G3DOA: Generalizable 3D Descriptor With Overlap Attention for Point Cloud Registration

Hengwang Zhao, Hanyang Zhuang, Chunxiang Wang, and Ming Yang

***Abstract*—Point cloud registration (PCR) is a key problem for robotics, autonomous driving, and other applications. Constructing generalizable 3D descriptors and determining whether a 3D descriptor is in the overlapping area are challenging tasks in PCR. Despite the fast evolution of learning-based 3D descriptors, existing methods are either sensitive to rigid transformation and scenario changes, or can not find the appropriate descriptors in the overlapping area, which means their generalization and descriptive ability are not enough for practice applications. To solve these problems, we propose a novel neural network, named G3DOA, which jointly learns generalizable rotation-invariant 3D descriptors and their overlap scores (representing the probability in the overlapping area), to enhance the generalization ability of the descriptors across different data collected by various laser sensors. To ensure the rotation and scale invariance of the point cloud in the input stage, we estimate the Local Reference Frame (LRF) of local patches and normalize the coordinates. To learn generalizable and distinctive descriptors, we propose a novel Cylindrical LRF convolution module with multi-scaled cylindrical shells and neural layers, which hierarchically encodes and aggregates the geometric information in different cylindrical shells. Moreover, to estimate the probability of whether a point is in the overlapping area, we propose an overlap attention module that extracts co-contextual information between the feature encodings of the two point clouds. The experiments show that G3DOA trained only on an indoor dataset can be efficiently generalized to complex outdoor datasets, and the generalization ability of G3DOA outperforms state-of-the-art learning-based 3D descriptors.**

***Index Terms*—Computer vision for automation, AI-based methods.**

I. Introduction

POINT cloud registration (PCR) aims to find the rigid transformation of two partially overlapped point clouds with unknown correspondences, which plays an important role in many robotics and intelligent vehicles related applications, e.g., mapping [1], localization [2], and object pose estimation [3]. Although this problem has been studied for many years, it is particularly challenging when the point clouds are noisy, partially visible and with large initial misalignment.

The PCR approaches can be roughly divided into two categories. One category computes the transformation by iteratively minimizing the geometric distance of nearest neighbors [4], [5] or optimizing the distance of probability distribution [1], [6]. But these approaches require an initial transformation, otherwise they tend to get trapped in local minima. Another category establishes correspondences by 3D descriptors [7], [8] then computes the transformation without any initial guess, whose performance depends on the quality of the descriptors and which point to select (e.g. whether the points are in the overlapping area). We follow the second approach to design generalizable 3D descriptors and find the appropriate points for registration.

Early hand-crafted 3D descriptors are constructed by the geometrical properties [7] of local surfaces or the relative coordinates of Local Reference Frames (LRF) [8]. Although guarantee rotation-invariance, all these handcrafted descriptors are usually sensitive to noise, due to the limitation of descriptive ability. Recently, learning-based 3D descriptors have been developed owing to the powerful feature representation capabilities of deep learning [9]–[12]. But these approaches have three main limitations. Firstly, many of these methods take the original 3D coordinates as input and rely on kernel-based point convolution [13] to extract pointwise features [14], [15], causing the learned descriptors to be rotation-variant. So their performance drops dramatically when they are applied to unseen scenarios during training or data with strong rotational changes. Secondly, although some of the recent approaches [12], [16] introduce learning-based rotation-invariant 3D descriptors and show the generalization ability in a certain extent, they can not sufficiently extract distinctive local geometric information, which limits their performance when generalized to different datasets. Thirdly, most of these approaches do not focus on whether a point is in the overlapping area, which also limits the registration performance using the 3D descriptors. Although the recent approach [17] learns descriptors and overlap scores using a kernel-based point convolution, it can not maintain rotation-invariant and is hard to be generalized across different datasets.

In this paper, we propose a novel neural network, named G3DOA, which jointly learns generalizable rotation-invariant 3D descriptors and their overlap scores (representing the probability in the overlapping area), to enhance the generalization ability of the descriptors across different data collected by

Manuscript received September 9, 2021; accepted December 30, 2021. Date of publication January 13, 2022; date of current version January 28, 2022. This letter was recommended for publication by Associate Editor Z. Min and Editor C. C. Lerma upon evaluation of the reviewers' comments. This work was supported by the National Natural Science Foundation of China under Grants 62173228 and 61873165. *(Corresponding author: Ming Yang.)*

Hengwang Zhao, Chunxiang Wang, and Ming Yang are with the Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai 200240, China (e-mail: zhaohw1995@sjtu.edu.cn; wangcx@sjtu.edu.cn; mingyang@sjtu.edu.cn).

Hanyang Zhuang is with the University of Michigan - Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: zhuanghany11@sjtu.edu.cn).

