



## Article

# Partial-to-Partial Point Cloud Registration by Rotation Invariant Features and Spatial Geometric Consistency

Yu Zhang \*, Wenhao Zhang and Jinlong Li

School of Physical Science and Technology, Southwest Jiaotong University, Chengdu 610036, China; zhangwenhao@my.swjtu.edu.cn (W.Z.)

\* Correspondence: zhangyu2@swjtu.edu.cn

**Abstract:** Point cloud registration is a critical problem in 3D vision tasks, and numerous learning-based point cloud registration methods have been proposed in recent years. However, a common issue with most of these methods is that their feature descriptors are rotation-sensitive, which makes them difficult to converge at large rotations. In this paper, we propose a new learning-based pipeline to address this issue, which can also handle partially overlapping 3D point clouds. Specifically, we employ rotation-invariant local features to guide the point matching task, and utilize a cross-attention mechanism to update the feature information between the two point clouds to predict the key points in the overlapping regions. Subsequently, we construct a feature matrix based on the features of the key points to solve the soft correspondences. Finally, we construct a non-learning correspondence constraint module that exploits the spatial geometric invariance of the point clouds after rotation and translation, as well as the compatibility between point pairs, to reject the wrong correspondences. To validate our approach, we conduct extensive experiments on ModelNet40. Our approach achieves better performance compared to other methods, especially in the presence of large rotations.

**Keywords:** point cloud registration; deep learning; spatial geometric consistency; rotation-invariant descriptors



**Citation:** Zhang, Y.; Zhang, W.; Li, J. Partial-to-Partial Point Cloud Registration by Rotation Invariant Features and Spatial Geometric Consistency. *Remote Sens.* **2023**, *15*, 3054. <https://doi.org/10.3390/rs15123054>

Academic Editors: John Trinder, Sara Gonizzi Barsanti and Sajjad Roshandel

Received: 2 April 2023

Revised: 6 June 2023

Accepted: 7 June 2023

Published: 10 June 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Three-dimensional point cloud registration is of great significance in robotics and computer vision to find a rigid body transformation to align a pair of point clouds with unknown point correspondences. It has many important applications in scene reconstruction [1–3], localization [4], autonomous driving [5] and so on. The most widely utilized traditional registration method is the iterative closest point (ICP) [6], which is alternated between the two steps: solving the point correspondences and rigid transformation. However, ICP is sensitive to initialization and often converges to the wrong local minima. Some global registration algorithms, i.e., GO-ICP [7] and fast global registration (FGR) [8], are proposed to overcome the limitations of ICP, but they can easily fail in the case of noise or partially overlapping point clouds.

In recent years, the deep learning model has dominated the field of computer vision. The point cloud registration algorithms based on deep learning [9–15] are faster and more robust than traditional algorithms. Roughly, they could be divided into two categories: correspondences-free methods and correspondences-learning methods. Correspondences-free methods [9–11] regress the rigid motion parameters by minimizing the difference of feature maps between two input point clouds. Although they have good robustness for noise, most of them hardly deal with partially overlapping 3D point clouds. The main idea of correspondences-learning methods is to establish correspondences through the high-dimensional features of each point. Examples range from deep closest point (DCP) [12], PRNet [13] and RPMNet [14] to IDAM [15]. However, most of these networks do not explicitly deal with error correspondences, and they often fail at large rotations.

Based on the above discussion, in this paper, we propose a learning-based pipeline for partially overlapping 3D point cloud registration with large rotations. We address the issue of sensitivity to rotation in feature descriptors by utilizing rotation-invariant features based on 4D point pair features (PPF) [16]. However, relying solely on high-dimensional rotation-invariant features can lead to overfitting during network training, and the lack of position information about the position of each point can result in similar features for points in smooth or symmetric regions, leading to mismatches in key points. In order to make the feature contain position information and be robust to rotation, we use a two-branch feature extraction strategy for the point clouds, and allow the rotation-invariant feature to guide the global feature after positional encoding. However, there are always a large number of wrong correspondences in feature matching. While weighting the correspondences is a common practice, such weights are closely tied to the matching features and may fail to eliminate incorrect point pairs. In order to solve this problem, we propose a non-learning correspondence constraint module, which does not rely on the feature of point cloud, but only utilizes the geometric invariance after rotation and translation. We leverage the bidirectional correlation of distance between the inline point pairs to reject the wrong correspondences. Finally, the transformation matrix is estimated using a differentiable singular value decomposition (SVD) layer. Extensive experiments demonstrate that the method we have proposed can effectively eliminate errors in noise-free data, and achieves better performance on noisy point clouds with large rotations compared to many traditional methods and methods based on deep learning.

## 2. Related Work

### 2.1. Traditional Point Cloud Registration Methods

The most widely utilized traditional local registration method is ICP [6]. It alternates between finding the point correspondences between the source and target point clouds and solving the least-squares problem [17]. Although many algorithms [18–20] utilize different methods to solve the time and convergence of ICP, unfortunately, ICP and its variants are sensitive to initialization and easily converge to the local minimum.

The global registration algorithm random sample consensus (RANSAC) [21] is another important registration algorithm. It usually utilizes fast point feature histogram (FPFH) [22] or signature of histogram of orientation (SHOT) [23] to extract the features of the point clouds, and randomly selects a fixed number of points for estimation in each iteration to compute a rough transformation. Although these methods can effectively remove outliers, they are very time-consuming. FGR [8] utilizes FPFH to describe the features of the point clouds and find the corresponding point pairs in the feature space. Go-ICP [7] utilizes a branch-and-bound scheme to search for the optimal solution in the pose space. Furthermore, 4PCS [24] finds a set of four corresponding points between two point clouds, and then uses the correspondences between these points to calculate the rigid transformation. The advantages of the 4PCS algorithm are high efficiency and strong robustness. However, most of these methods are very sensitive to noise, and do not work well on partially overlapping 3D point cloud registration.

### 2.2. Correspondences-Free Methods

PointNetLK [9] is the first to utilize deep learning to process 3D point cloud registration. It combines PointNet [25] and Lucas and Kanade [26] to register through feature alignment and iterative processing. PCRNet [10] is another global registration network that utilizes PointNet for feature extraction and utilizes multi-layer perceptron (MLP) for regression of rotation and translation parameters. OMNet [11] learns masks in a coarse-to-fine manner to reject non-overlapping regions, however, it is difficult to accurately estimate the masks without feature information interaction. Although these methods achieve good performance in their own experiments, their performance deteriorates when the point clouds are partially overlapping. In contrast, our work belongs to correspondences-learning meth-

ods, which require only a small amount of matching correspondences to achieve accurate and effective point cloud registration.

### 2.3. Correspondences-Learning Methods

DCP [12] utilizes the dynamic graph CNN (DGCNN) network [27] to extract the local features from point clouds for forming soft correspondences and solving least square problems through an SVD layer. However, it is assumed to have a one-to-one corresponding relationship in the two point clouds. DCP has been extended to PRNet [13], which includes a key points detection module to perform partial-to-partial registration. RPMNet [14] utilizes a differentiable Sinkhorn [28] layer and annealing to obtain soft assignments of point correspondences from hybrid features learned from both spatial coordinates and local geometry. IDAM [15] combines feature and Euclidean information into the corresponding matrix, and utilizes a two-stage learnable point elimination technique for registration. However, these methods depend on the similarity of feature descriptors of key points, and the network cannot converge if only encoding the coordinates through shared convolution layers when the rotation is large and the difference between the coordinates of two clouds is significant. In contrast to these methods, we adopt a two-branch feature description strategy that includes position information and rotation-invariant local features to obtain the high-dimensional embedding of the point clouds.

### 2.4. Rotation-Invariant Descriptors

The FPFH descriptor [22] is conventionally generated based on geometric properties of local surfaces such as curvature and normal deviation. On the other hand, PPF [16] utilizes Euclidean distances and angles between point vectors and normals to describe each pair of relations. Although these hand-crafted descriptors are rotationally invariant by design, they remain sensitive to noise. To address this issue, PPFNet [29] represents unorganized point clouds as a combination of points, normals and point pair features to describe local geometric features. In subsequent work [30], FoldingNet [31] is adopted instead of multiple MLPs as the backbone network to learn 3D local descriptors. Nevertheless, all those methods are constrained by their locality and do not take into account the absolute position of the points, which may result in a large number of mismatched points with similar local features being utilized as key points. So, in the implementation of our network, we incorporate it as an auxiliary branch.

