Multimedia Systems (2024) 30:23
https://doi.org/10.1007/s00530-023-01203-w

REGULAR PAPER

Coarse registration of point cloud base on deep local extremum detection and attentive description

Haotian Lu[1] · **Jianhui Nie**[1]

Received: 11 August 2022 / Accepted: 8 December 2023 / Published online: 19 January 2024

Abstract
Coarse registration of point cloud is a necessary step for object digitization. However, insufficient overlapping, large pose difference and the existence of noise and outliers seriously reduce the result. In this paper, several improvements were made to improve the registration effect under the above conditions. Firstly, a lightweight network for feature point detection based on local extremum is proposed to improve the repeatability and robustness of feature detection; Secondly, a feature description network combined with attention mechanism is constructed to generate highly differentiated descriptors for the feature points; Finally, a transformation parameters calculation strategy based on only two feature points is proposed, which improves the success probability under low overlapping. Experiments show that our feature detection, description and registration methods achieved satisfactory results in various challenging scenes and perform better than current mainstream methods.

Keywords Point cloud · Registration · Feature point · Feature descriptors · Deep learning

1 Introduction

Point cloud registration is a key problem in 3D computer vision with wide applications including 3D reconstruction [1], pose estimation [2], simultaneous localization and mapping (SLAM) [3], and Point cloud registration technology is also involved in emerging applications such as autonomous driving, robotics and augmented reality. Generally, the process is divided into two steps, coarse registration and fine registration. Among them, coarse registration is the basis of fine registration, and its result directly determines whether the iterative closest point (ICP) algorithm [4] can converge to the correct solution. In order to achieve accurate and robust coarse registration, a lot of research has been carried out. Some of the algorithms use highly robust fitting or optimization techniques, such as RANSAC [5, 6] and fast global registration (FGR) [7, 8]; other algorithms achieve the goal by detecting and matching feature points. During the research, several feature points detection methods is developed, such as uniform sampling [9], Harris [10], ISS [11], Narf [12], etc. To match those feature points robustly, a lot of feature description methods are proposed by employing the local spatial or geometric metric relationships, such as SI [15], SCOV [16], ROPS [17] and SHOT [18]. In recent years, many researchers also proposed feature descriptors based on deep learning networks (DNN). For example, Li [19] proposed an end-to-end framework to learn local multi-view descriptors of 3D point cloud and get better results than traditional methods in most of the test scenarios. Zeng [20] proposed the 3DMatch, which builds training samples from registered RGB-D data and generates describers by Siamese Neural Network. Despite the above progress, the stability of the coarse registration algorithm under extreme conditions, such as low overlap rate or large noise, still needs to be further improved.

In order to overcome the above problems, this paper proposes new strategies in three aspects: feature point detection, feature descriptor construction and registration conditions. Specifically, the contributions of this paper are as follows:(1) A lightweight network for feature point detection based on the idea of non-maximum suppression (NMS) is proposed. Compared with the method of calculation directly on the point cloud data, our method uses a deep learning method to fit the local surface changes, which can filter out the influence of noise better. At the same time, the NMS strategy

Communicated by S. Bakshi.

✉ Jianhui Nie
njh19@njupt.edu.cn

[1] School of Automation & Artificial Intelligence, Nanjing University of Posts and Telecommunications, Nanjing, China

also prevents the phenomenon of feature point aggregation and ensures the diversity of feature points. (2) We propose a lightweight feature descriptor construction network. The integration of attention mechanism makes it be able to make full use of the information contained in the point cloud and improve the discrimination of feature descriptors, also the Siamese structure makes sample construction and network training becomes easy. (3) By introducing the concept of virtual feature points, we reduce the number of feature points required for registration from 4 to 2, which improves the success of coarse registration under the condition of low overlapping. Experiments show that the proposed algorithm can get satisfactory results in various challenging scenarios, and the overall effect is better than the latest method.

The following sections are organized as follows. Section 2 reviews relevant work. Section 3 introduces the algorithm and its implementation details. Section 4 verifies the effectiveness of the proposed algorithm through experiments and compares it with the current mainstream algorithm. Finally, the thesis is summarized in Sect. 5, and further research direction is pointed out.

2 Related work

2.1 Coarse registration

At present, point cloud coarse registration algorithms can be divided into three categories: methods based on RANSAC, methods based on feature point, and methods based on deep learning.

(1) Methods based on RANSAC

In RANSAC-based methods, some points are randomly selected from the point clouds firstly. Then, the optimal corresponding points are sifted out by judging the spatial structure consistency. Finally, the registration parameters can be calculated from the matched points. 4PCS [21] is a typical RASAC-based method. The algorithm finds matching pairs by coplanar-four-points criterion and shows good robustness. However, it consumed a lot of time to eliminate the false matching pairs, thus limiting its application. To overcome the problem, Super 4PCS [22] established an intelligent index and eliminated invalid point pairs according to the normal angle constraint, which reduces the time complexity of 4PCS to constant. Besides, the Super Generalized 4PCS [23] algorithm adds non-coplanar optimization to the intelligent index strategy, the V4PCS [24] algorithm proposes the concept of volume consistency and MSSF-4PCS [25] uses multi-scale clustering to extract point features. Although great progress has been made, due to the complex characteristics of point cloud data, the registration effect of the above methods is still not satisfactory when the point cloud symmetry is strong or the details of the overlapping region are not obvious.

(2) Methods based on feature points

This kind of method usually first calculates the significance of each point in the point cloud under some measurement index, and then identifies the points whose significance is higher than a threshold as feature points; finally, the point cloud registration is realized by constructing and matching the descriptor of the feature points.

In terms of feature point detection, Harris 3D [10, 26] can extract corners in the point cloud with high efficiency. However, in practical application, it is prone to the problem of feature points gathering together. Therefore, it is not easy to build highly differentiated descriptors for these points. The subsequent ISS [11, 27] is a feature point extraction method based on eigenvalue analysis, which has obvious geometric significance. But the principal component calculation is easy to be affected by outliers. Therefore, the robustness needs to be further improved. The NARF feature proposed by steder et al. [12] can detect the points robustly and efficiently, but is more suitable for regular depth images. When applied to irregular point cloud, the repeatability of feature points is greatly reduced. Learning discriminative features for better localizing accurate and distinct keypoints across various objects is still a challenging task. Yang et al. [13] build a large-scale and diverse dataset named KeypointNet which contains 8,234 models with 103,450 keypoints and can boost the semantic understanding of 3D objects. Subsequently, Yang et al. [14] propose a self-supervised 3D keypoint detector UKPGAN based on the GAN-based sparsity control and salient information distillation modules, which is applicable to rigid/non rigid objects and real scenes.

