{T}(\cdot)$ is the operation of the ground-truth rigid transformation from \mathbf{P} to \mathbf{Q} , and r denotes the distance threshold.

Overlap loss. The task of predicting overlap scores is regarded as a binary classification problem and supervised as

$$\mathcal{L}_o^P = \frac{1}{|\mathbf{P}|} \sum_{i=1}^{|\mathbf{P}|} \left(\bar{\mathbf{o}}_i^P \log(\hat{\mathbf{o}}_i^P) + (1 - \bar{\mathbf{o}}_i^P) \log(1 - \hat{\mathbf{o}}_i^P) \right), \quad (15)$$

where $\bar{\mathbf{O}}^P$ denotes the ground-truth overlap of \mathbf{P} as

$$\bar{\mathbf{o}}_i^P = \begin{cases} 1, & \sum_j^M \mathbf{C}_{i,j} = 1 \\ 0, & \text{otherwise} \end{cases}. \quad (16)$$

The overlap loss \mathcal{L}_o^Q can be computed in the same way. We then get a total overlap loss: $\mathcal{L}_o = \mathcal{L}_o^P + \mathcal{L}_o^Q$.

Uncertainty loss. To train the uncertainty quantification module, we define the uncertainty loss \mathcal{L}_u , which is a weighted combination of a standard binary cross entropy (BCE) loss and a Kullback-Leibler (KL) divergence as

$$\mathcal{L}_u = \lambda \cdot \mathcal{L}_{BCE} \left(o^{(k)}, \bar{\mathbf{O}}^P \right) + \eta \cdot KL(\mathcal{N}(\mu, \sigma) \| \mathcal{N}(0, I)), \quad (17)$$

where $o^{(k)}$ denotes one example drawn from the overlap distribution, which is used to boost the diversity. λ and η are balance factors. The final uncertainty loss is $\mathcal{L}_u = \mathcal{L}_u^P + \mathcal{L}_u^Q$.

Completion loss. Following PCN [YKH*18], we use the chamfer distance (CD) to supervise the coarse and fine completion results as

$$\mathcal{L}_c^P = CD(\hat{\mathbf{P}}, \tilde{\mathbf{P}}) + CD(\bar{\mathbf{P}}, \tilde{\mathbf{P}}), \quad (18)$$

where $\tilde{\mathbf{P}}$ is the ground truth completion result. The total completion loss can be computed as $\mathcal{L}_c = \mathcal{L}_c^P + \mathcal{L}_c^Q$

4. Experiment

Datasets. We evaluate our approach on three datasets: ModelNet, ModelLoNet, and ScanObjectNN.

(i) ModelNet is generated from ModelNet40 [WSK*15], which contains synthetic point clouds sampled from 12,311 CAD models of 40 different categories. We sample 1024 points for each point cloud, and then adopt the same cropping setting in [YL20] to simulate the partial-overlap point cloud pairs. Then, one of the point cloud is transformed by a random rotation in the range of $[0, 45]^\circ$ and a random translation \mathbf{t} in the range of $[-0.5, 0.5]$ along each axis. Gaussian noise sampled from $\mathcal{N}(0, 0.01)$ and clipped to $[-0.05, 0.05]$ is also added to each coordinate of the points in the clean point clouds. Finally, each point cloud is shuffled to reorder all points. To evaluate the ability of our network to generalize to different shapes, we use the first 20 categories for training (5,112 samples) and validation (1,202 samples) respectively, and the remaining for testing (1,266 samples).

(ii) ModelLoNet is also generated from ModelNet40. The only difference from ModelNet is that ModelLoNet has an average overlap ratio of 53.6%, while ModelNet is 73.5%.