## 3. Method

This section describes the proposed point cloud registration model, and the entire network architecture is illustrated in Figure 1. The global features and rotation-invariant features of the two point clouds are extracted through two branches (Section 3.1). By employing the cross-attention mechanism, the features of the point clouds can perceive contextual information from each other, specifically focusing on key points within overlapping regions. Subsequently, a feature matrix is constructed on the features of the key points to address the soft correspondences (Section 3.2). Finally, a space geometric consistency constraint module (SGC) is utilized to reject the outliers (Section 3.3).

### 3.1. Feature Extraction Network

Global features are extracted using a simplified graph neural network (GNN) architecture. Unlike the approach described in the original paper [27], our network avoids the use of dynamically changing neighborhoods in the graph. This modification is made to prevent the feature information from being propagated differently across different regions, which can interfere with achieving symmetrical point cloud registration [32]. The feature extraction framework is shown in Figure 2. We only construct the graph structure between coordinates, not between features. Specifically, suppose we have a point cloud  $X$ ,  $N_i$  is the index of  $K$  points closest to point  $x_i$  in point cloud  $X$ , which can be obtained by the  $K$ -nearest neighbor algorithm ( $K$ -NN). Let  $u_i^{(n)}$  be the high-dimensional space feature vector

of the  $n$ th layer of the point  $x_i$  in the GNN. Then the feature of point  $x_i$  in the next layer is computed as:

$$u_i^{(n+1)} = f(\max_{j \in N_i} g(u_j^{(n)} - u_i^{(n)})), \tag{1}$$

where  $g$  is composed of two MLPs with normalization and ReLU activations,  $f$  is a single-layer MLP with the same input and output dimensions, which aims to further enhance the feature information, and  $\max$  is the element-wise max operation.

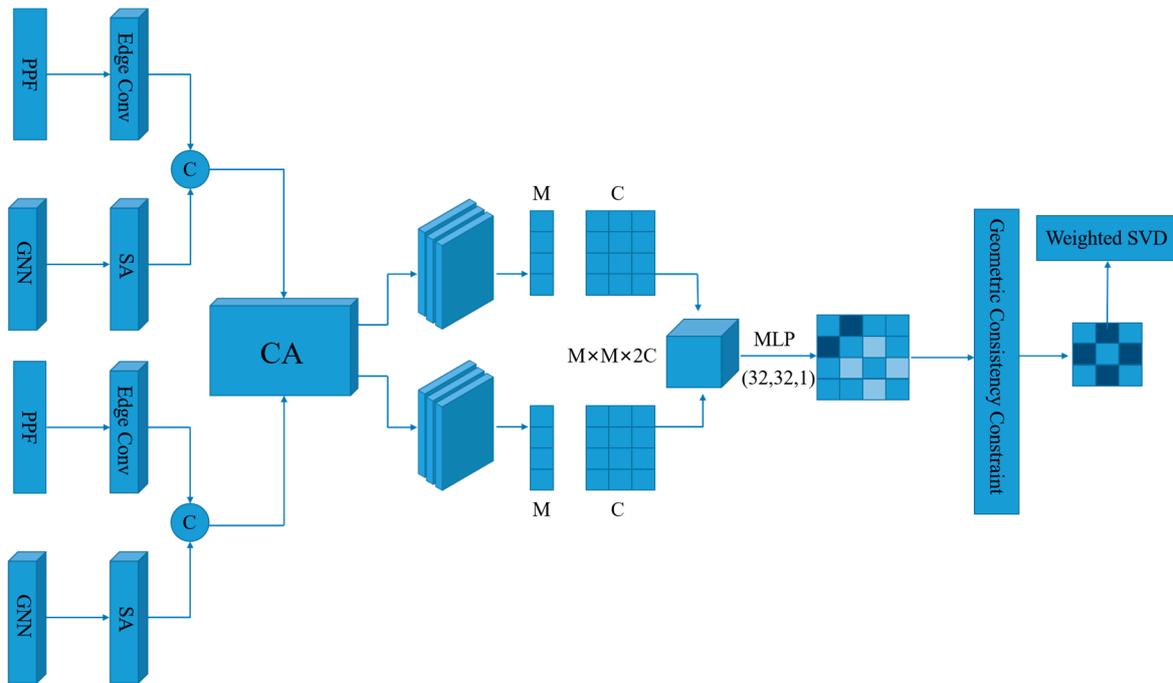


Figure 1. Overview of the network structure.

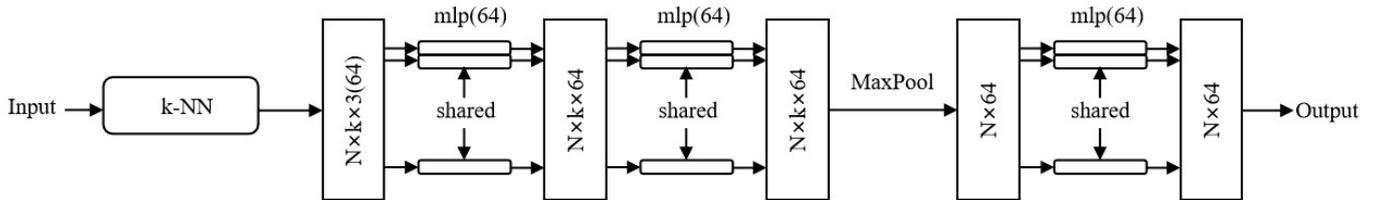


Figure 2. Overview of the GNN structure.

For the point cloud registration task, it is not enough to capture only the local features of the point cloud. In order to make the features of each point contain the information of the whole point cloud, we utilize the self-attention mechanism [33] to update the information of each point. We employ inner product calculation to assess the correlation between each point and other points in the point cloud. When two points exhibit a higher degree of correlation, their feature interaction is more pronounced. This method enables us to extend the local neighborhood feature of each point to encompass the global feature of the entire point cloud. As a result, we obtain a more comprehensive and accurate feature representation. Through this method, we can determine the importance weight of each point, which can be employed for feature fusion and selection purposes.

Specifically, as shown in Figure 3a, the input features are updated into query vector  $Q_{x_{sa}}$ , key vector  $K_{x_{sa}}$  and value vector  $V_{x_{sa}}$  through three convolution layers, respectively (Equation (2)). Additionally, the attention-based feature maps  $A_x$  are obtained as Equation (3), which is used to measure the degree of correlation between two points. In order to

prevent loss of information, we utilize the residual structure to obtain the final features (Equation (4)). Encoding is performed in exactly the same way for point cloud  $Y$ .

$$Q_{x_{sa}} = F_{input}W_{a_{sa}}, K_{x_{sa}} = F_{input}W_{b_{sa}}, V_{x_{sa}} = F_{input}W_{c_{sa}} \tag{2}$$

$$A_x = \text{softmax}(Q_{x_{sa}}K_{x_{sa}}^T), \tag{3}$$

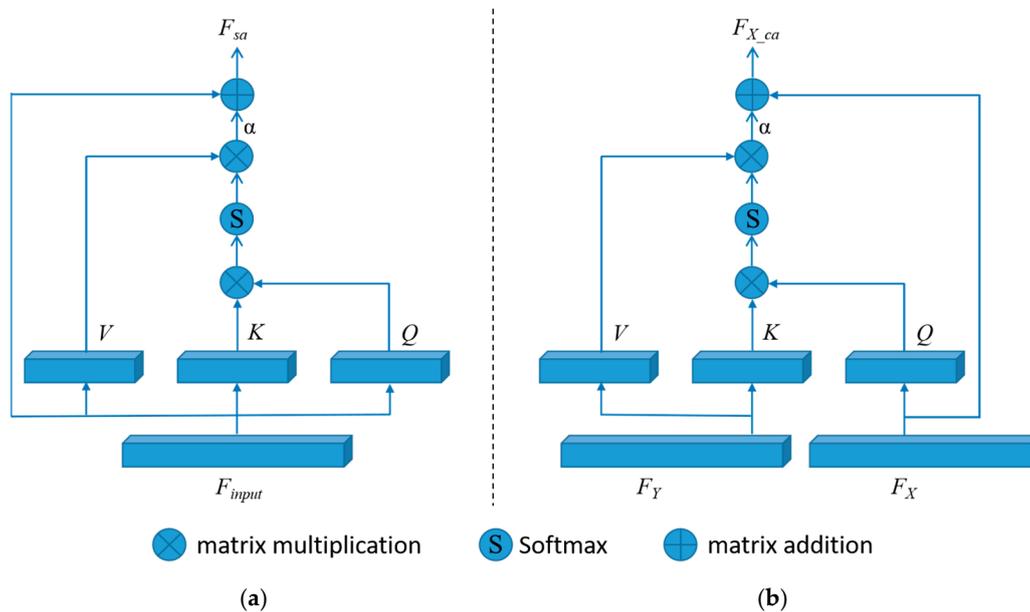
$$F_{sa} = F_{input} + \alpha V_{x_{sa}}A_x, \tag{4}$$