In terms of feature description, PFH [28] parameterizes the spatial differences between query points and neighborhood points and forms a multi-dimensional histogram to geometrically describe the nearest neighbors of points. The operator has rotation and translation invariance and is robust to sampling density change or noise. However, its computational complexity reaches o (nk^2), where n is the number of points in the point cloud and k is the number of neighborhood points used. To reduce the computational complexity, the FPFH describer [29] is proposed. By simplifying the calculation of feature histogram, FPFH reduces the time complexity to o(nk) successfully. At the same time, the histogram weighting of neighborhood points makes the algorithm can capture local features better. In the Next, Bi proposed the RICI descriptor [30], which enhances the tolerance of the algorithm to outliers. Tal et al. [31] proposed a self-rotation descriptor, which improves the discrimination of the descriptor using a more local computational scale.

At the same time, inspired by the SIFT descriptor [32] in the image field; they also proposed the local depth SIFT (LD-SIFT) descriptor [31] with rotation and scale invariance. PCEDNet [33] proposed a new parameterization and a new lightweight neural network structure, which has greatly improved the efficiency and classification ability.

(3) Methods based on deep learning

The rise of deep learning has brought new ideas to point cloud registration. Compared with the manual designed methods, methods based on deep learning can automatically find the potential laws and features in the data, so as to make full use of the original information and improve the effect of point cloud registration. In order to use the convolution network to process scattered point cloud, a common practice is to sample the point cloud to three-dimensional voxels or two-dimensional grids (such as 3DShapeNets [34], PointGrid [35], LORAX [36]), and then use convolution layers to generate a description of local geometric details. Especially, the 3DMatch [20] uses the registered depth data to make training samples, and then fits the hidden input–output relationship in the samples through a Siamese neural network, which improves the discrimination ability and robustness of the feature descriptor significantly. In order to extract feature information directly from point cloud, Qi et al. proposed PointNet [37], which processes each point independently through 1×1 convolution kennel. At the same time, it uses a symmetry function to eliminate the output change caused by the input order of the points.

Recently, the correspondences-based methods are gaining more and more attention, which constructs the correspondences for all source points without distinguishing inliers and outliers using virtual points. DCP [38] uses DGCNN [39] and Transformer [40] to learn the task-specific features. Rpm-net [41] leverage data-driven deep neural networks to learn local features from large-scale datasets. In DeepVCP [42], these virtual corresponding points are constructed based on the assumption that accurate initial motion parameters are provided as prior. Although shown to be more robust than traditional methods, they do not work well on partially visible point clouds. DCP was later extended to Prnet [43], which is a hard matching-based method and incorporates keypoint detection to handle partial visibility. However, this strategy can only work on the identified inliers and the drawback of one-to-many matching is ineluctable. [44] designs a dedicated soft-to-hard (S2H) matching procedure within the registration pipeline, which can be easily integrated with existing registration frameworks and has been verified in representative frameworks including DCP, Rpm-net. VRNet [45] constructs a pair of consistent point clouds by adjusting virtual corresponding points (vcps) to rectified virtual corresponding points (rcps) construct a pair of consistent point clouds, which effectively breaks the distribution limitation of VCPs and improves the registration performance and efficiency. SpinNet [46] propose a new neural architecture to extract local features which is rotation invariant, representative, and its descriptor achieve good results in point cloud registration. Pointdsc [47] select consistent correspondences after the initial matching to tackle the outliers. This approach is effective but complex. Predator [48] proposes an overlap attention module to handle point-cloud pairs with low overlap, but this is operationally complex as well as time-inefficient. DIP [49] presents a PointNet-based architecture for learning 3D local deep descriptors that can be used to register point clouds without requiring an initial alignment. Our network is also based on Siamese PointNet, but we add attention mechanism to the network, which enhances the expression ability of the network and obtain a better registration effect. At the same time, we simplify the network structure, making our network lighter and more efficient. Through experimental comparison, the registration rate of our network is as high as 97.08%, which is about 10% higher than DIP [49] and Predator [48], 30% higher than 3DMatch [20] and a traditional method [27] using ISS and FPFH.

2.2 Attention mechanism

Attention plays an important role in human perception [50–52]. When people see a scene, they will involuntarily choose the most important part to watch. Attention mechanism comes from the simulation of human vision. It allows the computer to efficiently and accurately screen out the useful message from the massive information. With the continuous progress of deep learning technology, the attention mechanism has been widely used. For example, Wang et al. [53] proposed the residual attention network, which uses an encoder-decoder style attention module. By refining the feature map, the network not only performs well on clean data, but also is robust to noisy inputs. Hu et al. [54] proposed the SE-Net, in which a new SE module is used. The module has a simple structure and strong universality. It can be embedded into any existing network, and the consumed computing resources only need to be increased by less than 10%. CBAM [55] is another popular attention module for convolution networks, which combines channel attention and spatial attention in series. Spatial attention allows the neural network to transform the spatial information in the original picture into another space while retain the key information. Channel attention can change the proportion of weight distribution between convolution channels to give more play to the network efficiency. PointNet extracts a global feature when extracting data features, thus ignoring local features and the differences between feature channels. This may cause the network to ignore potential relationships among feature information of point clouds. To solve the above

problems, this paper adds a channel attention mechanism to the network structure to fully extract the feature information contained in point clouds. At the same time, different weights are assigned to different feature information, which improves the network's ability to extract feature information.

3 Our algorithm

This paper presents a point cloud coarse registration method combining local extremum point extraction and deep learning feature descriptor. Specifically, we first extract curvature extremum points from source and target point clouds as feature points, respectively; then, we put the feature points into a lightweight attention based feature description network to generate the descriptor; finally, false matching pairs are filtered by RANSAC and the registration is completed with at least 2 matching pairs with the help of virtual feature points. The flowchart of the specific algorithm is shown in Fig. 1.