Digital Object Identifier 10.1109/LRA.2022.3142733

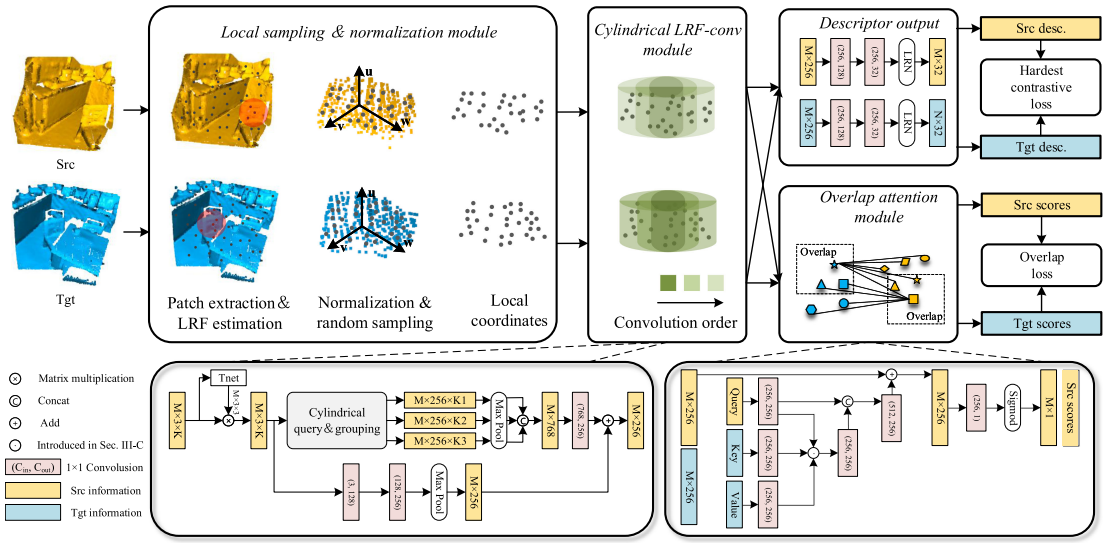

Fig. 1. Architecture of G3DOA. The inputs are the source point cloud (yellow) and the target point cloud (blue). The local sampling & normalization module estimates the LRF of each local patch (red area) and converts the point cloud into a rotation-invariant and scale-invariant representation, where the gray points indicate the randomly sampled points. The Cylindrical LRF-conv module hierarchically encodes and aggregates the geometric information in different cylindrical shells (dark green to light green). The descriptor output module and the overlap attention module output the descriptors and overlap scores respectively.

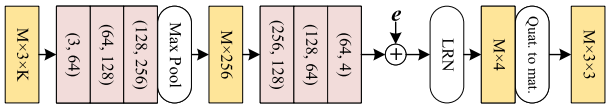

Fig. 2. Framework of the Transformation network.

various laser sensors, as shown in Fig. 3. Specifically, we first estimate the LRF of local patches and normalize the relative coordinates in each local patch to convert the point cloud into a rotation-invariant and scale-invariant representation in the input stage. To effectively extract the geometric information of each local patch and make the descriptor more distinctive and generalizable, we propose a novel Cylindrical LRF convolution module with multi-scaled cylindrical shells and neural layers, which hierarchically encodes and aggregates the geometric information in different cylindrical shells. Moreover, we propose an overlap attention module that extracts co-contextual information between the feature encodings of the two point clouds to estimate the probability of a point whether it is in the overlapping area. Finally, extensive experiments show that G3DOA trained only on the indoor 3DMatch dataset [9] can be efficiently generalized to complex outdoor datasets [18], [19], and the generalization ability of G3DOA outperforms state-of-the-art learning-based 3D descriptors.

II. Related Work

Handcrafted 3D descriptors. Pioneer works on hand-crafted 3D descriptors focus on how to describe the rotation-invariant local geometric information, which can be roughly divided into two categories: the LRF-free methods and the LRF-based methods. The LRF-free methods such as SI [20], LSP [21] and FPFH [7] are typically based on the rotation-invariant geometric properties of local surfaces. But these descriptors lack sufficient geometric details for the local surface. The LRF-based methods such as PS [3], SHOT [8] and RoPS [22] can not only characterize the geometric patterns of the local area but also effectively exploit the 3D spatial attributes. Despite significant progress, these hand-crafted 3D local descriptors are usually tailored to specific tasks and sensitive to noise, which still fails to handle highly noisy and large-scale real-world 3D point clouds.

Learning-based 3D descriptors. Recent methods leveraging the power of deep neural networks tend to have strong descriptive ability and robustness, which can be divided into rotation-variant 3D descriptors [9], [14], [15], [17], [23], [24] and rotation-invariant 3D descriptors [12], [16], [25]–[27]. The former whose input and feature extracting procedure can not maintain rotation-invariance achieves good performance on specific datasets by extensive data augmentation. The pioneering work 3DMatch [9] takes the local volumetric patches as input then leverages 3D CNN to learn local geometric patterns. FCGF [14] builds dense feature descriptors through 3D sparse convolutions. D3feat [15] jointly learns both dense feature detectors and local descriptors using a kernel-based point convolution [13]. However, all these methods are sensitive to rigid transformation and scenarios changes, which are hard to be generalized to different datasets. The latter whose input and feature extracting procedure are rotation-invariant shows better generalization ability. PPF-Net [25] uses the Point Pair Features as inputs which are fed into multiple MLPs to learn descriptors. PerfectMatch [16] introduces the voxelized Smoothed Density Value (SDV) to encode the local patch, which is fed into a Siamese architecture to learn the final descriptor. DIP [27] uses the normalized coordinates in local patches as input, then feed them into a PointNet [28] to learn descriptor. SpinNet [12] uses a spatial point transformer to project the original coordinates to a cylindrical space then apply a 3D cylindrical convolutional neural layers to learn descriptors. Although different strategies have been proposed to achieve rotation-invariant representation,

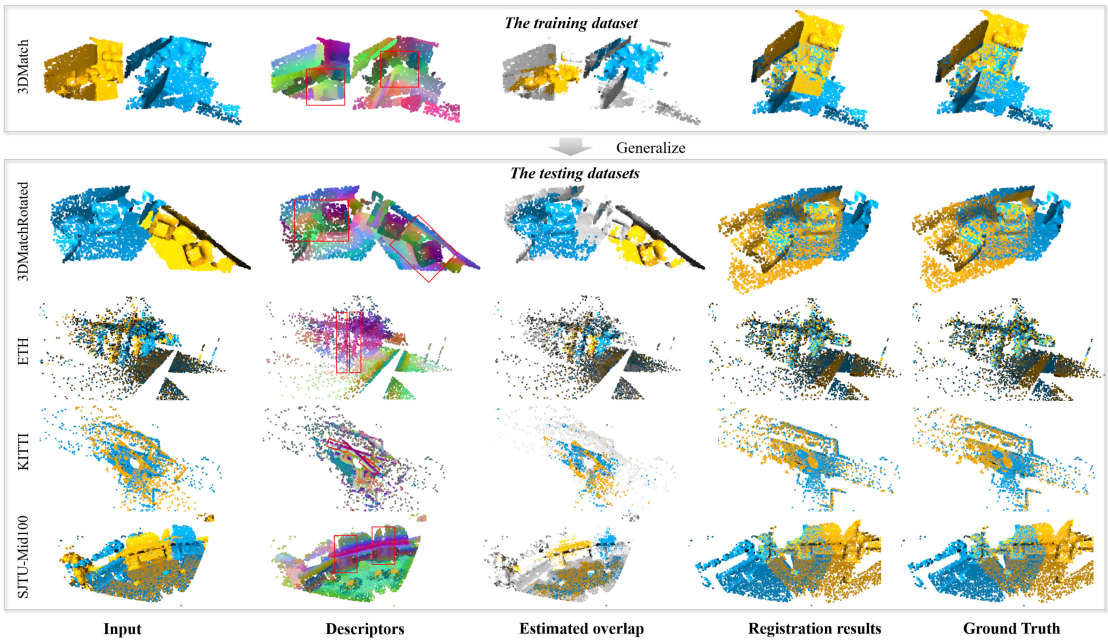

Fig. 3. Example results of G3DOA trained on 3DMatch [9] and tested on other datasets [18], [19]. The estimated overlapping areas are visualized in the 3-th column, where the colored points are in the overlapping areas and the grey points are in the non-overlapping areas. We use t-SNE [33] to visualize the descriptors (the 2-th column) by reducing the dimension to three and associating each low dimensional vector with RGB color. We can see that the colors of corresponding areas (the boxed points) are similar and that of non-corresponding areas are distinguishable, validating the rotation-invariance and distinctiveness of the descriptors.