where  $W_{a_{sa}}$ ,  $W_{b_{sa}}$  and  $W_{c_{sa}}$  denote the weights,  $W_{a_{sa}}$  and  $W_{b_{sa}}$  are implemented using a two-layer one-dimensional convolutional neural network and  $W_{c_{sa}}$  is implemented using a four-layer one-dimensional convolutional neural network.  $\alpha$  is a learnable weight, which determines the degree of influence between points. In order to design rotation-invariant features, we utilize PPF [16] as the initial input of the network, and utilize edge convolution [27] and max-pooling to project each local PPF signature to the  $c$ -dimensional local geometric description. For a point  $x_c$  in the point cloud  $X$ , we first define a local neighborhood  $N(x_c)$  which contains points within a distance of  $r \in \mathbb{R}$  from it. Each PPF can be defined as:

$$\text{PPF}(x_c, x_i) = (\angle(n_c, \Delta x_{c,i}), \angle(n_i, \Delta x_{c,i}), \angle(n_c, n_i), \|\Delta x_{c,i}\|_2), \tag{5}$$

where  $x_i \in N(x_c)$  and  $\Delta x_{c,i}$  represents the vector between  $x_c$  and  $x_i$ , and  $n_c$  and  $n_i$  are the normals of points  $x_c$  and  $x_i$ .  $\angle$  computes the angle between two vectors  $v_1$  and  $v_2$ , which can be defined as:

$$\angle(v_1, v_2) = \text{atan2}(\|v_1 \times v_2\|_2, v_1 \cdot v_2) \tag{6}$$



**Figure 3.** Illustrations of (a) self-attention mechanism and (b) cross-attention mechanism modules.

### 3.2. Key Points and Soft Matching

In order to reduce computational complexity and identify a small number of highly correlated correspondences, it is necessary to extract a subset of points for matching purposes. However, directly using an MLP to select key points may result in the network retrieving a large number of points that are not in the overlapping regions. To address this, information exchange between the two point clouds is required prior to sampling.

By leveraging the cross-attention mechanism, the feature information from both point clouds can be exchanged and combined effectively, enabling the identification of key points that are relevant for the overlapping regions. This approach helps to alleviate the issue of fetching unnecessary points and facilitates the selection of a smaller, more relevant set of points for matching. The module structure diagram shown in Figure 3b illustrates this process. In the cross-attention module, the initial embedding consists of source point cloud features and target point cloud features. The computation of feature interaction follows the approach outlined in Equations (2)–(4).

$$Q_{x\_ca} = F_X W_{a\_ca}, K_{y\_ca} = F_Y W_{b\_ca}, V_{y\_ca} = F_Y W_{c\_ca} \quad (7)$$

$$A_{xy} = \text{softmax}(Q_{x\_ca} K_{y\_ca}^T), \quad (8)$$

$$F_{X\_ca} = F_X + \alpha V_{y\_ca} A_{xy}, \quad (9)$$

where  $W_{a\_ca}$ ,  $W_{b\_ca}$  and  $W_{c\_ca}$  denote the weights, and  $\alpha$  is a learnable weight. The updated features obtained from the cross-attention module are passed through a fully connected layer with dimensions (64, 64, 1) to compute the matching probability  $s(i)$  for each point. This step follows the original network design of IDAM [15].

To generate the matching probability matrix, we stack the updated features of the key points and include additional features such as the distance between the point clouds and the pointing unit vector between point pairs. This results in an  $M \times M \times H$  matrix, where  $M$  represents the number of key points selected from point clouds  $X$  and  $Y$ , and  $H$  denotes the number of stacked channels.

To ensure the invariance of the input order, we apply MLP on the feature vector of each correspondence, which outputs scores. These scores capture the similarity between the corresponding points in the source point cloud  $X$  and the target point cloud  $Y$ . By applying the Softmax function along each row of the  $M \times M$  score matrix, we obtain the similarity matrix  $S$ . Each element  $S_{ij}$  in this matrix represents the probability that the point  $x_i$  and the point  $y_j$  are correctly matched. To construct soft correspondences, we select the point pair relation with the maximum probability in each row of the similarity matrix. This ensures that the most likely matches are identified, allowing for accurate correspondence estimation between the two point clouds.

### 3.3. Spatial Geometric Consistency Constraint Module

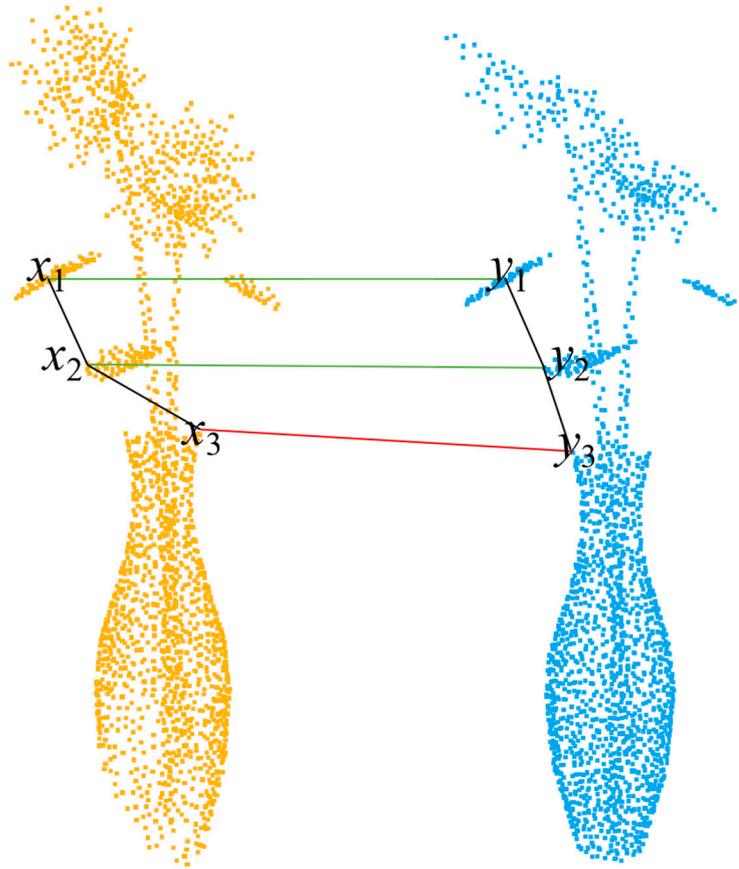
In the soft matching relationship, how to obtain the correct corresponding point pair information is a key problem. In this paper, we address this challenge by leveraging the spatial consistency provided by Euclidean transformations to eliminate incorrect correspondences. The fundamental idea is that the spatial geometric properties of a point cloud remain unchanged under rotation and translation, as depicted in Figure 4.

For instance, consider the inline point pairs  $(x_1, y_1)$  and  $(x_2, y_2)$  in their respective point clouds. These pairs maintain distance invariance despite the transformations. On the other hand, the point pair  $(x_3, y_3)$  is an incorrect correspondence due to similar features, preventing it from forming a compatible relationship with other valid inline correspondences. To establish the correct correspondences, we define  $x_i$  and  $y_i$  as a group of corresponding point pairs in the source point cloud  $X$  and target point cloud  $Y$ , respectively, and  $x_j, y_j$  represent another set of corresponding point pairs, then we can define:

$$d_{ij} = \left| \|x_i - x_j\| - \|y_i - y_j\| \right| \quad (10)$$

If the two groups are correct correspondences, the  $x_i x_j$  distance in point cloud  $X$  is consistent with the  $y_i y_j$  distance in point cloud  $Y$ , that is,  $d_{ij} = \delta$ , and  $\delta$  is an acceptable noise error (0 without noise). If one or two groups of mismatched point pairs exist, then  $d_{ij}$  is a non-regular random quantity. According to the spatial geometric consistency of the

point cloud after rotation and translation, the reciprocity between different point pairs, we can remove the wrong correspondences.



**Figure 4.** Corresponding relations between points. The green lines represent the correct correspondences, and the red line represents the error correspondence.