In this paper, we first choose curvature extremum points as feature points in (b). As curvature extremum points locate at the places with the most drastic surface changes in the model, it contains rich surface information, which is beneficial for generating feature descriptors. We regard it as the screening criterion and train a very lightweight net to detect the feature points. We also use a Siamese network with attention mechanism as the feature descriptor generator in (c), which can automatically find the most effective feature description strategy from a large number of samples, and effectively improve the discrimination of feature descriptors. Finally, by introducing virtual feature points, we reduce the minimum number of matching points required for registration from 4 to 2 in (d), which can effectively improve the registration efficiency and the registration success rate in the case of a low overlapping rate.

3.1 Curvature extremum points detection

(1) Theory

Curvature is a direct measurement of the local change of the surface. Generally, the greater the curvature, the richer the surface information contained. Therefore, if the curvature extreme points are used as feature points, they can encode the most surface information and improve the discrimination of feature descriptors. Following the idea of non-maximum suppression, this paper first calculates the gauss curvature value of each point in the point cloud, then fits the curvature distribution of the local neighborhood through a quadric surface, and finally recognizes the curvature extreme point by comparing the distance between the

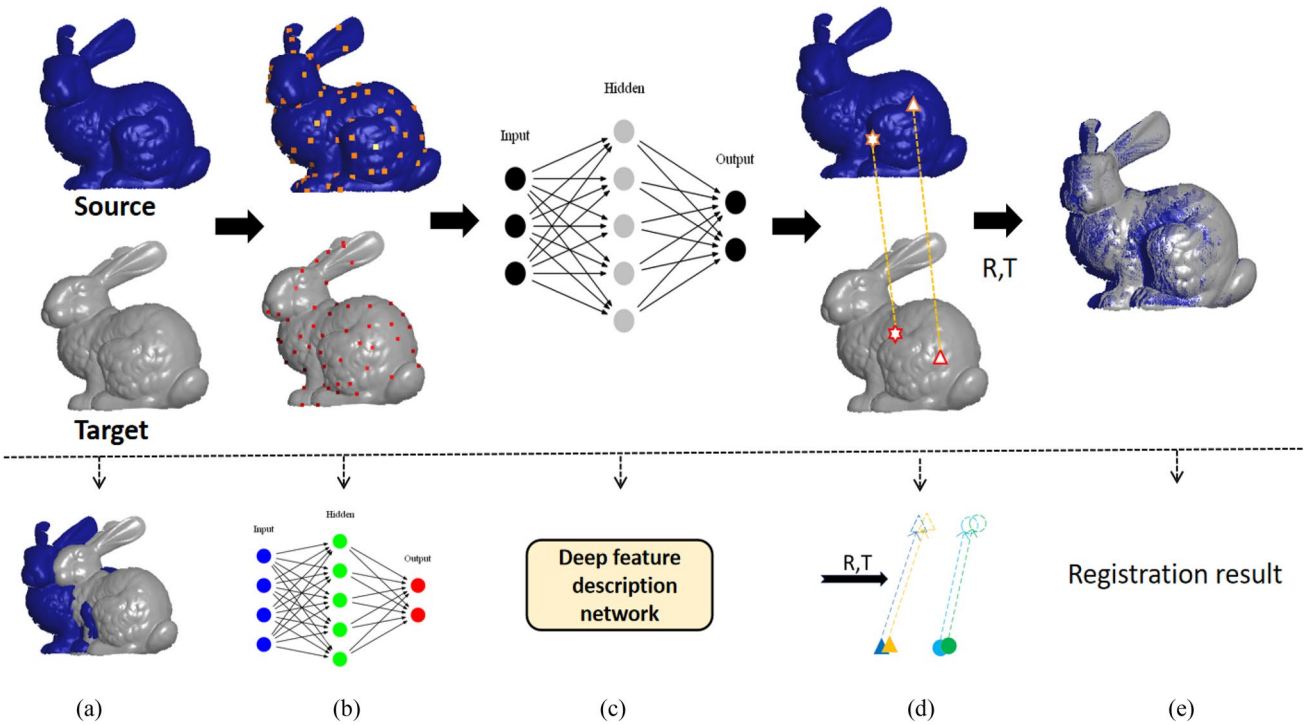


Fig. 1 Pipeline of our algorithm. Given the source and target point cloud (**a**), we first extract curvature extremum points using our feature point detection network (**b**);then we use our deep feature description network to generate the descriptors of feature points and find matching points (**c**); finally, registration is completed with the help of virtual feature points (**d**); the result of registration (**e**)

curvature extreme point from the quadric surface and the current point. The specific process is as follows:

(1) Create a Local Coordinate System (LCS) with the current point as the origin, its normal as the z-axis, and the maximum principal curvature direction as the x-axis. Then, translate the current point and its r neighborhoods to the LCS;
(2) Construct the objective equation in Eq. (1) and solve the parameter $b_0 \sim b_5$, where n is the number of neighborhood points, x_i, y_i are the x, y coordinate component of the neighborhood points under the LCS and c_i is the gauss curvature of neighborhood points;

$$\arg\min \sum_{i=1}^{n} \left(b_0 x_i^2 + b_1 y_i^2 + b_2 x_i y_i + b_3 x_i + b_4 y_i + b_5 - c_i\right)^2 \tag{1}$$

(3) Put $y=0$ into Eq. (1) to obtain the curvature curve at the maximum principal curvature direction as $f(x) = b_0 x^2 + b_3 x + b_5$, then calculate the extremal coordinate of $f(x)$ as $x_{\max} = -b_3 / (2b_0)$;
(4) Put $y=0$ into Eq. (1) to obtain the curvature curve at the maximum principal curvature direction as $f(y) = b_1 y^2 + b_4 y + b_5$, then calculate the extremal coordinate of $f(y)$ as $y_{\max} = -b_4 / (2b_1)$;
(5) Calculate the distance between the current point p and the curvature extremal point as $l = \left\| (x_{\max}, y_{\max}) \right\|$. If l is less than the average sampling density ρ of the point cloud, then identify it as a feature point; otherwise, p is far away from the real feature point and identified as a general point;
(6) Project the extremal point onto the point cloud by MLS surface [56] to obtain its accurate position.