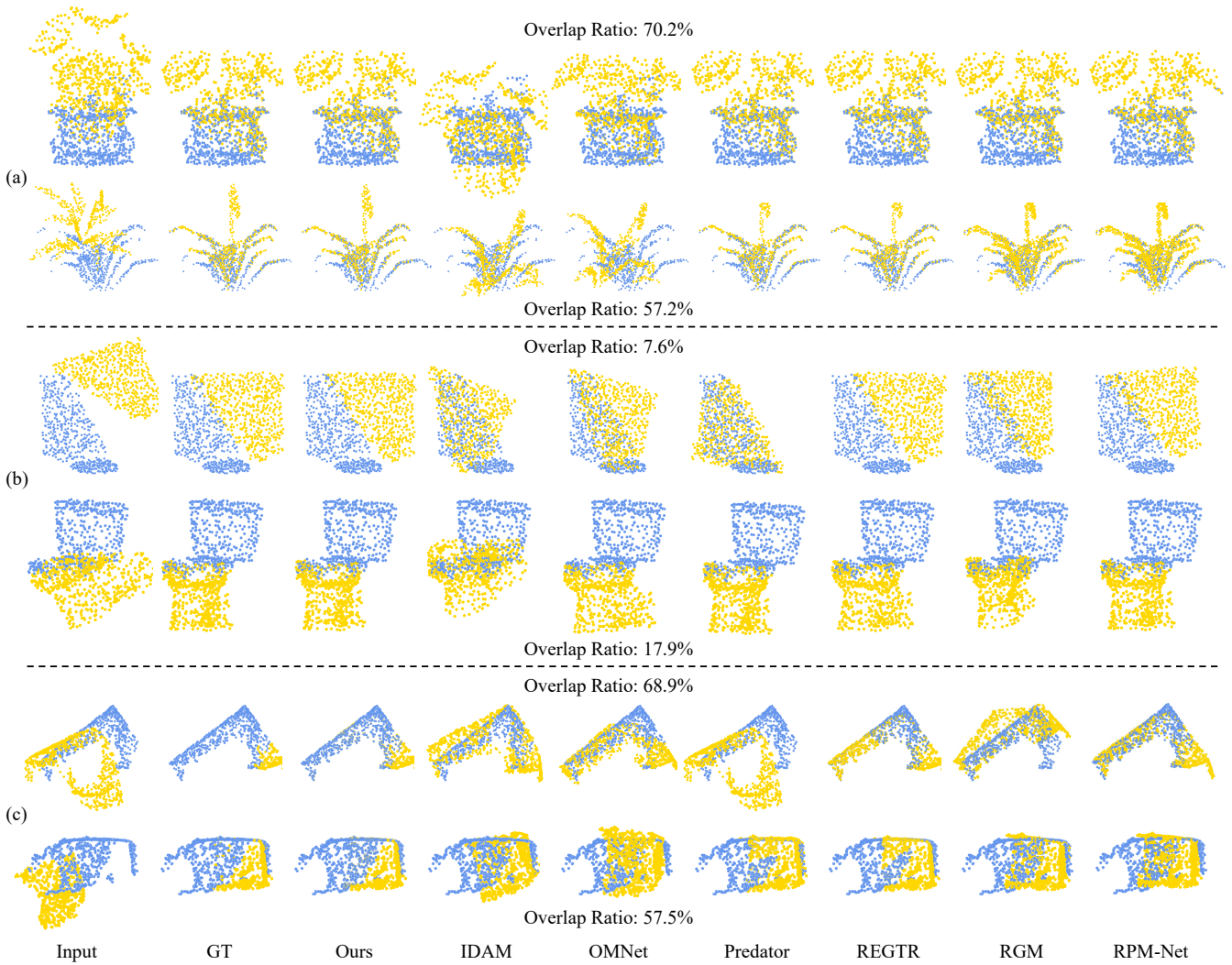


Figure 8: Comparison of registration results of six SOTA methods and *UTOPIC* on (a) ModelNet, (b) ModelLoNet, and (c) ScanObjectNN.

(iii) ScanObjectNN [UPH*19] is a real-world point cloud dataset based on scanned indoor scene data. We use the testing part of ScanObjectNN containing 581 samples. We do not retrain all compared network models. This dataset is only used for testing.

Evaluation metrics. For the consistency with prior work [WS19a], we use the anisotropic metrics of Root Mean Square Error (RMSE) and Mean Average Error (MAE) over Euler angles and translations. Besides, we evaluate the isotropic error for rotation and translation proposed in [YL20]. All angles are in degrees.

Comparison methods. We compare our method with six latest learning-based registration approaches [LZX*20, YL20, XLW*21, FLLW21, HGU*21, YL22]. OMNet [XLW*21], Predator [HGU*21] and REGTR [YL22] also utilize the overlap prediction. For all compared methods, we use their released public code and follow the same setting in their original papers to retrain the networks by our prepared training data.

