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

Zhilei Chen1, Honghua Chen†, Lina Gong1, Xuefeng Yan1, Jun Wang1, Yanwen Guo3, Jing Qin4, Mingqiang Wei1,2†

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

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...
We design a novel point cloud registration network, which is the data and detailed analysis show that our approach achieves state-bility rigid transformation parameters are computed, by considering subsequent overlap prediction and dense matching. Finally, qualitative results show that our approach significantly enhances the performance of registration.

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) 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, STD∗14] 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
assignment to prevent ambiguity. RPM-Net [YL20] uses the differentiable Sinkhorn layer and annealing algorithm to get soft assignments of point correspondences from hybrid 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 explicitly. YEK*20 leverages a probabilistic registration paradigm by formulating registration as the minimization of KL-divergence between two probability distributions 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. UCNet [ZFD*20] proposes probabilistic RGB-D saliency detection network via conditional variational autoencoders to model human annotation uncertainty. [YZL*21] utilizes a probabilistic representative 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.

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

3.1. Problem Formulation and Overview

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

\[
\min_{R, t} \sum_{i=1}^{N} \sum_{j=1}^{M} C_{i,j} \| R \mathbf{p}_i + t - \mathbf{q}_j \|_2^2,
\]

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

\[
\min_{R, t} \sum_{i=1}^{N} \sum_{j=1}^{M} C_{i,j} w_{i,j} \| R \mathbf{p}_i + t - \mathbf{q}_j \|_2^2,
\]

where \( w_{i,j} \) is the weight computed based on the overlap scores, and \( C \) 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 \( P \) and \( Q \), 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 [ZH21] layers (64, 64, 128, 256) followed by an MLP, as shown in Fig. 3. The learned point-wise feature of \( P \) and \( Q \) are represented as \( \hat{F}^P \in \mathbb{R}^{N \times V} \) and \( \hat{F}^Q \in \mathbb{R}^{M \times V} \), where \( V \) 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 [YKH18], for involving rich geometric shape knowledge into the point-wise feature extractor. The completion decoder produces coarse results \( \mathbf{P} \), \( \mathbf{Q} \) and fine results \( \hat{\mathbf{P}} \), \( \hat{\mathbf{Q}} \). During our early experiments, we find that it is not necessary to carefully design a completion model, like [WXH21, WYL21]. 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.](image)

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, YLS21, WYF22, 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. [QYW22] 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 \( \hat{F}^P \in \mathbb{R}^{N \times V} \) and \( \hat{F}^Q \in \mathbb{R}^{N \times V} \).

![Figure 4: Detailed architecture of the geometry transformer.](image)

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 $P$ (the same for $Q$). Given the features $F \in \mathbb{R}^{P \times d_t}$, where $d_t$ is the dimension of the hidden feature, the output features $Z \in \mathbb{R}^{P \times d_t}$ are the weighted sum of all projected input features as

$$Z_i = \sum_{j=1}^{P} a_{ij} \left( f_i W^Q \right)$$

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

$$e_{ij} = \frac{\left( f_i W^Q \right)^T \left( f_j W^K \right) + g_{ij} W^G}{\sqrt{d_t}}$$

where $g_{ij}$ is the geometric relation embedding, $W^Q, W^K$, and $W^V \in \mathbb{R}^{d_t \times d_t}$ are the projection weights for queries, keys and values, $W^G \in \mathbb{R}^{1 \times 1}$ 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 $O(N(d_t^2 + 3))$. Usually, the feature dimension $d_t$ is far larger than 3, our solution therefore has less space complexity.

**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 as

$$\eta_{ij} = \left( d^k_{i1} + d^k_{i2} + d^k_{i3} \right) - \left( d^l_{i1} + d^l_{i2} + d^l_{i3} \right) + \left\| p_i - p_j \right\|_2$$

Finally, the geometric relation embedding $g_{ij}$ is computed by aggregating the point-wise distance, triplet-wise angle and the triangle perimeter difference as

$$g_{ij} = CAT[\eta_{ij}, a_{ij}, \eta_{ij}]$$

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

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

$$Z^P_i = \sum_{j=1}^{Q} a_{ij} \left( f_i W^Q \right)$$

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

$$e_{ij} = \frac{\left( f_i W^Q \right)^T \left( f_j W^K \right)^T}{\sqrt{d_t}}$$

The cross-attention feature of $Q$ 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.