We first create a Euclidean spatial distance matrix  $M^d$  with dimension  $M \times M$ , where  $M_{ij}^d$  is the distance value calculated using Equation (10). Then we set a distance error limit  $\sigma$ , and utilize the relationship between  $M_{ij}^d$  and  $\sigma$  to update  $M^d$  into a matrix containing only 0 and 1. If  $M_{ij}^d \leq \sigma$ ,  $M_{ij}^d = 1$ , and otherwise 0. Taking the  $i$ th row as an example,  $\sum_{j=0}^{M-1} M_{ij}^d$  represents whether the  $i$ th point pair has more interaction with other correct correspondences. If the value of  $\sum_{j=0}^{M-1} M_{ij}^d$  is greater, the  $i$ th point pair is more likely to be the correct corresponding relationship. Finally, we select a small number of excellent corresponding point pairs and input them into the SVD module for the solution.

### 3.4. Loss Functions

The sampling of key points and the correct matching relationship are very important to the quality of point cloud registration, so that two loss functions are proposed to supervise the above two procedures separately.

**Key point loss:** This function is utilized to select the matching key points. It is difficult to label the point pair relationship in a noisy environment, so we utilize the soft match matrix for mutual supervision.

$$L_{key} = \frac{1}{M} \sum_{i=1}^M \left| s(i) - \sum_{j=1}^M S_{ij} \log(S_{ij}) \right|^2, \quad (11)$$

Correspondence loss: It is a standard cross entropy loss utilized to train the convolution module in soft correspondence. We define this loss as:

$$L_{match} = \frac{1}{M} \sum_{i=1}^M -\log(S_{ij^*}) \cdot 1 \left[ \|R^* x_i + t^* - q_{j^*}\|^2 \leq r^2 \right], \quad (12)$$

where,

$$j^* = \operatorname{argmin}_{1 \leq j \leq M} \|R^* x_i + t^* - y_j\|^2, \quad (13)$$

is the index of the point closest to the source point  $x_i$  in the target point cloud under the change of ground truth,  $R^*$  and  $t^*$  are ground truth.  $r$  is the super parameter controlling the minimum radius.

#### 4. Results

In this section, we verify and compare the performance of the proposed method through a large number of experiments, and analyze the experimental results. We compare our model with ICP [6], FGR [8], RANSAC [21], DCP [12], IDAM [15], RPMNet [14], PointNetLK [9] and Predator [34]. We also test the generalization of our model on real data. The optimization of the entire network's parameters is performed using the Adam optimizer. The initial learning rate is  $1 \times 10^{-3}$ , then we set it to  $1 \times 10^{-4}$  after 150 epochs, and 250 epochs are trained in total.

Most of our experiments are carried out on the ModelNet40 [35] dataset, which consists of 40 object categories. We utilize 9843 models for training and 2468 models for testing. Following the experimental settings of RPMNet, for a given shape, we randomly sample 1024 points to form a point cloud. We randomly generate three Euler angles within the range of  $[0, 45^\circ]$  or  $[0, 90^\circ]$ , and translations within the range of  $[-0.5, 0.5]$  for each point cloud. The original point cloud is utilized as the source and the transformed point cloud is utilized as the target point cloud.

We utilize the same metrics as [12,15] to evaluate the performance of all the methods. For the rotation matrix, we utilize root mean square error (RMSE(R)) and mean absolute error (MAE(R)). For the translation vectors, we utilize root mean square error (RMSE(t)) and mean absolute error (MAE(t)). If the overlapping regions of two clouds are exactly the same and rigid transformation is perfect, all of these error metrics should be zero, and all of the angle measurements in our results are in degrees. Since we utilize Open3D [36] to process point cloud data, it is important to note that Open3D interprets the coordinate values as meters (m) by default. Therefore, the translation errors are typically measured in meters (m) in our results.

##### 4.1. ModelNet40

###### 4.1.1. Unseen Shapes

In our first experiment, we classify all point clouds in the ModelNet40 dataset into training sets and test sets, and utilize different point clouds during training and testing. For ICP, FGR and RANSAC, we utilize the implementations in Intel Open3D [36], where the number of iterations for ICP is 30, and the search radius and the maximum number of neighborhood points of FPFH are 0.2 and 100, respectively. Since the data generation method is almost the same as that of RPMNet, in the experiment of  $[0, 45^\circ]$ , we directly utilize pretrained models of RPMNet for testing, and other experimental results are obtained after retraining.

In this experiment, the source point cloud and the target point cloud are identical and have one-to-one correspondence. In theory, the two clouds can completely overlap after rotation and translation. The experimental results are shown in Tables 1 and 2. The traditional methods are seriously influenced by the initialization, and for large rotation angles, they tend to converge to local minima. The methods based on deep learning show excellent performance when the rotation angle is within the range of  $[0, 45^\circ]$ . However,

many learning-based methods also fail when the rotation angle is too large. The high-dimensional features of the matched points after shared convolutional layers can have large gaps and seriously affect the subsequent matching of key points.

**Table 1.** Results for testing on point clouds of unseen shapes in  $[0, 45^\circ]$ .

Model	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
ICP	11.297	3.236	0.0788	0.0249
FGR	3.701	0.327	0.0171	0.0017
RANSAC	2.476	0.044	0.0072	0.0002
DCP	1.324	0.929	0.0096	0.0061
IDAM	0.086	0.044	0.0016	0.0004
RPMNet	0.241	0.026	0.0013	0.0002
PointNetLK	4.852	0.998	0.0340	0.0061
Predator	0.541	0.266	0.0064	0.0034
Ours	$<10^{-4}$	$<10^{-4}$	$<10^{-4}$	$<10^{-4}$

**Table 2.** Results for testing on point clouds of unseen shapes in  $[0, 90^\circ]$ .

Model	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
ICP	63.794	39.558	0.3113	0.1842
FGR	11.277	2.342	0.0560	0.0109
RANSAC	20.736	3.241	0.0808	0.0109
DCP	14.937	9.555	0.0962	0.0647
IDAM	0.124	0.053	0.0008	0.0003
RPMNet	3.387	0.543	0.0218	0.0030
PointNetLK	47.597	30.857	0.2785	0.1699
Predator	5.721	0.800	0.0145	0.0037
Ours	$<10^{-4}$	$<10^{-4}$	$<10^{-4}$	$<10^{-4}$

In contrast, our proposed method leverages rotation-invariant features to guide the matching task, enabling accurate selection of matching points even under large rotations. By enforcing spatial geometric consistency, we achieve an error of less than  $10^{-4}$ . Qualitative comparisons of the registration results can be found in Figure 5a.