how to effectively extract distinctive local geometric information is insufficiently explored, which limits the generalization ability of these methods.

At the same time, most of these methods can not determine whether a point is in the overlapping area, which also limits the registration performance using the 3D descriptors. Despite the recent approach [17] learns descriptors and overlap scores using a kernel-based point convolution [13], it can not maintain rotation-invariant and is difficult to be generalized across different datasets.

III. Methodology

The architecture of G3DOA is shown in Fig. 1, which is a Siamese architecture that processes pairs of point clouds using two branches with shared weights. G3DOA takes the source point cloud and target point cloud as input, and outputs the descriptors and overlap scores of the sampled points of each point cloud. The architecture of G3DOA can be decomposed into four main modules. (1) The local sampling and normalization module converts the point cloud into a rotation-invariant and scale-invariant representation, and reduces the number of points by random sampling (Section III-A). (2) The cylindrical LRF-conv module learns generalizable and distinctive features of local patches leveraging multi-scaled cylindrical shells and neural layers (Section III-B). (3) The overlap attention module extracts co-contextual information between the feature encodings of the two point clouds, and assigns each point an overlap score representing how likely the point is located in the overlapping area between the two input point clouds (Section III-C). (4) The descriptor output module project the feature encodings into 32-dimensional descriptors with L2 normalization (Section III-D).

A. Local Sampling and Normalization

The Local sampling and normalization module is to convert the point cloud into a representation that is rotation-invariant and scale-invariant in the input stage. This is the premise of making the learned 3D descriptor has the ability to generalize to different datasets. It contains a sequence of operations: patch extraction, LRF estimation, and points sampling and normalization.

Patch extraction. Given the source point cloud $\hat{P} = \{\hat{\mathbf{p}}_1, \hat{\mathbf{p}}_2, \ldots, \hat{\mathbf{p}}_{\hat{M}}\} \in \mathbb{R}^{\hat{M}\times 3}$ and the target point cloud $\hat{Q} = \{\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2, \ldots, \hat{\mathbf{q}}_{\hat{N}}\} \in \mathbb{R}^{\hat{N}\times 3}$, we randomly sample M points in each point cloud as center points for patch extraction (randomly permutate the points then select the first M points), generating $P = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_M\} \in \mathbb{R}^{M\times 3}$ and $Q = \{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_M\} \in \mathbb{R}^{M\times 3}$. The local patch of a center point $\mathbf{p}_i \in P$ is defined as $L_i = \{\hat{\mathbf{p}} : \|\hat{\mathbf{p}} - \mathbf{p}_i\|_2 \leq r, \hat{\mathbf{p}} \in \hat{P}, \mathbf{p}_i \in P\}$, where r denotes the radius of the patch and $\|\cdot\|_2$ denotes the Euclidean distance. To learn a general representation for patches including the same geometric structures but with different coordinates, and to make the patches having the same number of points for training the network using batch processing, we randomly sample l points in L_i and get $L_i' \subseteq L_i$. If the number of points in a patch is less than l, we randomly pick points in L_i as paddings until l points are sampled.

LRF estimation. For each center point $\mathbf{p}_i \in P$, we adapt the method proposed in [29] to estimate the LRF using the points in L_i'. The LRF is constructed by computing three orthogonal axes: the $\mathbf{u}_{\mathbf{p}_i}$ axis is defined by the normal vector of the local patch; the $\mathbf{w}_{\mathbf{p}_i}$ axis is defined by a weighted sum of the projection of vectors between $\mathbf{p}_i$ and the points in L_i' on the plane orthogonal to $\mathbf{u}_{\mathbf{p}_i}$; the $\mathbf{v}_{\mathbf{p}_i}$ axis is the cross-product between $\mathbf{u}_{\mathbf{p}_i}$ and $\mathbf{w}_{\mathbf{p}_i}$. The details of coping with the sign ambiguity problem of $\mathbf{u}_{\mathbf{p}_i}$

and improving the repeatability of $\mathbf{w}_{\mathbf{p}_i}$ can be referred in [29]. After computing the three axes, we combine them into a rotation matrix $R_{\mathbf{p}_i}$ using to transform the points in L_i' from the global reference frame of P to the LRF, where $R_{\mathbf{p}_i} = [\mathbf{u}_{\mathbf{p}_i}, \mathbf{v}_{\mathbf{p}_i}, \mathbf{w}_{\mathbf{p}_i}]^\top$.

Sampling and normalization. Although we have randomly sampled l points before LRF estimation, the number of points is still too large (the accurate estimation of LRF requires a certain number of points), which is not suitable as the direct input to the network. To improve the efficiency of the deep network and to achieve large batches processing during training, we re-sample K points from L_i' and get $L_i'' \subseteq L_i'$, where $K < l$. To achieve the rotation-invariance of the descriptor, we represent the coordinates of the points in L_i'' as the offset relative to their patch center $\mathbf{p}_i$, and we transform the points using the rotation matrix $R_{\mathbf{p}_i}$ of their LRF. To achieve the scale invariance the of descriptor, we normalize these relative coordinates by the radius of the patch r:

$$P_{L_i} = \{y : y = R_{\mathbf{p}_i}(\hat{\mathbf{p}} - \mathbf{p}_i)/r, \hat{\mathbf{p}} \in L_i,'' \mathbf{p_i} \in P\} \quad (1)$$

where $P_{L_i} \in \mathbb{R}^{3\times K}$ is the transformed and normalized coordinates of the local patch centered as $\mathbf{p}_i$.