Implementation details. All experiments are conducted on a single Nvidia RTX 2080Ti. We train *UTOPIC* for 200 epochs with a batch size of 4. The SGD optimizer is used with the initial learning rate of 0.001. The sampling number K is set to 50. The balance factors λ, η are 0.5 and 0.1. The repeat number N in geometry transformer is set to 3, and the distance threshold r is 0.075. The code is available at <https://github.com/ZhileiChen99/UTOPIC>.

4.1. Evaluation on Synthetic ModelNet and ModelLoNet

We quantitatively evaluate the effectiveness of our method on two synthetic datasets. As reported in Tab. 1, our approach achieves the smallest registration errors over all metrics compared with other approaches. REGTR [YL22] and Predator [HGU*21] also have impressive results thanks to their overlap prediction mechanisms. When evaluating on the dataset with lower overlap ratio, it is obvious that the performance of most methods degrades dramatically

except our approach. In terms of the mean absolute error of rotation in Tab. 1, the margin between *UTOPIC* and REGTR [YL22] on the ModelNet is even ten times larger than that on the ModelNet. Fig. 8 visualizes several cases that are unseen shapes for all neural networks. We can observe that *UTOPIC* achieves the most satisfactory registration results for partial point clouds of noise and sampling irregularity. We can also observe from Fig. 8 (b) that, our method still produces quality alignments, even though the two input pairs contain rather limited overlapping parts.

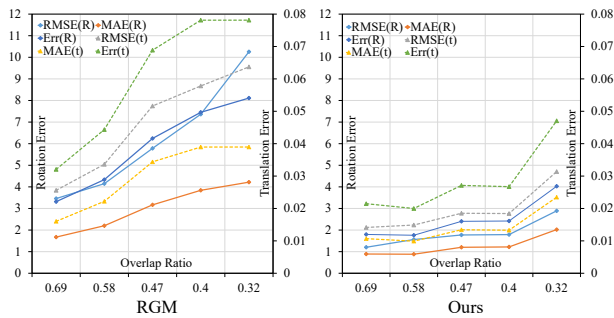


Figure 9: Error statistics of RGM [FLLW21] and our method under different overlap ratios.

4.2. Evaluation on ScanObjectNN

We also evaluate the generalization ability of our method on the ScanObjectNN dataset. The employed models for comparison are all trained on the ModelNet. As shown in the Tab. 1, the proposed *UTOPIC* produces almost the best results, which demonstrates a better generalization ability to real-scanned data. Nevertheless, the state-of-the-art methods like [HGU*21, FLLW21] are less robust to the data domain gap. Qualitative comparisons are demonstrated in Fig. 8, which are consistent with the quantitative statistics.

Table 2: Ablation studies on ModelNet. **O**, **C**, **U** and **G** represent models with overlap, completion, uncertainty and geometric relation embedding. The best performance is highlighted in **bold**.

O	C	U	G	RMSE(R)	MAE(R)	RMSE(t)	MAE(t)	Error(R)	Error(t)
				0.3682	0.1907	0.00347	0.00195	0.3495	0.00397
✓				0.2837	0.1481	0.00241	0.00149	0.2761	0.00300
✓	✓			0.2735	0.1470	0.00238	0.00138	0.2689	0.00277
✓	✓	✓		0.2691	0.1404	0.00237	0.00136	0.2600	0.00273
✓	✓	✓	✓	0.2238	0.1322	0.00212	0.00131	0.2464	0.00264

4.3. Analysis

In this section, we conduct various specified experiments to analyze the main ideas of *UTOPIC*.

Ablation study. We perform ablation studies on the ModelNet dataset for a better understanding of our four main components: (i) the overlap-guided weights (**O**), (ii) the completion decoder (**C**), (iii) the uncertainty quantification (**U**), and (iv) the geometric relation embedding (**G**). The detailed results are recorded in Tab. 2. Clearly, our full pipeline performs the best on all metrics. Removing any component degrades the overall performance, suggesting

that all components are beneficial for accurate registration results. In particular, we observe great improvements for the variant with the overlap-guided weights. This implies that overlap prediction is beneficial for partial point cloud registration. This observation also implies that the importance of uncertainty quantification for addressing the ambiguous overlap prediction problem.