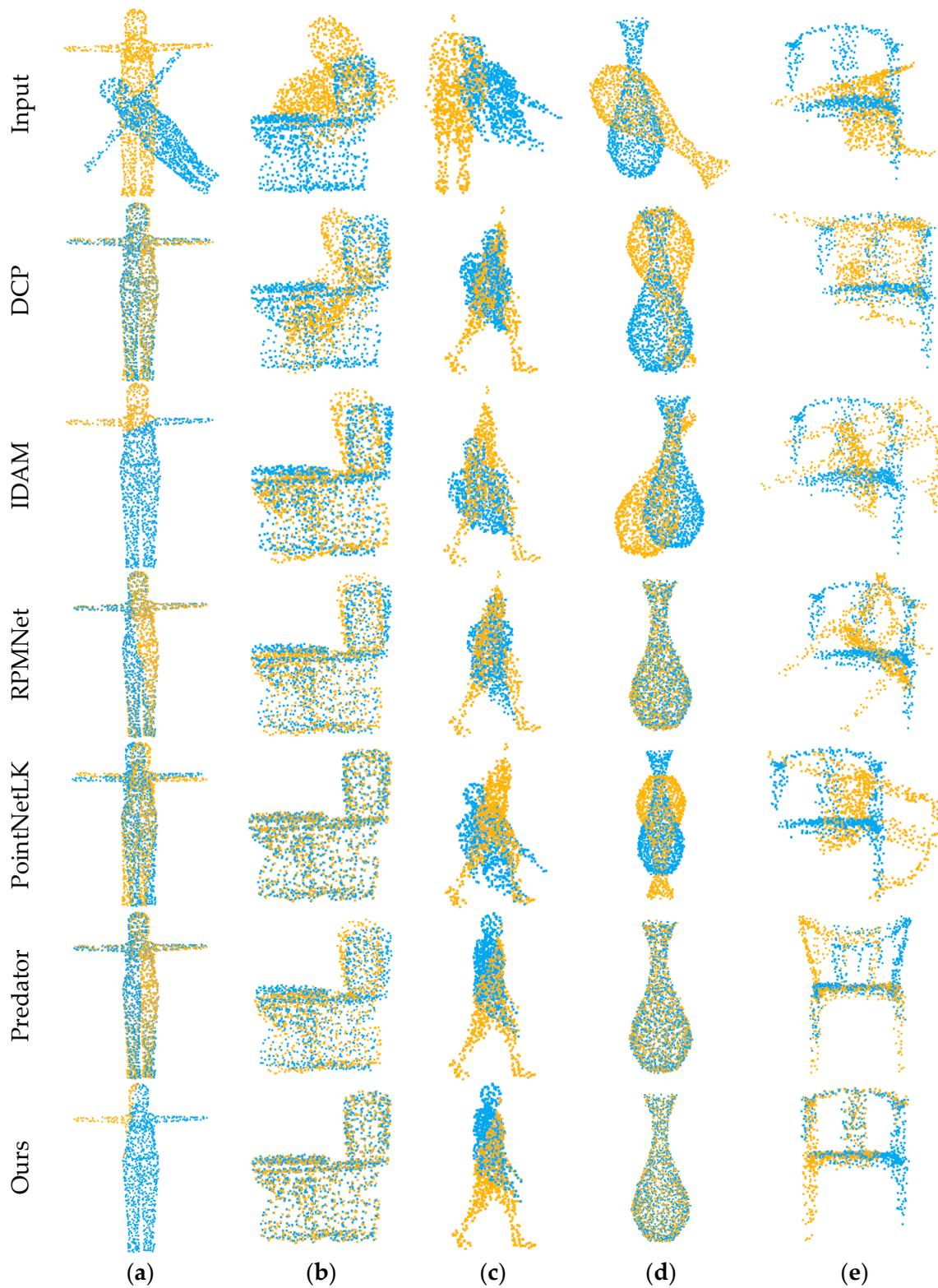
#### 4.1.2. Gaussian Noise

In this experiment, we add Gaussian noise sampled from  $N(0, 0.01^2)$  and clipped to  $[-0.05, 0.05]$  in the source and target point clouds. Since there is no one-to-one correspondence between the two point clouds, it is difficult for the network to approximate the ground truth. The experimental results are shown in Tables 3 and 4. As FPFH [22] is sensitive to noise, the errors of traditional methods such as FGR and RANSAC become large. Compared with the correspondences-free method (PointNetLK), the methods based on correspondences matching (DCP, IDAM, RPM and Predator) will become worse because of noise. This is because the methods based on global features focus on the features of the whole point cloud, not on the local features of the points. Compared with other methods, the proposed method achieves the best performance under larger rotations. A qualitative example of registration on noisy data can be found in Figure 5b,d.

#### 4.1.3. Partial Visibility

In order to generate partially overlapping point clouds, we sample a halfspace with a random direction and shift it so that approximately 70% of the points are retained for each point cloud [14]. The experimental results are shown in Tables 5 and 6. It can be seen that the errors of almost all methods will be larger, and the learning-based methods hardly converge under large rotations. Compared with these methods, although we achieved the best results, our RMSE(R) is more than four times that of MAE(R), and we try to analyze

the reasons for such result in experiment 4.2. Example results on partially visible data are shown in Figure 5c,e.



**Figure 5.** Qualitative registration examples on (a) clean data, (b) noisy data, (c) partially visible data, (d) noisy data with large rotation and (e) partially visible data with large rotation.

**Table 3.** Results for testing on point clouds of unseen shapes with Gaussian noise in  $[0, 45^\circ]$ .

Model	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
ICP	10.699	3.339	0.0749	0.0249
FGR	39.420	18.544	0.1935	0.1050
RANSAC	21.598	5.655	0.0997	0.0323
DCP	5.490	3.458	0.0382	0.0231
IDAM	3.250	1.616	0.0308	0.0158
RPMNet	1.000	0.343	0.0064	0.0032
PointNetLK	4.963	2.055	0.0352	0.0161
Predator	1.650	0.761	0.0121	0.0066
Ours	1.189	0.513	0.0128	0.0052

**Table 4.** Results for testing on point clouds of unseen shapes with Gaussian noise in  $[0, 90^\circ]$ .

Model	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
ICP	63.834	39.828	0.3115	0.1851
FGR	70.652	44.373	0.3087	0.1959
RANSAC	51.107	22.179	0.1988	0.0854
DCP	15.700	9.473	0.0964	0.0626
IDAM	13.871	5.633	0.0807	0.0359
RPMNet	6.669	1.933	0.0310	0.0114
PointNetLK	61.323	43.914	0.3228	0.2153
Predator	9.835	2.554	0.0319	0.0090
Ours	1.339	0.823	0.0147	0.0077

**Table 5.** Results for testing on partial visibility point clouds with Gaussian noise in  $[0, 45^\circ]$ .

Model	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
ICP	22.783	12.792	0.2027	0.1278
FGR	60.227	37.594	0.3130	0.2157
RANSAC	57.666	27.130	0.2552	0.1268
DCP	8.681	6.595	0.0879	0.0641
IDAM	6.093	3.892	0.0548	0.0341
RPMNet	2.350	0.893	0.0214	0.0083
PointNetLK	20.481	14.064	0.2111	0.1404
Predator	2.033	0.931	0.0233	0.0089
Ours	1.313	0.667	0.0211	0.0075

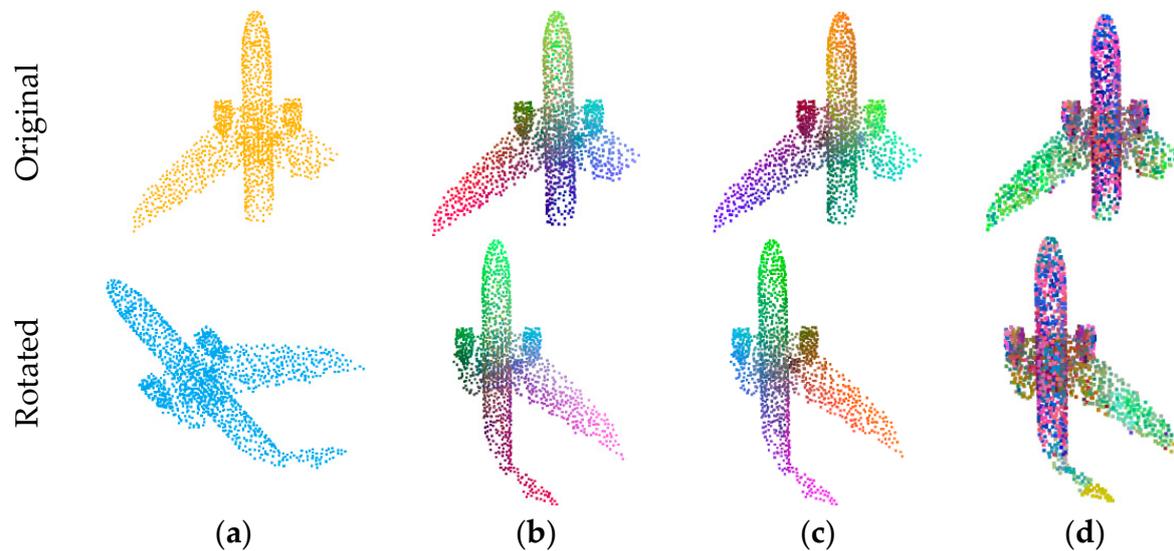
**Table 6.** Results for testing on partial visibility point clouds with Gaussian noise in  $[0, 90^\circ]$ .