The local sampling and normalization module is applied to each center point in P and Q, and generates $P_L \in \mathbb{R}^{M\times 3\times K}$ and $Q_L \in \mathbb{R}^{M\times 3\times K}$, as the input of the deep network. Besides, the proposed G3DOA is a Siamese architecture that processes pairs of point clouds using two branches with shared weights. In the following contents, we use the local patches of source point cloud P_L as an example to introduce our approach.

B. Cylindrical LRF-Conv Module

This module is designed to learn generalizable and distinctive descriptors from the normalized coordinates in each LRF leveraging the multi-scaled cylindrical shells and neural layers. Before this, inspired by [28], we apply a Transformation network (Tnet) to correct the noise of LRF estimation due to clutter or occlusions to promote the repeatability of the descriptors.

Transformation Network. The framework of Tnet is shown in Fig. 2, which includes a series of 1×1 convolution and max pooling. The Local Response Normalisation (LRN) layer performs L2 normalization of the output. Specifically, Tnet takes the pre-processed local patches $P_L \in \mathbb{R}^{M\times3\times K}$ as input and outputs normalized 4-dim vectors $\mathbf{A_Q} \in \mathbb{R}^{M\times4}$ that can be treated as quaternions. Different from the transformation network in PointNet directly outputting rotation matrices, Tnet directly outputs quaternions which ensures the transformation is strictly constrained to be SO(3), which is important to subsequent multi-scale cylindrical querying and grouping operations. Because the clutter and occlusions will lead to a slight deviation during LRF estimating, we also expect our designed Tnet to estimate a slight rotation to compensate for such deviation. Therefore, a unit vector $e = [1, 0, 0, 0]$ is added before the LRN layer to make the output vary around the unit vector e. Then, we convert the quaternions output $\mathbf{A_Q} \in \mathbb{R}^{M\times4}$ to rotation matrices $\mathbf{A_R} \in \mathbb{R}^{M\times3\times3}$ so they can be directly applied to local patches using matrix multiplications. By applying $\mathbf{A_R}$ to P_L, we get $P_{LA} = \mathbf{A_R}P_L \in \mathbb{R}^{M\times3\times K}$, which are the refined local patches using the predicted transformations.

Cylindrical LRF Convolution. Once the normalized and refined local patches $P_{LA} \in \mathbb{R}^{M\times3\times K}$ are obtained, it is crucial to encode local geometric structures from these local coordinates. To achieve this, we propose an efficient cylindrical LRF convolution operator. Specifically, we define a series of multi-scaled cylindrical shells in LRF space, which have a shared central axis and different radii, as is shown in Fig. 1 (taking three cylindrical shells as an example). These cylindrical shells have a ring-like shape and have no overlaps. The central axis of these cylindrical shells is defined by the normal direction $\mathbf{u}_{\mathbf{p}_i}$ and the radii are defined by $[0, r_{c_1}], [r_{c_1}, r_{c_2}], [r_{c_2}, r_{c_3}]$ (the order is from inside to outside). Because the norms of the local coordinates have been normalized to $[0, 1]$ in Section III-A, the radii can be adapted to datasets with different scales. Then, we define the cylindrical query & grouping operation to group the local coordinates in the LRF according to the cylindrical shells with different scales. In particular, which cylindrical shell a point belongs to is determined by the distance between its local coordinate and the $\mathbf{u}_{\mathbf{p}_i}$ axis. We sample K_1, K_2, and K_3 local coordinates from each cylindrical shell respectively (from inside to outside). The cylindrical query & grouping operation also contains a series of 1×1 convolutions to map these local coordinates to high dimensional space, resulting in $P_{LAC_n} \in \mathbb{R}^{M\times256\times K_n}$, where n is the serial number of the cylindrical shell. To make the features more robust to noise, we apply max pooling to aggregate features of the points in each cylindrical shell, resulting in $P_{LAC_n^{max}} \in \mathbb{R}^{M\times256}$. Suppose that the cylindrical shells yield the domain Ω_C. We use $|\Omega_C|$ to represent the number of cylindrical shells and we can define the convolution as:

$$P_{LAC}^{(s)} = \sum_{n=1}^{|\Omega_C|} \omega_n^{(s)} P_{LAC_n^{max}}^{(s-1)} \quad (2)$$

where ω_n denotes the weights applied on the n-th cylindrical shell, $P_{LAC_n^{max}}$ denotes the features of the n-th cylindrical shell, P_{LAC} denotes the features of the local patches, superscript (s) denotes the data or parameters of layer s. We implement this operation by concatenating the pooled features in each cylindrical shell then applying the 1×1 convolution, as is shown in Fig. 1. Besides, we applied the residual connection to integrate the global information of patches into the final output of the cylindrical LRF-conv module, resulting $P_{LAF} \in \mathbb{R}^{M\times256}$.

The proposed Cylindrical LRF-conv has some properties: (1) The multi-scaled cylindrical shells contribute to capturing hierarchical geometry information of local patches, which makes the descriptor more generalizable and distinctive. (2) Cylindrical LRF-conv is robust to noise, owing to the pooling operation in each cylinder-shaped voxel. (3) Compared with the cylindrical convolution in [12], the proposed method is more efficient, because Cylindrical LRF-conv learns hierarchical features by dividing fewer voxels (cylindrical shells).

C. Overlap Attention Module

So far, we get the features $P_{LAF} \in \mathbb{R}^{M\times 256}$ and $Q_{LAF} \in \mathbb{R}^{M\times 256}$ of the center points of two point clouds, which encode the hierarchical geometry information of local patches. But P_{LAF} has no information of Q_{LAF} and vice versa. Intuitively, if we want to estimate the overlap region, some cross-talks between two point clouds are essential. Like human operators aligning two partial point clouds, they need to compare the similarity of the local patches in the two point clouds to determine the overlapping area. To achieve this, we adopt a cross-attention [30] block based on the message passing formulation [31]. Akin to database retrieval in the Transformer architecture [32], the query $\mathbf{s}_i$, retrieves the values $\mathbf{v}_j$ of some elements based on their attributes which are the keys $\mathbf{k}_j$. In our case, for $\mathbf{p}_{LAF_i} \in P_{LAF}$ and $\mathbf{q}_{LAF_j} \in Q_{LAF}$, the query $\mathbf{s}_i$ is $\mathbf{p}_{LAF_i} \in \mathbb{R}^{256}$, the values $\mathbf{v}_j$ and the keys $\mathbf{k}_j$ are $\mathbf{q}_{LAF_j} \in \mathbb{R}^{256}$. The messages are computed as weighted averages of the values:

$$\mathbf{m}_i = \sum_{j:(i,j)\in\epsilon} a_{ij}\mathbf{v}_j \tag{3}$$

where attention weights $a_{ij} = softmax(\mathbf{s}_i^\top \mathbf{k}_j)/\sqrt{|\mathbf{s}_i|}$. The co-contextual features are computed as:

$$\mathbf{p}_{LAF_i}^{CA} = \mathbf{p}_{LAF_i} + MLP(cat[\mathbf{p}_{LAF_i}, \mathbf{m}_i]) \tag{4}$$

where $MLP(\cdot)$ is implemented by the 1×1 convolution, $cat[\cdot,\cdot]$ denotes the concatenate operation, $\mathbf{p}_{LAF_i}^{CA}$ is the final latent feature encoding with cross-attention which is conditioned on the features of another point cloud. Then the features are linearly projected to the overlap scores which can be interpreted as probabilities that the points in P lie in the overlap region. The detailed structure is shown in Fig. 1, where $\bigodot$ denotes the the calculation process of (4). Besides, the same cross-attention block is applied in both directions, so that information flows are $P_{LAF} \rightarrow Q_{LAF}$ and $Q_{LAF} \rightarrow P_{LAF}$.

D. Descriptor Output Module

The descriptor output module using a series of 1×1 convolutions project $P_{LAF} \in \mathbb{R}^{M\times 256}$ and $Q_{LAF} \in \mathbb{R}^{M\times 256}$ to 32-dimensional descriptors $\mathbf{F}_P \in \mathbb{R}^{M\times 32}$ and $\mathbf{F}_Q \in \mathbb{R}^{M\times 32}$. Lastly, the descriptors are normalized by a Local Response Normalisation (LRN) layer with L2 normalization.

E. Loss Functions

Descriptor loss. We use the hardest contrastive loss [14] to make the distance of corresponding descriptors to be smaller and that of non-corresponding descriptors to be at least a margin away. Give a positive pair $(\mathbf{f}_i \in \mathbf{F}_P, \mathbf{f}_j \in \mathbf{F}_Q)$, which are the descriptors of a pair of corresponding points, we mine the hardest negatives $(\tilde{\mathbf{f}}_i, \tilde{\mathbf{f}}_j)$ and define the loss function as:

$$\begin{aligned}\mathcal{L}_d = \frac{1}{M'} \sum_{(\mathbf{f}_i,\mathbf{f}_j)\in\mathcal{C}_+} \Bigg\{ & \left[D\left(\mathbf{f}_i,\mathbf{f}_j\right) - m_p\right]_+^2 / |\mathcal{C}_+| \\ & + \left[m_n - \min_{\tilde{\mathbf{f}}_i\in\mathcal{C}_-(\mathbf{f}_i)} D\left(\mathbf{f}_i,\tilde{\mathbf{f}}_i\right)\right]_+^2 / 2\left|\mathcal{C}_-(\mathbf{f}_i)\right| \\ & + \left[m_n - \min_{\tilde{\mathbf{f}}_j\in\mathcal{C}_-(\mathbf{f}_j)} D\left(\mathbf{f}_j,\tilde{\mathbf{f}}_j\right)\right]_+^2 / 2\left|\mathcal{C}_-(\mathbf{f}_j)\right| \Bigg\}\end{aligned} \tag{5}$$

where $D(\cdot,\cdot)$ is a distance measure, M' is the number of corresponding points (positive pairs), $m_p = 0.1$ and $m_n = 1.4$ are the margins for positive and negative pairs, $[\cdot]_+$ takes the positive part of its arguments, $\mathcal{C}_+$ is the set of descriptors of positive pairs and $\mathcal{C}_-$ is the set of descriptors used for the hardest-negative mining extracted from a minibatch, $\mathcal{C}_-(\mathbf{f_i})$ is defined as $\mathcal{C}_-(\mathbf{f}_i) = \{\tilde{\mathbf{f}}_i : \|\tilde{\mathbf{p}} - \mathbf{p}_i\|_2 > r_s, \tilde{\mathbf{f}}_i \in \mathcal{C}_-\}$, where r_s is a safe radius to avoid selecting negative points that are spatially close to the anchors.

Overlap loss. We treat the estimation of the overlap scores as a binary classification problem and supervise the overlap scores using the overlap loss $\mathcal{L}_o = \frac{1}{2}(\mathcal{L}_o^P + \mathcal{L}_o^Q)$, where

$$\mathcal{L}_o^P = \frac{1}{|P|}\sum_{i=1}^{|P|} \bar{o}_{\mathbf{p}_i} \log\left(o_{\mathbf{p}_i}\right) + \left(1-\bar{o}_{\mathbf{p}_i}\right)\log\left(1-o_{\mathbf{p}_i}\right) \tag{6}$$

where $o_{\mathbf{p}_i}$ is the predicted overlap score, $\bar{o}_{\mathbf{p}_i}$ is the ground truth overlap label obtained by applying ground truth transformation and nearest neighbor searching.

The total loss $\mathcal{L} = \mathcal{L}_d + \mathcal{L}_o$ is the sum of the descriptor loss and the overlap loss.

IV. Experiments

A. Datasets

To show the generalization ability of G3DOA, four real-world datasets are used in the experiments, where 3DMatch [9] is used for training and testing, the ETH [18], KITTI [19] and SJTU-Mid100 datasets are only used for testing.

The 3DMatch dataset is an indoor point cloud dataset consisting of 62 real-world scenes collected by RGB-D cameras, which are split into 54 scenes for training and 8 scenes for testing. Each scene includes several partially overlapped fragments with their ground truth transformations. To evaluate the rotation invariance ability of G3DOA, we follow [12], [15] to create 3DMatchRotated for testing, where each point cloud is rotated around all the three axes by angles randomly and independently sampled in [0, 360°].

The ETH dataset is an outdoor dataset captured by static terrestrial scanners, which contains partially overlapped scans of field vegetation in different seasons, and the ground truth transformations. Four sequences resulting 713 point cloud pairs are used in the experiments.