**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, WYP’22, YLLB22] 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 $P$, we assign each point

\[\text{Figure 5: Left: An illustration of the geometric relation embedding. Right: Computation flow of the geometry self-attention.}\]

\[\text{Figure 6: Overview of the uncertainty quantification module. This module decomposes the sample operation into a trainable part and a random part for end-to-end learning.}\]
a Gaussian distribution parameterized by the mean \(\mu_i\) and the variance \(\sigma_i\), which are predicted by different MLP layers from \(\hat{F}^P\). The overlap score of \(p\) can be then sampled from the learned distribution, \(o_i \sim \mathcal{N}(\mu_i, \sigma_i)\). Following [YWL*21], we decompose the sampling operation into two steps (see from Fig. 6). First, one random sample \(e_i\) is obtained from the standard Gaussian distribution \(\mathcal{N}(0, I)\), i.e., \(e_i \sim \mathcal{N}(0, I)\), and the desired sample is computed by \(\mu_i + e_i \sigma_i\). To measure point-wise overlap uncertainty, we then resample \(K\) overlap scores for each point from the learned distribution, denoted as \(O^P = \{o(1), \ldots, o(K)\}\). 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

\[
U^P = \text{norm}(\text{var}(O^P)),
\]

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

### Affinity matching

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

\[
A_{i,j} = (\hat{F}^P)^T W (\hat{F}^Q),
\]

where \(W \in \mathbb{R}^{V \times V}\) 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 [Sim64] algorithm to calculate the soft correspondence matrix \(\hat{C}\). More details please refer to [YL20].

### 3.4. Overlap-guided Registration

As previously investigated, approaches [HGU*21, XWL*21, WYW*22, YL23] 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. [WYW*22] applies gumbel softmax to sample the overlap correspondences. OMNet [XWL*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 [HZM21]. Hence, we propose an overlap-guided registration module to tackle this problem.

With the uncertainty-aware features \(\hat{F}^P\), \(\hat{F}^Q\), another geometry transformer is adopted to predict the reliable overlap scores, \(\hat{O}^P\) and \(\hat{O}^Q\). Meanwhile, the final soft correspondence \(\hat{C}\) is also obtained by an affinity matching module. We then leverage the linear assignment problem (LAP) solver [JV87] to compute the binary hard correspondence \(C\) from \(\hat{C}\). 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:

\[
w_{i,j} = \left\{ \begin{array}{ll}
\hat{a}_{i,j} \hat{a}_{i,j} & \hat{C}_{i,j} = 1 \\
0 & \hat{C}_{i,j} = 0
\end{array} \right.
\]

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 \(\mathcal{L} = \mathcal{L}_C + \mathcal{L}_a + \mathcal{L}_c + \mathcal{L}_e\).

**Registration loss.** We adopt the cross-entropy loss between the predicted soft correspondence \(\hat{C}\) and the ground truth correspondence \(C\) to train our model. The formula is as

\[
\mathcal{L}_r = - \sum_{i,j}^N (C_{i,j} \log \hat{C}_{i,j} + (1 - C_{i,j}) \log (1 - \hat{C}_{i,j})).
\]

The ground-truth correspondence \(C\) is computed by mutual near-
© 2022 The Author(s)

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     | Ours            | 0.2238  | 0.1322 | 0.00212 | 0.00131| 0.2464    | 0.00264  |
|              | Ours            | 15.9602 | 8.0429 | 0.13187 | 0.07255| 15.2811   | 0.15084  |
| ModelLoNet   | Ours            | 2.5463  | 0.8588 | 0.02098 | 0.00765| 1.5934    | 0.01555  |
|              | Ours            | 12.6069 | 7.9538 | 0.11367 | 0.07984| 15.2959   | 0.16384  |
| ScanObjectNN | Ours            | 2.1477  | 0.7448 | 0.02268 | 0.00614| 1.3842    | 0.01247  |
|              | Ours            | 3.6790  | 0.2633 | 0.01394 | 0.00179| 0.5601    | 0.00366  |

est neighbour searching as

\[ C_{i,j} = \begin{cases} 1, & N(N(q_i, T(p))) = p_i, N(N(T(p_i), Q)) = q_j \\ 0, & \text{otherwise} \end{cases} \]

where \( N(N(\cdot)) \) denotes the spatial nearest neighbor, \( T(\cdot) \) is the operation of the ground-truth rigid transformation from \( P \) to \( Q \), and \( \rho \) denotes the distance threshold.

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

\[ L_P = \frac{1}{|P|} \sum_{i=1}^{|P|} \left( \theta_i^p \log (\tilde{\theta}_i^p) + (1 - \theta_i^p) \log (1 - \tilde{\theta}_i^p) \right), \]

where \( \tilde{\theta}_i^p \) denotes the ground-truth overlap of \( P \) as

\[ \tilde{\theta}_i^p = \begin{cases} 1, & \sum_{j=1}^{|Q|} C_{i,j} = 1 \\ 0, & \text{otherwise} \end{cases} \]

The overlap loss \( L_P^O \) can be computed in the same way. We then get a total overlap loss: \( L_o = L_P^b + L_P^O \).