Model	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
ICP	64.598	50.813	0.3567	0.2479
FGR	75.859	55.222	0.3931	0.2858
RANSAC	77.101	45.179	0.3289	0.1893
DCP	21.719	15.889	0.1882	0.1383
IDAM	16.242	8.789	0.1080	0.0586
RPMNet	9.773	3.413	0.0526	0.0227
PointNetLK	56.729	48.011	0.3802	0.2956
Predator	12.826	3.784	0.0551	0.0158
Ours	6.439	1.360	0.0414	0.0111

#### 4.2. Key points and Correspondences

In this experiment, in order to verify the validity of rotation-invariant features, we visualize the point cloud feature maps generated by PointNet, DGCNN and the method utilized in this paper, respectively. We utilize t-SNE [37] to reduce the dimension of high-dimensional features. As shown in the Figure 6, we aligned the rotated target point cloud

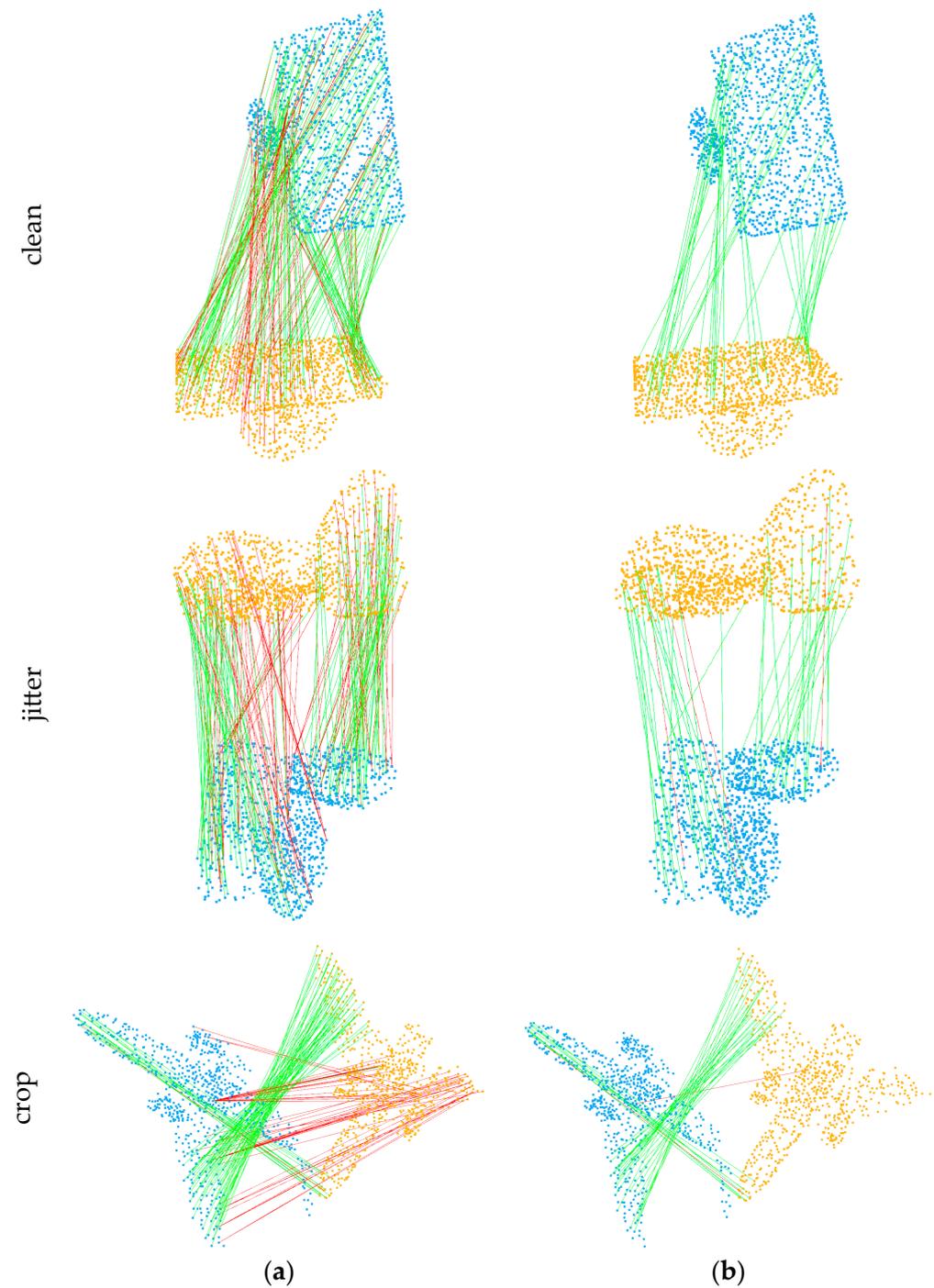
for better visualization. It can be seen that the feature descriptor we designed is invariant to rotation. DGCNN and PointNet are highly related with the input position, which is very sensitive to rotation. Different from these methods, we do not rely on the position information of individual points, but utilize the relative geometric information of the domain points to weaken the interference of rotation angle.



**Figure 6.** Illustration of the rotational invariance. We use t-SNE to visualize the learned descriptors of source and target point clouds. (a): Point Clouds, (b): Pointnet, (c): DGCNN, (d): Ours.

In order to further observe the role of feature matching and spatial consistency constraints, we also visualize the soft correspondences and hard correspondences. As shown in the Figure 7, we show the matching of points in three scenarios (clean, jitter and crop), respectively. Since all points in  $X$  have exact correspondences in  $Y$ , the corresponding points match best in the clean scenario, and the crop scenario has the most incorrect correspondences due to partial overlap and noise. Although there are a large number of outliers in the soft correspondences, the SGC module can effectively extract the correct correspondences from them. We also conduct comparative experiments between the RANSAC method and the spatial geometric consistency constraint module. We use the inlier ratio of the correspondences before and after input as a performance metric. To reject incorrect matches, we use the RANSAC method in the experiment. It is a rejection-based algorithm based on random sampling that estimates model parameters and rejects incorrect matches. Meanwhile, we also propose the spatial geometric consistency constraint module to optimize the match relationships. This module can impose spatial and geometric consistency constraints on the input match relationships and improve the inlier ratio. We compare the inlier ratio before and after using this module to evaluate its performance in optimizing match relationships. As shown in the Table 7, our proposed spatial geometric consistency constraint module outperformed the RANSAC method in most cases and performed better in rejecting incorrect matches.

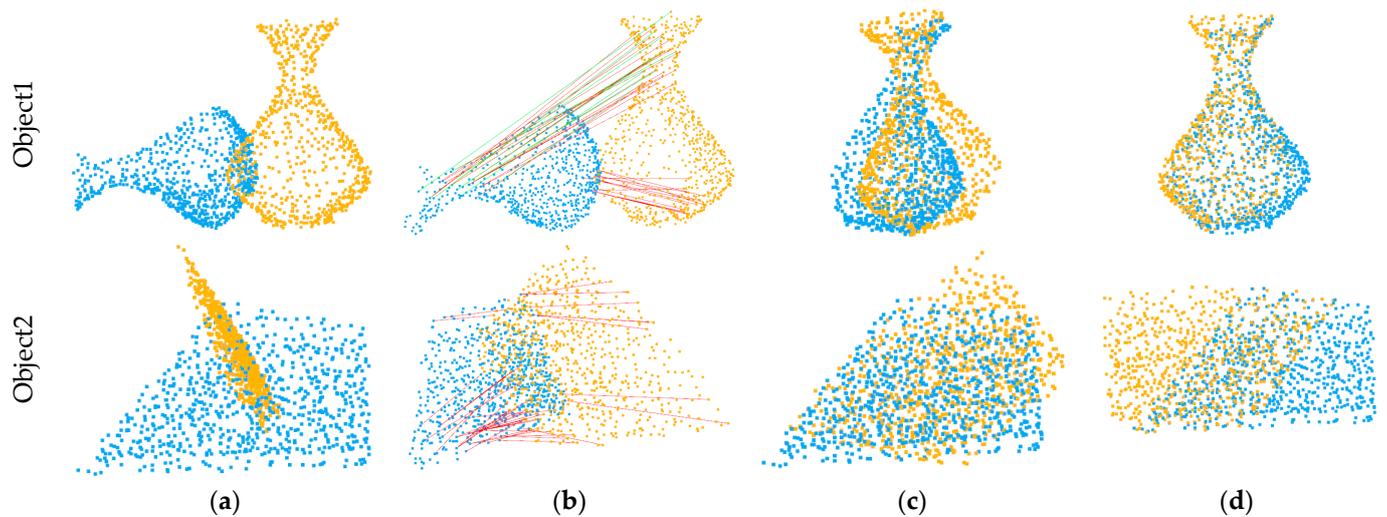
In Figure 8, we also show the situation where the network mismatches the key points due to symmetry interference. The large gap between RMSE(R) and MAE(R) indicates that there are a large number of outliers in the test data. As shown in Figure 8 and Table 8, we visualize the registration situation and corresponding relationship of bad cases. Due to the similarity of the distribution of points in the symmetric region, a large number of mismatched points have very similar features, so there are many abnormal cases in the test data.