In order to prevent the aggregation of extreme points, which will adversely affect the discrimination of feature descriptors, we further screen the initial feature points by using the idea of NMS, that is, we only retain the point with the largest curvature in the local neighborhood as the final feature point. However, it should be pointed out that the curvature is a second-order differential of a continuous surface and it is difficult to be estimated accurately in the case of data missing or noise using traditional methods. This also makes it difficult to detect the curvature extreme points on the noise model, which will greatly reduce the repeatability of feature points. Thanks to the progress of deep learning, it brings an opportunity to the development of feature detection. Different from the traditional methods, deep learning can improve the robustness of feature detection through the diversity of training data.

(2) Feature point detection network

The computational cost of the traditional algorithm for detecting feature points is relatively small, but the accuracy is reduced due to noise and other factors in the non-ideal case. Therefore, the main challenge is how to extract enough expressive features for each point to make the model more accurate while ensuring that the computational cost is not too large. Aiming at this challenge, we take the above theory as the screening criterion, and train a lightweight network as a feature point detector. The specific structure of the network is shown in Fig. 2. In this network, we take the position and curvature of the point cloud as input and extract the features through 3 Convolution layers and a Max-pooling layer, then use the full connection layer to map the learned features to the sample label space. The last layer of the network is the softmax layer, which is used to map the features of the last fully connected layer to a score vector.

For the implementation details of our detection task, we added curvature information to help the network detect feature points better, i.e., taking (x, y, z, curvature) as input. The output of the network is a two-dimensional score vector, representing the probabilities of two categories (feature points or not), respectively. Finally, we screen the required

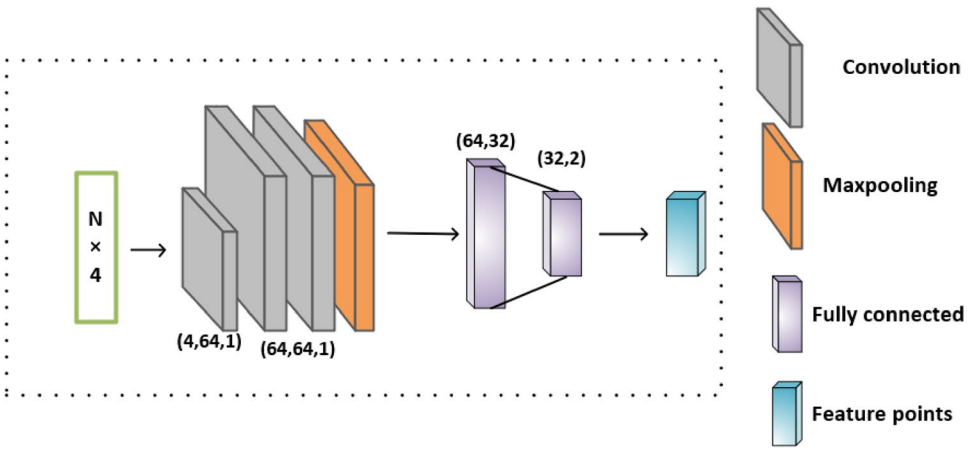


Fig. 2 Feature point detection network

feature points by setting a probability threshold. In training, since we regard this task as a classification problem, we choose cross-entropy as the loss function which is often used to evaluate the performance of a classification model. We also added different levels of noise and random rotation to the samples to increase the diversity of samples. Subsequent experiments show that those strategy is effective, which makes our feature point detection network receives better repeatability and robustness than the current mainstream algorithms.

3.2 Deep feature description network

In order to process the scattered point cloud directly, our feature description network uses PointNet as the backbone, which solves the problem of point cloud disorder by using a symmetric function. Also, it uses a spatial transformation network to solve the network output changing caused by point cloud pose variation.

Nevertheless, PointNet was originally designed for point cloud classification, so its feature description is not optimal for point cloud registration, that is, the direct use of PointNet structure cannot get the optimal feature descriptor. Therefore, as shown in Fig. 3, we modify the network structure to be Siamesed, and use the Contrastive Loss to fine tune the network parameters, so as to make the generated feature descriptor meet the needs of point cloud registration. At the same time, as the output of the network becomes binarized, the sample preparation process can be greatly simplified, and the difficulty of network training can also be greatly reduced. Specifically, we take the feature points(x, y, z) mentioned above as the input, finally generate a 1×1024 descriptor for finding matching points. During training, we input the descriptor into the Contrastive Loss for network's learning.

It is noteworthy that compared with point cloud classification, the local attribute of feature description is more obvious, that is, the required neighborhood points to generate feature is much less and the solution space range required to search is much smaller. Therefore, as shown in Fig. 4, we simplified the MLP part of the original network. Specifically, our network has reduced four full connection layers, three batch normalization layers and two drop out layers. Experiments show that the above modifications do not affect the discrimination of feature description.

In order to make the network be able to capture the channel differences, we insert an attention module at the last of each Siamese branch, that is, the CBAM module mentioned above. The original CBAM module contains two parts: the channel and the spatial attention. In the channel attention module, the network compresses the feature map in the spatial dimension, obtains a one-dimensional vector, so as to obtain the channel weighted vector and extract important channel information. As CBAM is a lightweight module, the computing resource consumption of this module is almost negligible. On the other hand, spatial attention pays more attention to the spatial information of data. As mentioned before, the T-Net has eliminated the impact of pose changing, therefore, spatial attention has little improvement to our network. More importantly, spatial attention will highlight the main characteristics of the data while ignore the neighborhood characteristics, which will have a negative impact on the feature description. Therefore, we dropped the spatial attention mechanism and only used the channel attention in our network.

After the above modifications, the final structure of our network is shown in Fig. 3. In the Siamese part of the network, the two branches deal with the feature points from the source point cloud and the target point cloud, respectively. Specifically, the local neighborhood data set of each feature point $(x_1, x_2, ...x_n)$ is mapped into a one-dimensional vector after a multi-layer representation network, as shown in Eq. (2):

$$F = f(x_1, x_2, ...x_n) = \gamma\left(\max_{i=1,...n}\{h(x_i)\}\right), \tag{2}$$

where f represents the mapping from a point set to the vector, which is invariance to the order of the input points, F is the result vector, γ and h_* are continuous functions, which represent multi-layer perceptron networks.