Different overlap ratios. Although the comparisons on ModelNet have indicated the superiority of our *UTOPIC* for low-overlap point cloud pairs, it is still meaningful to analyse the variation tendency of the performance when overlap ratio decreases gradually. We compare the performance of our approach with RGM [FLLW21] on ModelNet40. We use the same crop setting of PR-Net [WS19b]. The number of points is set to 768, 700, 640, 600 and 560 to generate point clouds with approximate overlap ratios of 0.69, 0.58, 0.47, 0.40, and 0.32, respectively. Fig. 9 shows the registration errors for different overlap ratios. As observed, our approach is very stable until the overlap ratio decreases to 0.32, but still better than RGM [FLLW21].

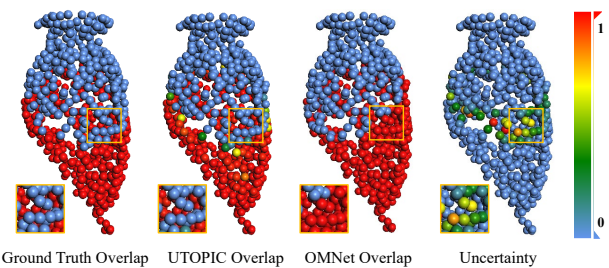


Figure 10: Visualization of predicted overlap scores and uncertainty. We colorize both of the overlap scores and the uncertainties based on the right color scale. With the color redder, one point is more likely to be an overlapping or uncertain point.

Visualization of predicted overlap scores and uncertainty. We visualize the predicted overlap scores and the uncertainty in Fig. 10. With the color redder, one point is more likely to be an overlapping or uncertain point. By comparing the predicted overlap scores between OMNet [XLW*21] and our method and the ground-truth result, it is obvious to see that the detected overlapping regions of our method are more accurate. OMNet [XLW*21] mistakes some ambiguous points and produces a wrong overlap mask. Besides, the redder points in the rightmost sub-figure are the points with higher uncertainty. They are almost located at the boundary between the overlapping regions and non-overlapping regions. This is consistent with our design intuition: precise uncertainty quantification facilitates better overlap prediction (Fig. 10 (b)).

Table 3: Error statistics of different self-attention modules.

UTOPIC	RMSE(R)	MAE(R)	RMSE(t)	MAE(t)	Error(R)	Error(t)
vanilla self-attention	0.2691	0.1404	0.00237	0.00136	0.2600	0.00273
self-attention w/APE	0.2667	0.1466	0.00256	0.00149	0.2718	0.00299
self-attention w/LPE	0.2656	0.1400	0.00226	0.00132	0.2567	0.00266
geometry self-attention	0.2238	0.1322	0.00212	0.00131	0.2464	0.00264

Different self-attention modules. To testify the effectiveness of the geometry self-attention, we compare four methods in Tab. 3:

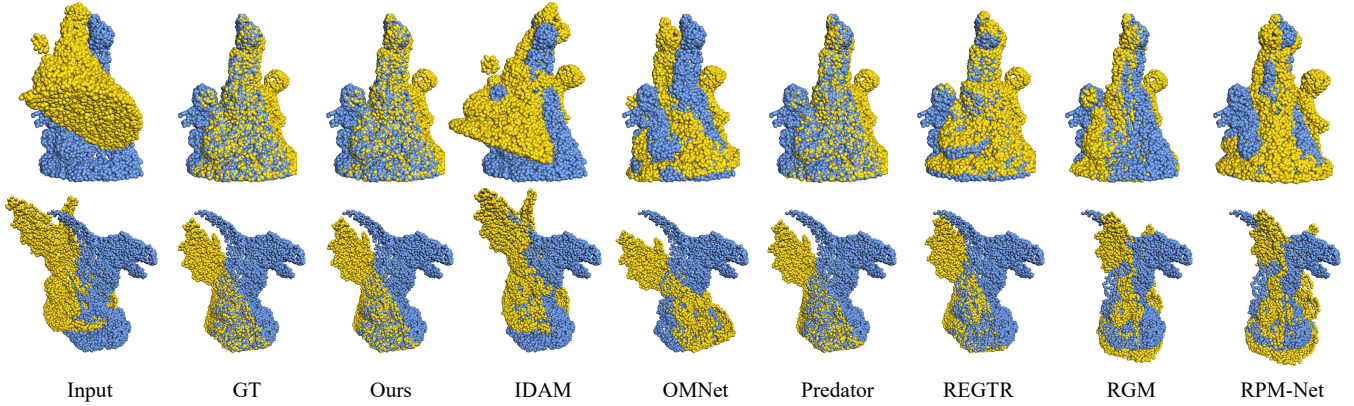


Figure 11: Comparison of registration results on two unseen complex shapes of tiny details from PU-GAN [LLF*19].