**Figure 7.** Correspondences predicted by the network. (a): Soft correspondences, (b): Hard correspondences.

**Table 7.** Inlier ratio in correspondences under different methods.

Model	Clean	Jitter	Crop
Input	80.23%	54.34%	36.82%
RANSAC	84.88%	61.34%	46.17%
SGC	98.26%	77.31%	60.40%



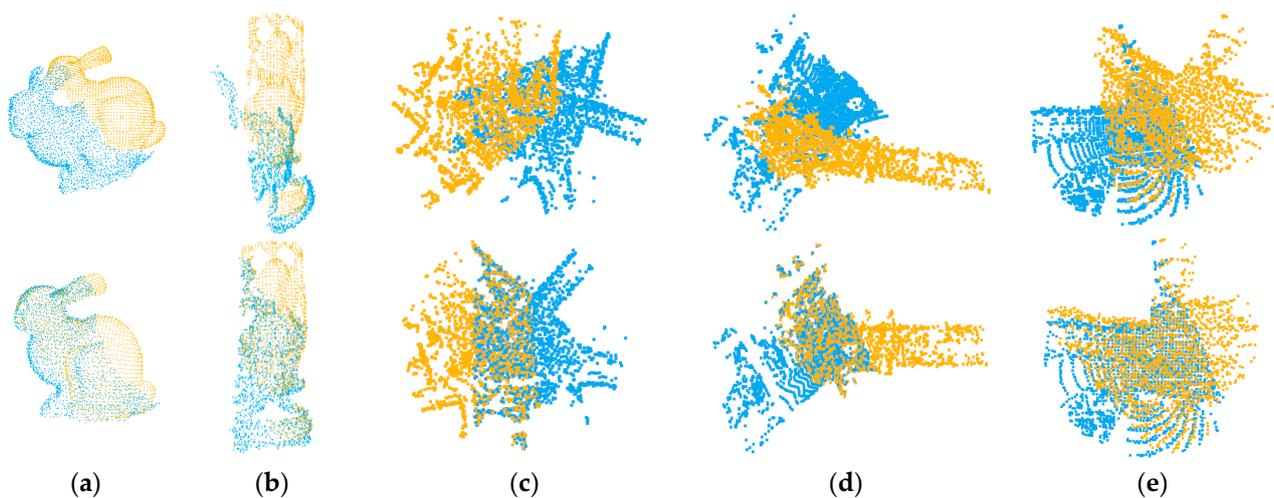
**Figure 8.** Failed cases on ModelNet40. Object 1 and Object 2 are comprised of indistinguishable surfaces. (a): Input, (b): Hard correspondences, (c): Results, (d): Ground truth.

**Table 8.** Results of failure cases on ModelNet40.

Object	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
Object1	16.950	13.826	0.1583	0.1243
Object2	26.790	19.357	0.1389	0.1372

#### 4.3. Real Data

In this section, we conduct experiments on the Stanford 3D Scan datasets [38] and odometry KITTI [39] to further evaluate the generalizability. For Stanford 3D Scan datasets, we sample 768 points on these 3D meshes separately to generate point clouds. We also downsample voxels from the original KITTI dataset to 2000–2500 points. The network parameters in this section are the weights trained in the ModelNet40 dataset without fine-tuning. The partially overlapping point clouds are generated by manner in Pnet [13]. Some qualitative examples are shown in Figure 9.



**Figure 9.** Results on the real dataset. The top row shows the initial positions of the two point clouds, and the bottom row shows the results of registration. (a,b) Stanford 3D Scan data, (c–e) KITTI data.

#### 4.4. Ablation Study

In order to demonstrate how each component affects the performance of the network, in this section, we conduct the ablation study, in which we gradually add and remove different modules in the network to evaluate their contributions to the final matching performance. The experiments are carried out on the partial visibility point clouds with noise. Tables 9 and 10 illustrate the results of ablation studies under  $[0, 45^\circ]$  and  $[0, 90^\circ]$ , respectively, where SA, CA, PPF and SGC, respectively, represent self-attention, cross-attention, deep high-dimensional features based on PPF and the spatial geometric consistency constraint module. The symbol ✓ represents the addition of a module to the network. According to the results, it is found that cross-attention can combine the information of two point clouds, which is suitable for processing partially overlapping point clouds. Additionally, the rotation-invariant feature based on PPF is effective for large rotations. In addition, the proposed correspondence module can weaken the effect of wrong correspondences and further improve the accuracy of the network.

**Table 9.** Results of ablation study in  $[0, 45^\circ]$ .

SA	CA	PPF	SGC	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
✓				5.884	3.829	0.0548	0.0334
✓	✓			4.161	2.682	0.0405	0.0231
✓	✓	✓		3.771	2.446	0.0339	0.0191
✓	✓	✓	✓	1.313	0.667	0.0211	0.0075

**Table 10.** Results of ablation study in  $[0, 90^\circ]$ .

SA	CA	PPF	SGC	RMSE(R)(deg)	MAE(R)(deg)	RMSE(t)(m)	MAE(t)(m)
✓				19.428	6.443	0.0894	0.0457
✓	✓			12.07	5.955	0.0850	0.0434
✓	✓	✓		8.336	4.215	0.0562	0.0284
✓	✓	✓	✓	6.439	1.360	0.0414	0.0111

## 5. Discussion

In comparison with other methods, our proposed approach exhibits better performance in both  $[0, 45^\circ]$  and  $[0, 90^\circ]$ , especially in the presence of large rotations, demonstrating its robustness. Further analysis of the experimental results will be conducted to discuss the advantages and limitations of our method.

In the noise-free experiment, our method achieves incredible results compared with other methods, which may seem unrealistic. This is because the target point cloud is generated by rotating and translating the source point cloud. Therefore, in the absence of noise, the two point clouds are exactly the same and have a one-to-one correspondence. Additionally, only four matched points are needed to recover the correct rigid transformation. However, in reality, data are rarely one-to-one correspondences. Nevertheless, this experiment can reflect the constraint ability of our spatial geometric consistency constraint module on outliers.

In the noisy experiment, although our method is slightly inferior to the RPM network, it demonstrates excellent performance in large rotation angles due to the supplement of rotation-invariant features. To further prove the effectiveness of this module, we conducted ablation experiments. The comparison between Tables 9 and 10 shows that the improvement of introducing the rotation-invariant module is limited in  $[0, 45^\circ]$ , but introducing this module within  $[0, 90^\circ]$  significantly improves the results, reducing the error from 12.07 to 8.34.

To validate the effectiveness of our constraint module, we compared it with RANSAC. The experimental results show that our module can significantly improve the inlier ratio

of correspondences compared with RANSAC. Furthermore, we also demonstrated the importance of the constraint module in the network through ablation experiments.

However, some experimental results show that this method has many limitations. In the partial visibility experiments in  $[0, 90^\circ]$ , the value of RMSE(R) is about 4.5 times that of MAE(R), which indicates that there are a lot of outliers in the test data. We have visualized the registration results and correspondences predicted of the bad cases. It can be seen that due to the large amount of symmetric data in the ModelNet40 dataset, there are a large number of non-matching points with similar features in the symmetric regions, which seriously affects the final registration results. It can be seen from Table 7 that when the point clouds are partially visible, there are many outliers in the soft correspondences, resulting in a low inline rate of the input correspondence. How to select the matching points with distinguishing features in indistinguishable surfaces is still a difficult problem.

Secondly, our method cannot converge when the point clouds have low overlap. Supervision and training of overlapping regions may alleviate this problem. Additionally, the proposed network cannot be directly applied to large scale datasets, because our method of feature extraction is to operate on each point, not to extract features while sampling such as Pointnet++ [40]. In future work, our focus will be on combining this work with feature extraction methods such as KPConv [41] or FCGF [42] to process large scene datasets in an end-to-end manner, and using the methods proposed in this paper to guide super-point matching and precise registration.

## 6. Conclusions

In this paper, we propose a novel network to tackle partially overlapping 3D point cloud registration. In contrast to previous works, we focus on the impact of large rotations on feature matching and the problem of feature mismatch caused by similar regions. Since large rotations can result in significant differences between the key point features of two point clouds, we introduce a high-dimensional rotation-invariant feature module in the feature extraction stage to reduce the gap between corresponding point features. Additionally, apart from incorporating self-attention mechanisms to enhance point cloud global features, we employ a cross-attention mechanism to identify overlapping regions between the two point clouds. To mitigate the impact of mismatched correspondences, we not only weight each matching point pair based on point cloud features, but also propose a non-learning module that exploits the intrinsic rotation invariance of point clouds and rejects mismatches by constraining inter-relations. Extensive experiments demonstrate that our proposed method not only achieves superior performance in the presence of large rotations but also effectively improves the proportion of correct correspondences.