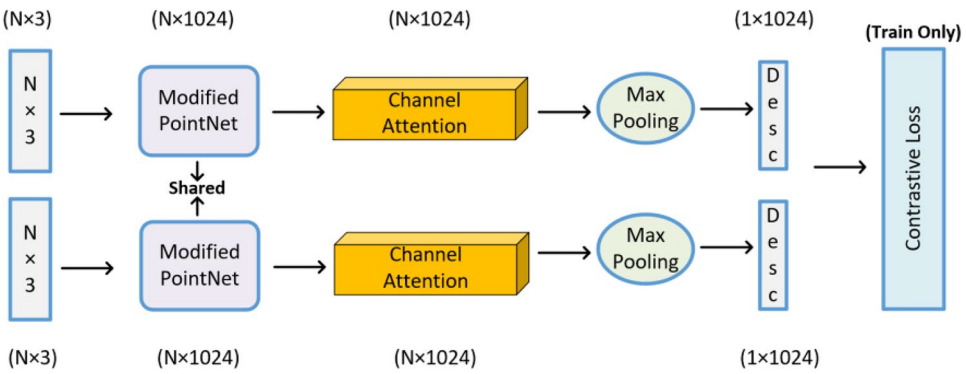


Fig. 3 Our deep feature description network

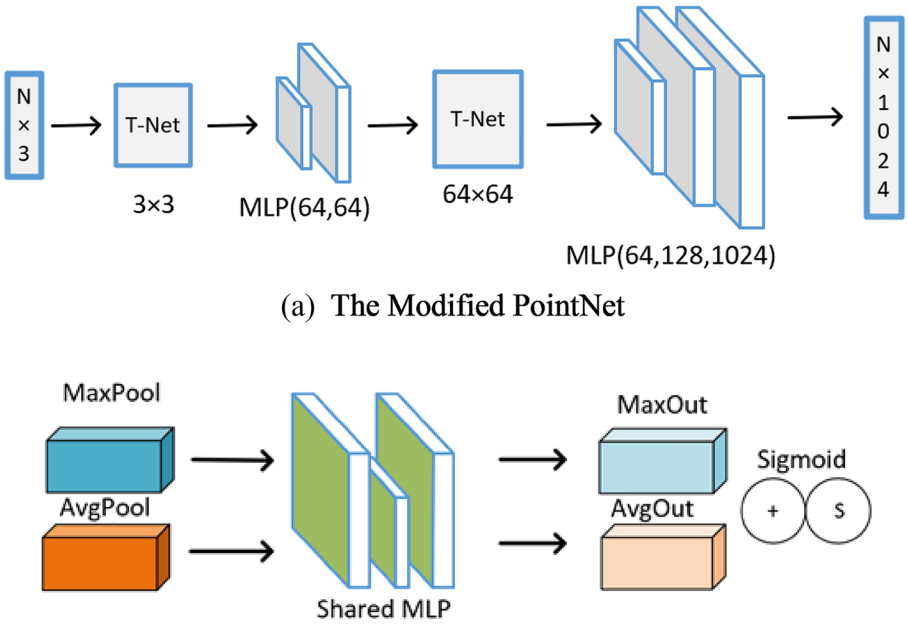


Fig. 4 The modified PointNet and Channel Attention

Next, we project the feature vector F into the channel attention module to generate a new feature map to emphasize the differences of channels. The process is shown in the Eq. (3):

$$\begin{aligned} M_C(F) &= s(MLP(\text{AvgPool}(F)) + MLP(\text{MaxPool}(F))) \\ &= s\left(W_1\left(W_0\left(F^C_{\text{avg}}\right)\right) + W_1\left(W_0\left(F^C_{\text{max}}\right)\right)\right), \end{aligned} \tag{3}$$

where $M_C(F)$ is the output of the attention module, $W_0 \in R^{C/r \times C}$, $W_1 \in R^{C \times \frac{C}{r}}$, σ represents sigmoid, r is the reduction ratio.

It can be seen from the equation that the feature vector passes through the average pool layer and the maximum pool layer in parallel, then passes through a shared MLP, and finally generates a 1024-dimensional feature description vector through sigmoid processing and maximum pool.

3.3 Registration with virtual feature points

In order to ensure the stability of the calculation, it usually needs at least four pairs of matching points to get the registration parameters for traditional methods. However, when the overlapping of two point clouds is too small or the matching is disturbed by noise and outliers, no enough matching points can be found. In this paper, we propose the concept of virtual feature points. As shown in Fig. 5a, b, ▲and ●are the two real matching points. In order to complete the calculation of registration parameters, we move these matching points along their normal vector for a distance d to generate virtual matching points (represented by dotted lines). Finally, with the help of virtual corresponding points, the number of matching points meets the requirements, and then the registration parameters can be calculated.

At the same time, a fewer matching points means fewer sampling in the RANSAC algorithm. According to the sampling formula as shown in Eq. (4), if we set the sampling success confidence τ to 0.99 and the proportion w of real matching points in the initial result to 50%, then in classical RANSAC, the minimum number of samples n is 4, that is, the required sampling times k is 71. Instead, in virtual corresponding point algorithm, n is 2, and the number of sampling required is 16, which shows that our algorithm is more efficient than traditional methods.

$$k = \log(1-t)/\log(1-w^n). \tag{4}$$

Based on the above analysis, the calculation process of registration parameters combined with virtual corresponding points and RANSAC is as follows:

(1) Randomly select two pairs of matching feature points in the source point set and the target point set, and then move the two points along their normal vector for a distance of 10 ρ, where ρ is the average sample density.
(2) Check the distance between the virtual corresponding points and the distance between the real corresponding points. If the difference exceeds 3 ρ, it indicates that they are not real matching points, and return to step (1).