(i) vanilla self-attention [YLS*21], (ii) absolute position embedding (APE) [SDMR20], (iii) learned position embedding (LPE) [CWY*21], and (iv) geometric relation embedding. We observe that the learned position embedding slightly improves the performance, while the absolute position embedding causes degradation. This is because low dimensional coordinate information without non-linear mapping is not consistent with the high dimensional features. By contrast, our approach achieves the best performance.

Table 4: Analysis of different uncertainty schemes.

UTOPIC	RMSE(R)	MAE(R)	RMSE(t)	MAE(t)	Error(R)	Error(t)	OA
w/ PU, RM	0.3561	0.1562	0.00295	0.00152	0.2852	0.00309	82.51%
w/o U, RM	0.3017	0.1547	0.00259	0.00149	0.2791	0.00300	91.19%
w/o RM	0.2612	0.1466	0.00230	0.00142	0.2713	0.00286	95.31%
w/ U, RM	0.2238	0.1322	0.00212	0.00131	0.2464	0.00264	96.46%

Analysis of different uncertainty schemes. We conduct several experiments to elaborate the crucial role of uncertainty schemes. As shown in Tab. 4, we regard *UTOPIC* as the baseline, and compare four variants: (i) with predicted uncertainty (**PU**) and random masking (**RM**), but the variance in Eq. 8 is replaced by the predicted σ_i of the Gaussian distribution, (ii) without uncertainty (**U**) and random masking (**RM**), (iii) without random masking (**RM**), (iv) with uncertainty (**U**) and random masking (**RM**). The overlap accuracy is reported as **OA** in all experiments. The first row in Tab. 4 indicates that the registration results of this variant degrade dramatically, which shows the necessity of the computation for uncertainty in Eq. 8. This is because the predicted variance may not accurately describe the real variance of the Gaussian distribution, while the sample variance can approximate the real variance more reliably. To demonstrate the importance of uncertainty for overlap prediction as our initial motivation, we remove the uncertainty and random masking from *UTOPIC* and compute the overlap accuracy. We find that the variant without uncertainty only achieves an overlap accuracy of 91.19% on average, while the variant with uncertainty achieves 96.46%. Comparing with *UTOPIC*, the Error(**R**) of the variant without random masking degrades from 0.2464 to 0.2713 and the Error(**t**) degrades from 0.00264 to 0.00286, which demonstrates the benefits of the random masking strategy.

Unseen complex 3D shapes with tiny geometry details. We additionally present the visual comparisons on two unseen complex 3D shapes with tiny geometry details. The inputs are sampled with 5000 points from the dataset in [LLF*19]. Notably, all compared methods are trained on the ModelNet. As shown in Fig. 11, the sculpture and gargoyle are more geometrically complicated than the models in ModelNet40, but our approach also works well.

Table 5: Timing statistics in seconds for different approaches performed on ModelNet and ModelLoNet. N is the number of input points. We report the time for one pair of point clouds.

Method	ModelNet ($N:717$)	ModelLoNet ($N:512$)
IDAM [LZX*20]	0.07	0.07
RPM-Net [YL20]	0.16	0.13
OMNet [XLW*21]	0.03	0.03
RGM [FLLW21]	0.17	0.14
Predator [HGU*21]	0.29	0.23
REGTR [YL22]	0.05	0.05
Ours	0.06	0.06

Timing. Table 5 reports the average running time (in seconds) of different approaches. The testing data is collected from the datasets of ModelNet and ModelLoNet. We conduct the test on a single Nvidia RTX 2080Ti with Intel Core i7-4790 @ 3.6GHz. We find that only OMNet [XLW*21] and REGTR [YL22] are faster than our method, while their performance is less satisfactory.