**Author Contributions:** Conceptualization, W.Z., Y.Z. and J.L.; methodology, W.Z. and Y.Z.; software, W.Z.; validation, W.Z. and Y.Z.; formal analysis, W.Z.; investigation, W.Z. and J.L.; resources, W.Z. and Y.Z.; data curation, W.Z.; writing—original draft preparation, W.Z.; writing—review and editing, W.Z. and J.L.; visualization, W.Z.; supervision, Y.Z. and J.L.; project administration, Y.Z. and J.L.; funding acquisition, Y.Z. and J.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This study was supported by the Sichuan Province Science and Technology Support Program (2021YJ0080) and the Natural Foundation International Cooperation Project (61960206010).

**Data Availability Statement:** The data presented in this study are openly available in [35,38,39].

**Acknowledgments:** We are grateful to anonymous reviewers.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

ICP	Iterative Closest Point
FGR	Fast Global Registration
DCP	Deep Closest Point
PPF	Point Pair Features
RANSAC	Singular Value Decomposition
FPFH	Fast Point Feature Histogram
SHOT	Signature of Histogram of Orientation
MLP	Multi-Layer Perceptron
DGCNN	Dynamic Graph CNN
GNN	Graph Neural Networks
K-NN	K-Nearest Neighbor
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
SGC	Space Geometric Consistency
SA	Self-attention
CA	Cross-attention

## References

- Izadi, S.; Kim, D.; Hilliges, O.; Molyneaux, D.; Newcombe, R.; Kohli, P.; Shotton, J.; Hodges, S.; Freeman, D.; Davison, A.; et al. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In Proceedings of the 24th Annual ACM Symposium on Utilizer Interface Software and Technology, Santa Barbara, CA, USA, 16–19 October 2011; pp. 559–568.
- Eldefrawy, M.; King, S.A.; Starek, M. Partial Scene Reconstruction for Close Range Photogrammetry Using Deep Learning Pipeline for Region Masking. *Remote Sens.* **2022**, *14*, 3199. [[CrossRef](#)]
- Zhang, Z.; Dai, Y.; Sun, J. Deep learning based point cloud registration: An overview. *Virtual Real. Intell. Hardw.* **2020**, *2*, 222–246. [[CrossRef](#)]
- Chen, K.; Lopez, B.T.; Agha-Mohammadi, A.-A.; Mehta, A. Direct lidar odometry: Fast localization with dense point clouds. *IEEE Robot. Autom. Lett.* **2022**, *7*, 2000–2007. [[CrossRef](#)]
- Zheng, Y.; Li, Y.; Yang, S.; Lu, H. Global-PBNet: A novel point cloud registration for autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 22312–22319. [[CrossRef](#)]
- Besl, P.J.; McKay, N.D. Method for registration of 3-D shapes. In Proceedings of the Sensor Fusion IV: Control Paradigms and Data Structures, Boston, MA, USA, 12–15 November 1991; SPIE: Bellingham, WA, USA, 1992; Volume 1611, pp. 586–606.
- Yang, J.; Li, H.; Campbell, D.; Jia, Y. Go-ICP: A globally optimal solution to 3D ICP point-set registration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 2241–2254. [[CrossRef](#)] [[PubMed](#)]
- Zhou, Q.Y.; Park, J.; Koltun, V. Fast global registration. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 766–782.
- Aoki, Y.; Goforth, H.; Srivatsan, R.A.; Lucey, S. Pointnetlk: Robust & efficient point cloud registration using pointnet. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7163–7172.
- Sarode, V.; Li, X.; Goforth, H.; Aoki, Y.; Srivatsan, R.A.; Lucey, S.; Choset, H. Pcnnet: Point cloud registration network using pointnet encoding. *arXiv* **2019**, arXiv:1908.07906.
- Xu, H.; Liu, S.; Wang, G.; Liu, G.; Zeng, B. Omnet: Learning overlapping mask for partial-to-partial point cloud registration. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3132–3141.
- Wang, Y.; Solomon, J.M. Deep closest point: Learning representations for point cloud registration. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3523–3532.
- Wang, Y.; Solomon, J.M. Prnet: Self-supervised learning for partial-to-partial registration. *arXiv* **2019**, arXiv:1910.12240.
- Yew, Z.J.; Lee, G.H. Rpm-net: Robust point matching using learned features. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11824–11833.
- Li, J.; Zhang, C.; Xu, Z.; Zhou, H.; Zhang, C. Iterative distance-aware similarity matrix convolution with mutual-supervised point elimination for efficient point cloud registration. In Proceedings of the 16th European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020; pp. 378–394.
- Drost, B.; Ulrich, M.; Navab, N.; Ilic, S. Model globally, match locally: Efficient and robust 3D object recognition. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; IEEE: San Francisco, CA, USA, 2010; pp. 998–1005.
- Low, K.L. *Linear Least-Squares Optimization for Point-to-Plane ICP Surface Registration*; University of North Carolina: Chapel Hill, NC, USA, 2004; Volume 4, pp. 1–3.

18. Bouaziz, S.; Tagliasacchi, A.; Pauly, M. Sparse iterative closest point. In *Computer Graphics Forum*; Blackwell Publishing Ltd.: Oxford, UK, 2013; Volume 32, pp. 113–123.
19. Rusinkiewicz, S.; Levoy, M. Efficient variants of the ICP algorithm. In Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling, Quebec City, QC, Canada, 28 May–1 June 2001; pp. 145–152. [[CrossRef](#)]
20. Chen, Y.; Medioni, G. Object modelling by registration of multiple range images. *Image Vis. Comput.* **1992**, *10*, 145–155. [[CrossRef](#)]
21. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395. [[CrossRef](#)]
22. Rusu, R.B.; Blodow, N.; Beetz, M. Fast point feature histograms (FPFH) for 3D registration. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; IEEE: Kobe, Japan, 2009; pp. 3212–3217.
23. Tombari, F.; Salti, S.; Di Stefano, L. Unique signatures of histograms for local surface description. In Proceedings of the 11th European Conference on Computer Vision, Heraklion, Greece, 5–11 September 2010; Springer: Berlin, Heidelberg, 2010; pp. 356–369.
24. Aiger, D.; Mitra, N.J.; Cohen-Or, D. 4-points congruent sets for robust pairwise surface registration. *ACM Trans. Graph.* **2008**, *27*, 1–10. [[CrossRef](#)]
25. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
26. Lucas, B.D.; Kanade, T. An iterative image registration technique with an application to stereo vision. In Proceedings of the IJCAI'81: 7th International Joint Conference on Artificial Intelligence, Vancouver, BC, Canada, 24–28 August 1981; Volume 2, pp. 674–679.
27. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph. (Tog)* **2019**, *38*, 1–12. [[CrossRef](#)]
28. Sinkhorn, R. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Ann. Math. Stat.* **1964**, *35*, 876–879. [[CrossRef](#)]
29. Deng, H.; Birdal, T.; Ilic, S. Ppfnet: Global context aware local features for robust 3d point matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 195–205.
30. Deng, H.; Birdal, T.; Ilic, S. Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 602–618.
31. Yang, Y.; Feng, C.; Shen, Y.; Tian, D. Foldingnet: Point cloud auto-encoder via deep grid deformation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 206–215.
32. Ginzburg, D.; Raviv, D. Deep Weighted Consensus Dense Correspondence Confidence Maps for 3d Shape Registration. In Proceedings of the 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 16–19 October 2022; IEEE: Bordeaux, France, 2022; pp. 71–75.
33. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2017; pp. 5998–6008.
34. Huang, S.; Gojcic, Z.; Usvyatsov, M.; Wieser, A.; Schindler, K. Predator: Registration of 3d point clouds with low overlap. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 4267–4276.
35. Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3d shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1912–1920.
36. Zhou, Q.Y.; Park, J.; Koltun, V. Open3D: A modern library for 3D data processing. *arXiv* **2018**, arXiv:1801.09847.
37. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
38. Curless, B.; Levoy, M. A volumetric method for building complex models from range images. In Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, New Orleans, LA, USA, 4–9 August 1996; pp. 303–312.
39. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
40. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2017; p. 30.
41. Thomas, H.; Qi, C.R.; Deschaud, J.E.; Marcotegui, B.; Goulette, F.; Guibas, L.J. Kpconv: Flexible and deformable convolution for point clouds. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6411–6420.
42. Choy, C.; Park, J.; Koltun, V. Fully convolutional geometric features. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8958–8966.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.