The KITTI Odometry dataset is an outdoor sparse point cloud dataset acquired by Velodyne HDL-64E LiDAR sensors,

which contains 11 sequences of scans in outdoor driving scenarios. For a fair comparison, we following [12], [15] use sequences 8-10 as the test set, and use the ICP algorithm to reduce noise in the ground truth transformations. Only the point cloud pairs with at least 10 m intervals are selected, resulting in 555 point cloud pairs for testing.

The SJTU-Mid100 dataset is a self-collected dataset in an outdoor campus environment using a Livox Mid100 laser sensor, which uses a new non-repetitive scanning pattern to obtain point clouds. The ground truth is provided by RTK-GPS and refined by ICP. Only the point cloud pairs with at least 5 m intervals are selected, resulting in 50 point cloud pairs for testing.

B. Implementation Details

During training, we down sample the point clouds in the 3DMatch dataset using a voxel size of 0.025 m and randomly sample $M = 512$ patch centers for each point cloud. We set $r = 0.5\ m$ for local patch extraction. For each local patch, we randomly sample $l = 2000$ points for LRF estimation. After LRF estimation, we randomly sample $K = 512$ points feed into the network. The safe radius in the hardest-contrastive loss is set to $r_s = 0.1\ m$. We train the network for 10 epochs, performing around 16 K iterations per epoch. Each iteration processes a point cloud pair. We use SGD optimizer with initial learning rate of 0.1 that decreases by a factor 0.1 every 3 epochs, and with weight decay $5 \cdot 10^{-5}$.

During testing, we set $K = 1024$ to capture richer geometric information. The radius of the patches are set to $r = 0.6\ m$ in the 3DMatch dataset, $r = 1.5\ m$ in the ETH dataset, $r = 3.0\ m$ in the KITTI Odometry dataset and $r = 3.0\ m$ in the SJTU-Mid100 dataset. The point clouds in the ETH, KITTI Odometry and SJTU-Mid100 datasets are down sampled with a voxel size of 0.05 m. For all the datasets, the radii of the cylindrical voxels are $r_{c1} = 0.4\ m$, $r_{c2} = 0.8\ m$, and $r_{c3} = 1.0\ m$ because the coordinates in local patches are scale normalized. We all randomly sample $M = 5000$ patch centers for descriptor generation and we discard points with a predicted overlap score below 0.5. The final rigid transformation is estimated by RANSAC [16] with 50000 max iterations. All of the experiments are conducted on the platform with a NVIDIA GTX 1080 Ti GPU, an Intel Xeon E5-2620 CPU, and 16 G memory.

C. Evaluation Metrics

Feature Matching Recall (FMR) is used as main evaluation metric to measure the performance of learned descriptors on the 3DMatch and ETH datasets. Suppose that there are H pairs of point clouds in a dataset and each pair of point clouds P_h and Q_h can be aligned by the ground truth transformation $\mathbf{T}_h = \{\mathbf{R}_h, \mathbf{t}_h\}$, the average FMR is defined as:

$$FMR = \frac{1}{H}\sum_{h=1}^{H} \mathbb{1} \left(\left[\frac{1}{|\mathbf{\Omega}_h|} \sum_{(\mathbf{p}_i, \mathbf{q}_j) \in \mathbf{\Omega}_h} \mathbb{1}\left(\|\mathbf{p}'_i - \mathbf{q}_j\|_2 < \tau_1 \right) \right] > \tau_2 \right) \quad (7)$$

TABLE I
RESULTS ON THE 3DMATCH DATASET

	3DMatch		3DMatchRotated		Feat.	Rot.
	FMR ↑	STD ↓	FMR ↑	STD ↓	Dim.	Aug.
FPFH [7]	35.9	13.4	36.4	13.6	33	No
SHOT [8]	23.8	10.9	23.4	9.5	352	No
3DMatch [9]	59.6	8.8	1.1	-	512	No
PPFNet [25]	62.3	10.8	0.3	-	64	No
PPF-FoldNet [26]	71.8	10.5	73.1	10.4	512	No
PerfectMatch [16]	94.7	2.7	94.9	2.5	32	No
FCGF [14]	95.2	2.9	95.3	3.3	32	Yes
D3Feat-rand [15]	95.3	2.7	95.2	3.2	32	Yes
D3Feat-pred [15]	95.8	2.9	95.5	3.5	32	Yes
Predator [17]	96.1	2.9	95.7	3.3	32	Yes
DIP [27]	94.8	4.6	94.6	4.6	32	No
SpinNet [12]	**97.6**	**1.9**	**97.5**	**1.9**	32	No
G3DOA(Ours)	96.3	2.7	96.1	2.9	**32**	**No**

The bold entries indicate the best performance and the underline entries indicate the second-best performance.

where $\mathbf{p}'_i = \mathbf{R}_h\mathbf{p}_i + \mathbf{t}_h$, $\|\cdot\|$ denotes the Euclidean distance, $\tau_1 = 10\ cm$ is the inlier distance threshold, $\tau_2 = 0.05$ is the inlier ratio threshold, $\mathbb{1}$ is the indicator function, Ω_h is the set of point correspondences found in the descriptor space via mutual nearest-neighbour searching [16].

Registration Metrics. We use the Relative Rotation Error (RRE), Relative Translation Error (RTE) and Success Rate (SR) to evaluate the registration performance of our method on the KITTI Odometry dataset. For a pair of point clouds P_h and Q_h, the RRE is calculated as: $RRE_h = arccos(\frac{trace(\mathbf{R}_h^\top \bar{\mathbf{R}}_h)-1}{2})$, where $\bar{\mathbf{R}}_h$ denotes the estimated rotation matrix. The RTE is calculated as: $RTE_h = \|\mathbf{t}_h - \bar{\mathbf{t}}_h\|_2$, where $\bar{\mathbf{t}}_h$ denotes the estimated translation matrix. A pair of point clouds is considered as successfully registered if $RTE_h < 2\ m$ and $RRE_h < 5°$[12]. SR is the number of successful point cloud pairs divided by the total number H.