5. Failure Cases and Limitations.

Despite the promising performance of our approach, it still has some limitations. First, if the distribution of the overlapping points is sparse and scattered, it is hard for our approach to align point clouds well (see from the first row of Fig. 12). This is because clustering points in the same overlapping areas can help generate high-certainty overlapping points. Second, our method may fail to register the point clouds that have no distinctive geometry structures (see from the bottom row in Fig. 12). This is because geometrically indistinguishable features lead to confusing mismatches. Finally, our

approach does not conduct any downsampling operation. This hinders our method being directly adapted to large-scale point clouds.

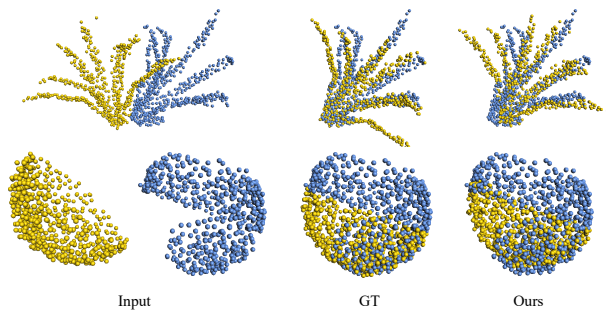


Figure 12: Two failure examples. *UTOPIC* fails when: (i) the distributions of overlap regions are sparse and scattered (Row 1); (ii) the inputs lack geometrically indistinguishable features (Row 2).

6. Conclusion

We present *UTOPIC*, a novel uncertainty-aware overlap prediction network for partial point cloud registration. *UTOPIC* utilizes the overlap uncertainty quantification scheme for the first time to solve the ambiguous overlap prediction problem. Through the completion decoder and geometric relation embedding, our method captures rich features of shape knowledge and transformation-invariant geometry information. Thanks to the reliable overlap scores and exact dense correspondences, *UTOPIC* aligns point clouds in high accuracy, even handling low-overlap or noisy shapes. In the future, we will try to extend our method to tackle more challenging cases of large-scale (e.g., LiDAR data) point cloud registration.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62172218, No. 62032011), the Free Exploration of Basic Research Project, Local Science and Technology Development Fund Guided by the Central Government of China (No. 2021Szvup060), the Natural Science Foundation of Guangdong Province (No. 2022A1515010170), the Innovation and Technology Fund - Midstream Research Programme for Universities from Hong Kong Innovation and Technology Commission (No. MRP/022/20X), and the General Research Fund from Hong Kong Research Grants Council (No. 15218521).

References

- [AGSL19] AOKI Y., GOFORTH H., SRIVATSAN R. A., LUCEY S.: Pointnetlk: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 7163–7172. 3
- [AHY*21] AO S., HU Q., YANG B., MARKHAM A., GUO Y.: Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 11753–11762. 2
- [BLZ*21] BAI X., LUO Z., ZHOU L., CHEN H., LI L., HU Z., FU H., TAI C.-L.: Pointdsc: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 15859–15869. 3