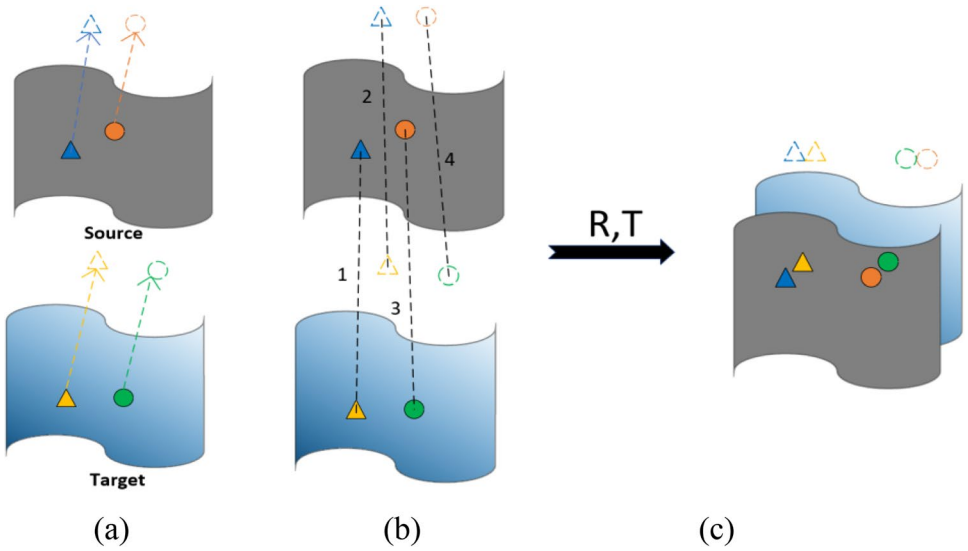


Fig. 5 Registration with virtual feature points. **a** Given two pairs of real matching points in source and target point cloud and then generate virtual corresponding feature points. **b** Using RANSAC for registration. **c** Real and virtual matching points after registration

(3) Calculate the registration parameter R|t using the corresponding points, and transform all feature points in source point cloud by R|t.
(4) Search as many matching pairs as possible in the remaining feature points. That is, if the distance between the feature points in the source point cloud and a feature point in the target point cloud is less than 3 ρ, it will be added to the matching point set.
(5) Optimize registration parameters using all matching point pairs found.
(6) Repeat step (1–5) according the Eq. (4), and select the one that finds the most corresponding points as the criterion to obtain the optimal transformation parameters.

It is noteworthy that even if the accuracy of normal vector may be affected by noise, our method is not expected to be so since the major focus is on the consistency of normal vector. By using a larger neighborhood to calculate the normal vector, we can ensure that the consistency of the normal vector is not affected by the noise and improve the efficiency in the meantime. Although this may lead to the increase of absolute error in numerical value compared to the general method, it will not affect the final result as long as the consistency of normal vector is maintained..

4 Experiments

This section verified the algorithm in this paper from the aspects of repeatability of feature extraction, ablation experiment for feature point detection, discrimination of descriptors, universality of the algorithm, etc. The algorithm is implemented in pytorch and tested on a PC with Intel Core i7 8700 CPU@3.2 GHz, 16GB RAM and NVDIA GTX 1080Ti GPU. When testing the alignment effect, we mainly use the data set from the Stanford 3D scan repository, in addition to data sets from [20] and [57], and some models from ModelNet40 [58] and the KITTI outdoor LiDAR dataset [59].

For training details, we train our feature point detection network/feature description network using Stochastic Gradient Descent for 50/70 epochs, with initial learning rate 0.01, momentum 0.9, and weight decay 10^{-4}. The learning rate is exponentially decayed by 0.5/0.1 every 10 epochs. To ensure the smooth form in the loss trajectory, we use batch size 128 and 64, respectively.

4.1 Repeatability of feature points

In order to fully verify the effect of our algorithm, we selected four models (armadillo, bunny, dragon and Buddha) as the test objects and take repeatability as a metric to measure the results. Specifically, we calculate the distance between each pair of adjacent points and take 5 ρ(the average sampling density of a point cloud) as threshold. If the position draft after noise is added is less than the threshold, the feature point is recognized as repeatable. We report the percentage of repeatable in Table 1.

To simulate the real data, we add the noise along the normal vector, that is, along the surface of the object (coordinate z axis), which produces the maximum error. In the

Table 1 Average repeatability

Repeatability	Dragon	Armadillo	Bunny	Happy
ISS	49.23	72.87	82.2	59.26
Harris	65.95	76.83	98.13	45.9
UKPGAN	**67.22**	80.91	**99.16**	67.75
Ours	58.86	**83.41**	98.94	**69.69**

experiments, we test repeatability by adding 0.5 ρ noise to the point cloud models and comparing the feature position offset with that before adding. Also, we compared our results with three well-known feature point detectors, ISS, Harris and UKPGAN [14].

Figure 6 shows the experimental results of the armadillo model. The blue points in the figure are the feature points detected without noise, and the purple points are the feature points detected with 0.5 ρ noise added to the model. Statistic shows that the repetition rate of our algorithm is 83.41%, that of UKPGAN is 80.91%, that of Harris/ISS is 76.83%/72.87%, which means our method is robust and stable.

Using the same criteria, we tested the repeatability of feature detection on the other three models, and the qualitative results are shown in Figs. 7, 8 and 9, respectively. Statistics show that the average repeatability of our algorithm is 17% higher than that of ISS, 10% higher than that of Harris and as accurate as UKPGAN. The reason for this is that, both ISS and Harris methods calculate directly on the discrete point cloud. Therefore, the surface information at non-sampling points cannot be used, and the change of sampling points caused by noise is more likely to affect the stability of calculation. UKPGAN perform better than ISS and Harris thanks to estimating local reference frame (LRF). Its salient information distillation can force UKPGAN to