D. 3DMatch Training and Testing

In the first experiment, we evaluate the performance of G3DOA that training on 3DMatch and testing on 3DMatch and 3DMatchRotated, where FMR and its standard deviation (STD) are used as the metrics. We compare the proposed G3DOA with several approaches: the hand-crafted descriptors FPFH [7] and SHOT [8], and the learning-based descriptors 3DMatch [9], PPFNet [25], PPF-FoldNet [26], PerfectMatch [16], FCGF [14], D3Feat [15], Predator [17], DIP [27], and SpinNet [12]. Table I shows the results. We can see that G3DOA outperforms most of the state-of-the-art approaches both on 3DMatch and 3DMatchRotated without any rotation augmentation, which shows the rotation-invariance and distinctiveness of the descriptor. The qualitative results are shown in the first two rows of Fig. 3. Because the testing set of 3DMatch is in the same domain as the training set (all collected by RGB-D sensors), the rotation-variant descriptors [14], [15], [17] get the FMR more than 90% by extensive data augmentation, but these approaches are difficult to generalize to unseen datasets collected by different laser sensors. Although SpinNet [12] is slightly better than G3DOA on 3DMatch, its generalization ability and efficiency are not as good as G3DOA, which will be analyzed below.

TABLE II
RESULTS ON THE ETH DATASET

	Gazebo (FMR ↑)		Wood (FMR ↑)		
	Summer	Winter	Autumn	Summer	AVG ↑
FPFH [7]	38.6	14.2	14.8	20.8	22.1
SHOT [8]	73.9	45.7	60.9	64.0	61.1
3DMatch [9]	22.8	8.3	13.9	22.4	16.9
PerfectMatch [16]	91.3	84.1	67.8	72.8	79.0
FCGF [14]	22.8	10.0	14.8	16.8	16.1
D3Feat-rand [15]	45.7	23.9	13.0	22.4	26.2
D3Feat-pred [15]	85.9	63.0	49.6	48.0	61.6
Predator [17]	41.3	24.6	28.7	42.4	34.3
DIP [27]	90.8	88.6	96.5	95.2	92.8
SpinNet [12]	92.9	91.7	92.2	94.4	92.8
G3DOA(Ours)	**94.6**	**97.9**	**99.1**	**100.0**	**98.1**

TABLE III
RESULTS ON THE KITTI ODOMETRY DATASET

	RTE (cm)		RRE (degree)		Success
	AVG ↓	STD ↓	AVG ↓	STD ↓	Rate (%) ↑
FCGF [7]	27.1	5.58	1.61	1.51	24.19
D3Feat-rand [15]	37.8	9.98	1.58	1.47	18.47
D3Feat-pred [15]	31.6	10.1	1.44	1.35	36.76
Predator [17]	35.4	11.3	1.45	1.10	66.13
SpinNet [12]	15.6	**1.89**	0.98	0.63	81.44
G3DOA(Ours)	**9.32**	3.61	**0.53**	**0.55**	**96.76**

TABLE IV
EFFECTS OF DIFFERENT COMPONENTS

CyLRF-conv	3DCCL [12]	OA	3DMatch (FMR ↑)	ETH (FMR ↑)	KITTI (SR ↑)
	✓		97.6	92.8	81.4
	✓	✓	**97.9**	93.8	88.6
✓			95.2	95.8	90.2
✓		✓	96.3	**98.1**	**96.8**

E. Generalization to Different Datasets

The second experiment is to evaluate the generalization ability of the approaches on unseen datasets collected by various sensors. All of the learning-based descriptors are only trained on the 3DMatch dataset. The large domain gap of the training set and the testing sets poses a great challenge to the generalization of all approaches. Table II shows the results on the ETH dataset. We can see that the performance of rotation-variant learning-based descriptors [14], [15], [17] drop a lot on the ETH dataset. Because they only realize approximately rotation-invariant features in one domain through extensive data augmentation but are sensitive to domain changes. Although the hand-crafted rotation-invariant descriptors [7], [8] can be generalized to different datasets, they show relatively poor performance because of the sensitivity to noise and occlusion. The learning-based rotation-invariant descriptors [12], [16], [27] can also be generalized to different datasets and show better performance than the hand-crafted rotation-invariant descriptors, owing to the learning process makes the descriptors more robust to noise and occlusion. But their performance is still degraded compared to that in Table I. We deduce that it is because the descriptors are not distinctive enough to distinguish non-corresponding local areas that have a similar appearance. Besides, these approaches can not handle the impact of the non-overlapping areas. The proposed G3DOA outperforms other approaches, improving by 5.3% in terms of FMR compared with Spinnet [12] and DIP [27]. We infer that the improvement is attributed to the Cylindrical LRF-conv module extracting hierarchical local geometric information which makes the descriptor more distinctive while maintaining rotation-invariance, and the overlap attention module removing most of the irrelevant points in the non-overlapping area. The qualitative results are shown in the third row of Fig. 3.

We evaluate the registration performance of the descriptors on the KITTI Odometry dataset, which is a large-scale outdoor dataset. Table III shows the results. Because KITTI and 3DMatch are collected by different types of laser sensors and in different scenarios, the learning-based rotation-variant approaches [14], [15], [17] can not effectively generalize from 3DMatch to KITTI dataset. In contrast, SpinNet [12] and the proposed G3DOA, which are learning-based rotation-invariant descriptors, successfully generalize from 3DMatch to KITTI dataset. Meanwhile, G3DOA improves by 15.32% in terms of SR compared with SpinNet, which validates the effectiveness of the proposed Cylindrical LRF-conv and the overlap attention module. The qualitative results are shown in the fourth row of Fig. 3. Additionally, we test G3DOA on the SJTU-Mid100 dataset and the results are shown in the fifth row of Fig. 3. We can see that G3DOA can be effectively generalized to the data collected by the non-repetitive scanning sensors.