- [BM92] BESL P. J., MCKAY N. D.: Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures* (1992), vol. 1611, Spie, pp. 586–606. 2
- [CDK20] CHOY C., DONG W., KOLTUN V.: Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 2514–2523. 3
- [CPK19] CHOY C., PARK J., KOLTUN V.: Fully convolutional geometric features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 8958–8966. 2
- [CSYT22] CHEN Z., SUN K., YANG F., TAO W.: Sc2-pcr: A second order spatial compatibility for efficient and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 13221–13231. 3
- [CWL*22] CHEN H., WEI Z., LI X., XU Y., WEI M., WANG J.: Repcdnet: Feature-aware recurrent point cloud denoising network. *International Journal of Computer Vision* 130, 3 (2022), 615–629. 2
- [CWX*22] CHEN H., WEI Z., XU Y., WEI M., WANG J.: Imlovenet: Misaligned image-supported registration network for low-overlap point cloud pairs. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings* (2022), pp. 1–9. 2
- [CWY*21] CHEN G., WANG M., YUE Y., ZHANG Q., YUAN L.: Full transformer framework for robust point cloud registration with deep information interaction. *arXiv preprint arXiv:2112.09385* (2021). 10
- [DKD09] DER KIUREGHIAN A., DITLEVSEN O.: Aleatory or epistemic? does it matter? *Structural safety* 31, 2 (2009), 105–112. 3
- [FLLW21] FU K., LIU S., LUO X., WANG M.: Robust point cloud registration framework based on deep graph matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 8893–8902. 2, 3, 7, 8, 9, 10
- [HGU*21] HUANG S., GOJIC Z., USVYATSOV M., WIESER A., SCHINDLER K.: Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 4267–4276. 2, 5, 6, 7, 8, 9, 10
- [HMZ20] HUANG X., MEI G., ZHANG J.: Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 11366–11374. 3
- [HMA21] HUANG X., MEI G., ZHANG J., ABBAS R.: A comprehensive survey on point cloud registration. *arXiv preprint arXiv:2103.02690* (2021). 6
- [JV87] JONKER R., VOLGENANT A.: A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing* 38, 4 (1987), 325–340. 6
- [LCL*21] LU F., CHEN G., LIU Y., ZHANG L., QU S., LIU S., GU R.: Hregnet: A hierarchical network for large-scale outdoor lidar point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 16014–16023. 3
- [LH22] LI Y., HARADA T.: Leopard: Learning partial point cloud matching in rigid and deformable scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 5554–5564. 4
- [LHF*21] LEE D., HAMSICI O. C., FENG S., SHARMA P., GERNOETH T.: Deeppro: Deep partial point cloud registration of objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 5683–5692. 3
- [LKCP21] LEE J., KIM S., CHO M., PARK J.: Deep hough voting for robust global registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 15994–16003. 3
- [LLF*19] LI R., LI X., FU C.-W., COHEN-OR D., HENG P.-A.: Pagan: a point cloud upsampling adversarial network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 7203–7212. 10