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

\[ L_u^P = \lambda \cdot L_{BCE}(\sigma^b, \tilde{\sigma}^b) + \eta \cdot KL(N(\mu, \sigma)\|N(0, I)), \]

where \( \sigma^b \) denotes one example drawn from the overlap distribution, which is used to boost the diversity, \( \lambda \) and \( \eta \) are balance factors. The final uncertainty loss is \( L_u = L_u^P + L_u^O \).

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

\[ L_c^P = CD(P, \hat{P}) + CD(\hat{P}, \hat{T}), \]

where \( P \) is the ground truth completion result. The total completion loss can be computed as \( L_c = L_c^P + L_c^O \).

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 clouds is transformed by a random rotation in the range of \([0, 45]^\circ\) and a random translation \( t \) in the range of \([-0.5, 0.5]\) along each axis. Gaussian noise sampled from \( N(0, 0.01) \) 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 $\lambda, \eta$ 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 ModelLoNet 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.](image)

### 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.

![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.](image)

### 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 ModelLoNet 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].

### 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) | MAE(t) | Error(R) | Error(t) |
|---|---|---|---|---------|--------|--------|----------|----------|
|   |   |   |   | 0.3082  | 0.1907  | 0.00195 | 0.00495  | 0.00397  |
| √ |   |   |   | 0.2837  | 0.1481  | 0.00241 | 0.00440  | 0.00300  |
| √ | √ |   |   | 0.2735  | 0.1470  | 0.00238 | 0.00389  | 0.00277  |
| √ | √ | √ |   | 0.2601  | 0.1404  | 0.00237 | 0.00336  | 0.00273  |
| √ | √ | √ | √ | 0.2238  | 0.1322  | 0.00212 | 0.00131  | 0.00264  |

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

| UTOPIC | RMSE(R) | MAE(R) | MAE(t) | Error(R) | Error(t) |
|--------|---------|--------|--------|----------|----------|
| vanilla self-attention | 0.2691  | 0.1404  | 0.00239 | 0.00156  | 0.2600  |
| self-attention w/APA | 0.2667  | 0.1406  | 0.00256 | 0.00149  | 0.2718  |
| self-attention w/LPE | 0.2686  | 0.1400  | 0.00226 | 0.00132  | 0.2567  |
| geometry self-attention | 0.2238  | 0.1322  | 0.00212 | 0.00131  | 0.2664  |

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

| Method     | ModelNet (N=717) | ModelLoNet (N=512) |
|------------|------------------|--------------------|
| IDAM       | 0.07             | 0.07               |
| RPM-Net    | 0.16             | 0.13               |
| OMNet      | 0.03             | 0.03               |
| RGM        | 0.17             | 0.14               |
| Predator   | 0.29             | 0.23               |
| REGTR      | 0.05             | 0.05               |
| Ours       | 0.06             | 0.06               |

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 \( \sigma^2 \) 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 additional 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 gargoyles 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       | 0.07             | 0.07               |
| RPM-Net    | 0.16             | 0.13               |
| OMNet      | 0.03             | 0.03               |
| RGM        | 0.17             | 0.14               |
| Predator   | 0.29             | 0.23               |
| REGTR      | 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. 2021szsvp060), the Natural Science Foundation of Guangdong Province (No. 2022A151010170), the Innovation and Technology Fund - Midstream Research Program 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] AOKE Y., GOFORTH H., SRIVATSA R. A., LUCEY S.: Pointnetlk: Robust & efficient point cloud registration using pointnetk. 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.: Pournsc: 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
[BMS92] 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] CHOU 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
[CPC19] CHOU 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., FANG F., TAO W.: S2cp-c: 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.: Repcnet: 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 point 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 KRIEPHERHIAN A., DITLEVESSEN O.: Aleatory or epistemic? does it matter? Structural safety 31, 2 (2009), 105–112. 3
[FLL19] FUKU 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 (2019), pp. 8893–8902. 2, 3, 7, 8, 9, 10
[HGU+21] HUANG S., GODIC Z., USSVATSOV 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, 3, 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 correspondence. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020), pp. 11366–11374. 3
[HMZA21] HUANG X., MEI G., ZHANG J., ABBAS R.: A comprehensive survey on point cloud registration. arXiv preprint arXiv:2103.02690 (2021). 6
[JVN78] 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., LEE 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
[LLH22] LI Y., HARADA T.: Lepard: 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., GERNOTH 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.: Pu-gan: a point cloud upsampling adversarial network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2019), pp. 7203–7212. 10

© 2022 The Author(s)
Computer Graphics Forum © 2022 The Eurographics Association and John Wiley & Sons Ltd.
Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2021), pp. 7443–7452. 4

[WYW∗22] Wang Y., Yan C., Feng Y., Du S., 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., Zheng B.: Ommnet: 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., Pan 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.: Rnmp-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, 5, 6, 7, 8, 9, 10

[YL22] Yew Z. J., Lee G. H.: Regr: 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), 145–154. 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 salency detection via conditional variational autoencoders. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2020), pp. 8852–8859. 3

[ZFF∗20] 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