F. Ablation Studies

In this experiment, we analyze how different components affect the performance of our method. To study the effectiveness of the Cylindrical LRF-conv module (CyLRF-conv), we replace CyLRF-conv with the 3D Cylindrical Convolution Layers (3DCCL) in SpinNet [12]. For the 3DCCL in SpinNet [12], we follow the hyperparameters and data preprocessing procedure mentioned in their paper. To study the effectiveness of the Overlap Attention module (OA), we remove OA from the G3DOA architecture and integrate OA into SpinNet [12]. We train all these models on the 3DMatch dataset and test them on the 3DMatch, ETH, and KITTI datasets. The results are reported in Table IV. Comparing rows 1 & 3 and rows 2 & 4, we find that 3DCCL [12] only performs better on the training dataset (3DMatch); however, the proposed CyLRF-conv shows a stronger generalization ability across different datasets. We infer that the fine-grained cylindrical voxels in 3DCCL [12] help the network capture more details on the training dataset but they are relatively sensitive to dataset changing (different densities or scales). In contrast, the coarse-grained cylindrical shells in CyLRF-conv extract hierarchical information while being robust to dataset changing promoting the generalization ability. Besides, the coordinates used in CyLRF-conv are normalized in the LRF, which is also conducive to generalizing to different datasets without tuning too many parameters. Comparing rows 1 & 2 and rows 3 & 4, we observe that OA improves the performance of both 3DCCL [12] and CyLRF-conv, and OA significantly improves the SR on the KITTI dataset. We believe that OA removes most of the irrelevant points in the non-overlapping area, which contributes to final registration.

TABLE V
AVERAGE RUNNING TIME PER POINT

	Data prep. [ms]	Inference [ms]	Total [ms]
SpinNet [12]	2.17	7.94	10.11
G3DOA(ours)	2.40	**0.53**	**2.93**

G. Running Time

We analyze the average data preparation and inference time of generating one descriptor of SpinNet [12] and G3DOA on the same platform (mentioned in Section IV-B). Table V shows the results. We can see the total time of a descriptor extracted by G3DOA is around 3 times faster than SpinNet [12]. We deduce that is because of the higher efficiency of Cylindrical LRF-conv using fewer voxels to aggregate features, and the reduction of the number of points inputting to the network through random sampling.

V. CONCLUSION

In this paper, we aim at proposing generalizable 3D descriptors to enhance the performance of PCR. A novel neural network (G3DOA) to jointly learn distinctive rotation-invariant 3D descriptors and their overlap scores has been developed. The network consists of two main modules which are the Cylindrical LRF-conv module and overlap attention module. The Cylindrical LRF-conv module hierarchically encodes the rotation-invariant local geometric information, while the overlap attention module is used to estimate the probability of whether a point is in the overlapping area. Extensive experiments demonstrate that the G3DOA can be efficiently generalized across different datasets. The performance of our model (only trained on 3DMatch) outperforms state-of-the-art methods on two complex outdoor datasets. Moreover, these datasets are collected from different types of laser scanners (Kinect, Velodyne LiDAR, etc.) and in various scenarios (indoor, field, city, etc.), and the excellent performance of G3DOA in them demonstrates its potential for various mobile robotic tasks such as mapping and localization.

REFERENCES

[1] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3D-NDT," *J. Field Robot.*, vol. 24, no. 10, pp. 803–827, 2007.

[2] R. Dubé *et al.*, "Incremental-segment-based localization in 3-D point clouds," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1832–1839, Jul. 2018.

[3] C. S. Chua and R. Jarvis, "Point signatures: A new representation for 3D object recognition," *Int. J. Comput. Vis.*, vol. 25, no. 1, pp. 63–85, 1997.

[4] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[5] M. Brossard, S. Bonnabel, and A. Barrau, "A new approach to 3D ICP covariance estimation," *IEEE Robot. Automat. Lett.*, vol. 5, no. 3, pp. 744–751, Jan. 2020.

[6] L. Li, M. Yang, C. Wang, and B. Wang, "Robust point set registration using signature quadratic form distance," *IEEE Trans. Cybern.*, vol. 50, no. 5, pp. 2097–2109, May 2020.

[7] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2009, pp. 3212–3217.

[8] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 356–369.

[9] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3DMatch: Learning local geometric descriptors from RGB-D reconstructions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1802–1811.

[10] J. Li and G. H. Lee, "USIP: Unsupervised stable interest point detection from 3D point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 361–370.

[11] H. Zhao, Z. Liang, C. Wang, and M. Yang, "CentroidReg: A global-to-local framework for partial point cloud registration," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2533–2540, Feb. 2021.

[12] S. Ao, Q. Hu, B. Yang, A. Markham, and Y. Guo, "Spinnet: Learning a general surface descriptor for 3D point cloud registration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11753–11762.

[13] H. Thomas, C. R. Qi, J.-E' Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6411–6420.

[14] C. Choy, J. Park, and V. Koltun, "Fully convolutional geometric features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8958–8966.

[15] X. Bai, Z. Luo, L. Zhou, H. Fu, L. Quan, and C.-L' Tai, "D3Feat: Joint learning of dense detection and description of 3D local features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6359–6367.

[16] Z. Gojcic, C. Zhou, J. D. Wegner, and A. Wieser, "The perfect match: 3D point cloud matching with smoothed densities," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5545–5554.

[17] S. Huang, Z. Gojcic, M. Usvyatsov, A. Wieser, and K. Schindler, "Predator: Registration of 3D point clouds with low overlap," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4267–4276.

[18] F. Pomerleau, M. Liu, F. Colas, and R. Siegwart, "Challenging data sets for point cloud registration algorithms," *Int. J. Robot. Res.*, vol. 31, no. 14, pp. 1705–1711, 2012.

[19] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The kitti vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.

[20] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.

[21] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1252–1262, 2007.

[22] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, "Rotational projection statistics for 3D local surface description and object recognition," *Int. J. Comput. Vis.*, vol. 105, no. 1, pp. 63–86, 2013.

[23] Y. Wang and J. M. Solomon, "Deep closest point: Learning representations for point cloud registration," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3523–3532.

[24] H. Deng, T. Birdal, and S. Ilic, "3D local features for direct pairwise registration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3244–3253.

[25] H. Deng, T. Birdal, and S. Ilic, "PPFNet: Global context aware local features for robust 3D point matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 195–205.

[26] H. Deng, T. Birdal, and S. Ilic, "PPF-FoldNet: Unsupervised learning of rotation invariant 3D local descriptors," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 602–618.

[27] F. Poiesi and D. Boscaini, "Distinctive 3D local deep descriptors," in *Proc. IEEE 25th Int. Conf. Pattern Recognit.*, 2021, pp. 5720–5727.

[28] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.

[29] J. Yang, Q. Zhang, Y. Xiao, and Z. Cao, "Toldi: An effective and robust approach for 3D local shape description," *Pattern Recognit.*, vol. 65, pp. 175–187, 2017.

[30] P.-E' Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: Learning feature matching with graph neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4938–4947.

[31] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1263–1272.

[32] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[33] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.