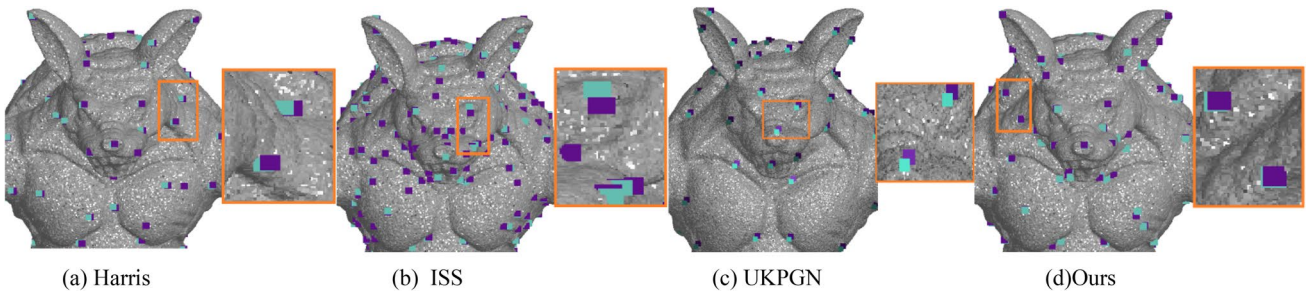


Fig. 6 Feature point drift under 0.5 ρ noise for the Armadillo model

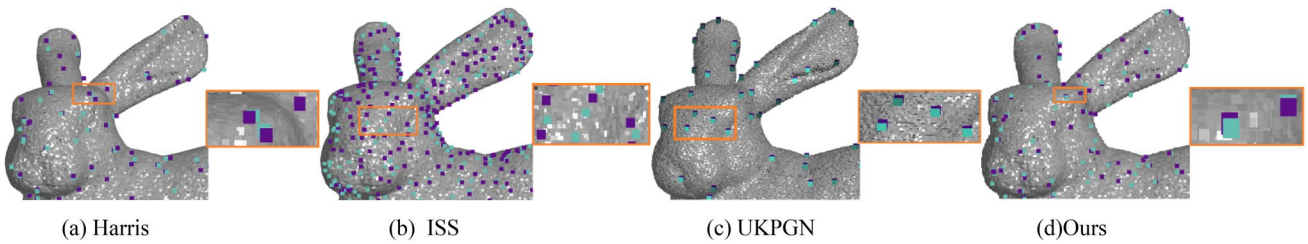


Fig. 7 Feature point drift under 0.5 ρ noise for the Bunny model

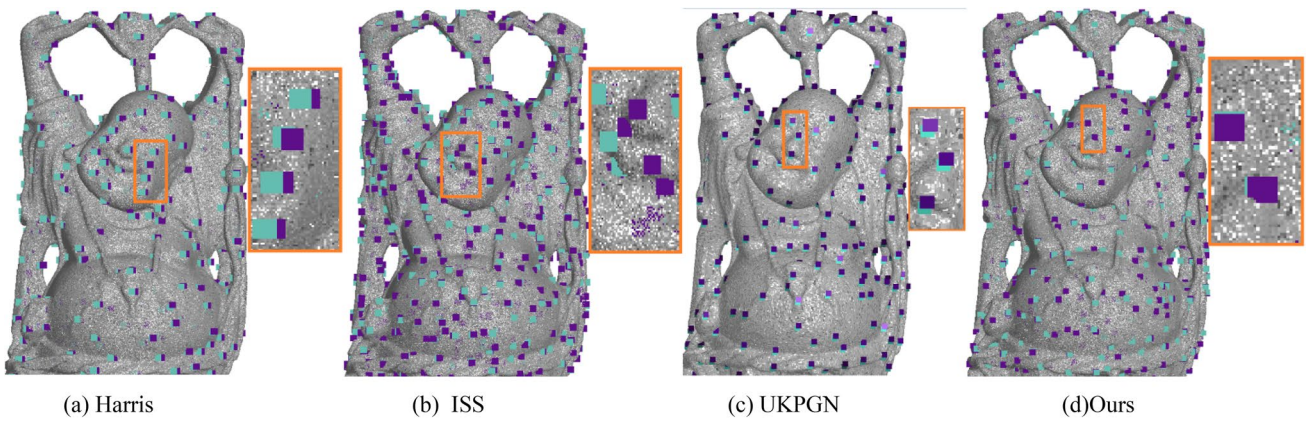


Fig. 8 Feature point drift under 0.5 ρ noise for the Buddha model

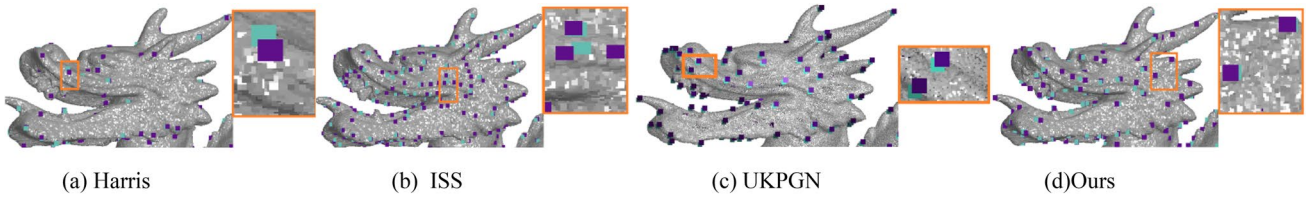


Fig. 9 Feature point drift under 0.5 ρ noise for the dragon model

extract irrelevant points of point cloud models so that it can achieves high repeatability on many models. However, the focus of its training was not the model with rich surface feature information, which resulted in his performance in model armadillo and Buddha being inferior to ours. We use a continuous surface to fit the local curvature distribution, which can make more comprehensive use of the potential surface information. Also, the least square criterion used in the fitting can further filter out the influence of noise. More importantly, out lightweight network is much simpler than UKPGAN, which greatly improves the efficiency of subsequent registration. Consequently, out method outperforms UKPGAN and is significantly better than Harris and ISS (Fig. 10).

4.2 Ablation experiment

4.2.1 Feature detection network

In an attempt to ensure that our detection network has the highest accuracy and the minimum network computation, we compared the detection accuracy of the network with and without transformation matrix, and when the number and dimensions of convolutional layers are different.

Accuracy and recall are common measures to determine the validity of a detection network. They are defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

Here, TP, FP and FN are the true positives, false positives and false negatives, respectively.

PointNet set up two layers of STN network to solve the problem of invariance under transformations. They are designed to adjust the position of point cloud in space and for the alignment of features, respectively. PointNet's original task was to classify 3D objects, while we classify points in the point cloud model, so the features of each point can be extracted well without adding the transformation matrix. We specifically selected the armadillo model and add 5 ρ noise for ablation experiments. Specifically, we take the feature points under the clean point cloud as the benchmark and enumerate seven schemes for comparison. Among them, scheme 7 is to calculate the curvature extremum points with the traditional algorithm and screen the feature points with the idea of NMS (so-called NMS).