- [LWF22] LI X., WANG L., FANG Y.: Unsupervised category-specific partial point set registration via joint shape completion and registration. *IEEE Transactions on Visualization and Computer Graphics* (2022). 3
- [LXW*22] LU D., XIE Q., WEI M., XU L., LI J.: Transformers in 3d point clouds: A survey. *arXiv preprint arXiv:2205.07417* (2022). 2
- [LZB22] LIU J., ZHANG J., BARNES N.: Modeling aleatoric uncertainty for camouflaged object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2022), pp. 1445–1454. 3
- [LZX*20] LI J., ZHANG C., XU Z., ZHOU H., ZHANG C.: Iterative distance-aware similarity matrix convolution with mutual-supervised point elimination for efficient point cloud registration. In *European conference on computer vision* (2020), Springer, pp. 378–394. 2, 7, 8, 10
- [PRG*20] PAIS G. D., RAMALINGAM S., GOVINDU V. M., NASCIMENTO J. C., CHELLAPPA R., MIRALDO P.: 3dregnet: A deep neural network for 3d point registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 7193–7203. 3
- [QYW*22] QIN Z., YU H., WANG C., GUO Y., PENG Y., XU K.: Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 11143–11152. 3, 4, 5
- [RBB09] RUSU R. B., BLODOW N., BEETZ M.: Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE international conference on robotics and automation* (2009), IEEE, pp. 3212–3217. 2
- [SDMR20] SARLIN P.-E., DETONE D., MALISIEWICZ T., RABINOVICH A.: Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 4938–4947. 10
- [Sin64] SINKHORN R.: A relationship between arbitrary positive matrices and doubly stochastic matrices. *The annals of mathematical statistics* 35, 2 (1964), 876–879. 6
- [SLG*19] SARODE V., LI X., GOFORTH H., AOKI Y., SRIVATSAN R. A., LUCEY S., CHOSET H.: Pcnnet: Point cloud registration network using pointnet encoding. *arXiv preprint arXiv:1908.07906* (2019). 3
- [STDS14] SALTI S., TOMBARI F., DI STEFANO L.: Shot: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding* 125 (2014), 251–264. 2
- [UPH*19] UY M. A., PHAM Q.-H., HUA B.-S., NGUYEN T., YEUNG S.-K.: Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF international conference on computer vision* (2019), pp. 1588–1597. 7, 8
- [VSP*17] VASWANI A., SHAZEER N., PARMAR N., USZKOREIT J., JONES L., GOMEZ A. N., KAISER Ł., POLOSUKHIN I.: Attention is all you need. *Advances in neural information processing systems* 30 (2017). 2
- [WLD*21] WANG H., LIU Y., DONG Z., WANG W., YANG B.: You only hypothesize once: Point cloud registration with rotation-equivariant descriptors. *arXiv preprint arXiv:2109.00182* (2021). 2
- [WS19a] WANG Y., SOLOMON J. M.: Deep closest point: Learning representations for point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 3523–3532. 2, 4, 8
- [WS19b] WANG Y., SOLOMON J. M.: Pcnnet: Self-supervised learning for partial-to-partial registration. *Advances in neural information processing systems* 32 (2019). 2, 9
- [WSK*15] WU Z., SONG S., KHOSLA A., YU F., ZHANG L., TANG X., XIAO J.: 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 1912–1920. 7
- [WXH*21] WEN X., XIANG P., HAN Z., CAO Y.-P., WAN P., ZHENG W., LIU Y.-S.: Pmp-net: Point cloud completion by learning multi-step point moving paths. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 7443–7452. 4
- [WYF*22] WANG Y., YAN C., FENG Y., DU S., DAI Q., GAO Y.: Storm: Structure-based overlap matching for partial point cloud registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022). 2, 3, 4, 5, 6
- [WYW*18] WU Q., YANG H., WEI M., REMIL O., WANG B., WANG J.: Automatic 3d reconstruction of electrical substation scene from lidar point cloud. *ISPRS Journal of Photogrammetry and Remote Sensing* 143 (2018), 57–71. 2
- [XLW*21] XU H., LIU S., WANG G., LIU G., ZENG B.: Omnet: Learning overlapping mask for partial-to-partial point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 3132–3141. 5, 6, 7, 8, 9, 10
- [XWL*21] XIANG P., WEN X., LIU Y.-S., CAO Y.-P., WAN P., ZHENG W., HAN Z.: Snowflakenet: Point cloud completion by snowflake point deconvolution with skip-transformer. In *Proceedings of the IEEE/CVF international conference on computer vision* (2021), pp. 5499–5509. 4
- [YEK*20] YUAN W., ECKART B., KIM K., JAMPANI V., FOX D., KAUTZ J.: Deepgmr: Learning latent gaussian mixture models for registration. In *European conference on computer vision* (2020), Springer, pp. 733–750. 3
- [YKH*18] YUAN W., KHOT T., HELD D., MERTZ C., HEBERT M.: Pcn: Point completion network. In *2018 International Conference on 3D Vision (3DV)* (2018), IEEE, pp. 728–737. 4, 7
- [YL20] YEW Z. J., LEE G. H.: Rpm-net: Robust point matching using learned features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 11824–11833. 2, 3, 4, 6, 7, 8, 10
- [YL22] YEW Z. J., LEE G. H.: Regtr: End-to-end point cloud correspondences with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 6677–6686. 2, 3, 4, 5, 6, 7, 8, 9, 10
- [YLS*21] YU H., LI F., SALEH M., BUSAM B., ILIC S.: Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. *Advances in Neural Information Processing Systems* 34 (2021). 4, 10
- [YLX*19] YI C., LU D., XIE Q., LIU S., LI H., WEI M., WANG J.: Hierarchical tunnel modeling from 3d raw lidar point cloud. *Computer-Aided Design* 114 (2019), 143–154. 2
- [YYH*21] YAN Z., YI Z., HU R., MITRA N. J., COHEN-OR D., HUANG H.: Consistent two-flow network for tele-registration of point clouds. *IEEE Transactions on Visualization and Computer Graphics* (2021). 3
- [YZL*21] YANG F., ZHAI Q., LI X., HUANG R., LUO A., CHENG H., FAN D.-P.: Uncertainty-guided transformer reasoning for camouflaged object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 4146–4155. 3, 6
- [ZFD*20] ZHANG J., FAN D.-P., DAI Y., ANWAR S., SALEH F. S., ZHANG T., BARNES N.: Uc-net: Uncertainty inspired rgb-d saliency detection via conditional variational autoencoders. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 8582–8591. 3
- [ZFF*21] ZHOU H., FENG Y., FANG M., WEI M., QIN J., LU T.: Adaptive graph convolution for point cloud analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 4965–4974. 4
- [ZSD*22] ZHANG Z., SUN J., DAI Y., ZHOU D., SONG X., HE M.: End-to-end learning the partial permutation matrix for robust 3d point cloud registration. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2022), pp. 3399–3407. 2