As shown in Table 2, the first three schemes contain STN and convolutional layers, while the last three schemes only contain convolutional layers. The sizes of the convolutional layers a, b, c and d are $4\times64\times1$,$64\times64\times1$,$64\times64\times1$,$64\times$

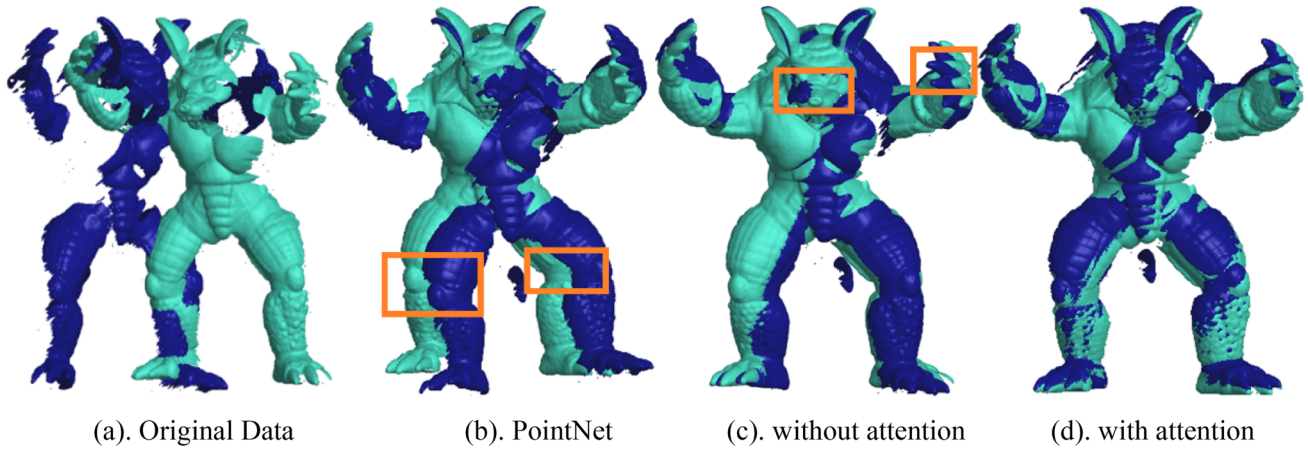


Fig. 10 Registration result with describers from original PointNet (**b**), our feature description network without (**c**) and with attention module (**d**)

Table 2 Precision and recall

Scheme		Precision	Recall
1	STN + 4conv(a, b, c, d)	0.79	0.76
2	STN + 3conv(a, b, c)	**0.85**	0.93
3	STN + 2conv(a, b)	0.74	0.78
4	Only 4conv(a,b, c, d)	0.8	0.88
5	Only 3conv(a, b, c)	0.83	**0.94**
6	Only 2conv(a, b)	0.79	0.8
7	NMS	0.77	0.79

$64 \times 1, 64 \times 128 \times 1$, respectively. We can see from the table that deep learning schemes performed better than traditional methods, and the scheme with three convolutional layers has the best results. The precision of scheme 2 is slightly better than that of scheme 5, but the recall is not as good as the latter. Moreover, since scheme 2 contains two STN networks, its computation is much larger. In summary, we choose the scheme 5 as our final feature point detection model.

4.2.2 Channel attention

To achieve the best registration effects, we registered the Armadillo model with original PointNet, our revised feature description network without/with attention module as backbone, respectively. As shown in Fig. 7, when using original training weight of the PointNet, there is a large deviation on the model legs, and the overlap rate is only 64.15% after the registration, which shows that the PointNet is not suitable for point cloud registration before modification. After we simplified the MLP layer and turned the network into a Siamese structure, the generated descriptors become more in line with the needs of point cloud registration, and the registration effect was greatly improved, with an overlap rate of 94.04%. Finally, when the attention module is added, the network learns the differences between channels, and the registration rate is improved to 99.41%, which verified the effect of the attention module.

4.3 Discrimination of descriptors

In order to verify the discrimination of feature descriptors, we first extract 500 curvature extreme points from the registered data as feature points; then, the feature descriptors are calculated by using the source point cloud and target point cloud before registration. Finally, we establish the matching relationship of feature descriptors through KD tree, and count the proportion of correctly matched feature points to the overall feature points (we call this criterion Fetch-Ratio later). Obviously, the better the discrimination of feature descriptors, the more feature points can be correctly matched.

Table 3 Fetch-ratio

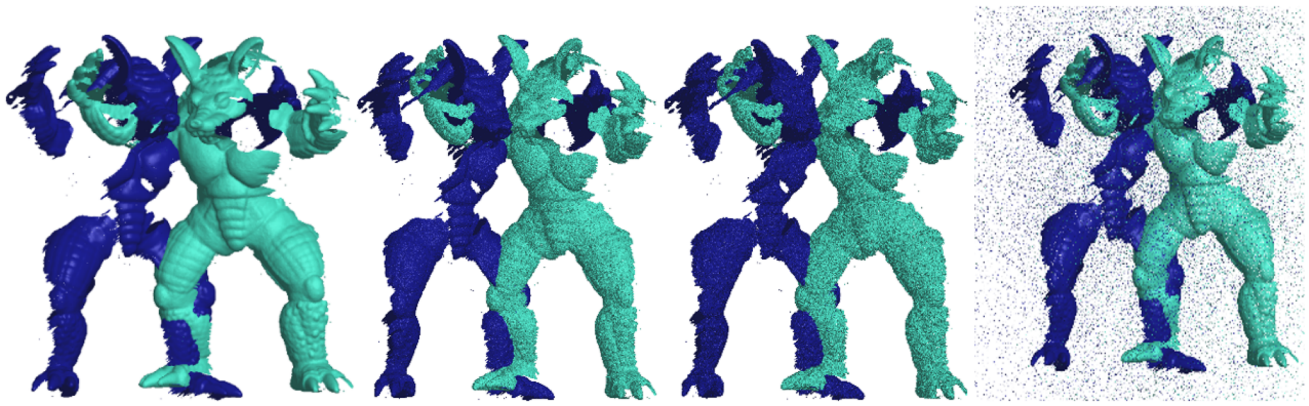
Fetch-ratio	3DMatch %	DIP %	Ours %
Armadilo5 ρ	37.76	69.12	71.43
Armadilo10 ρ	27.55	30.89	33.67
Armadilo20 ρ	19.39	18.76	19.39
Bunny5 ρ	11.76	36.14	33.33
Bunny10 ρ	13.73	17.25	17.65
Bunny20 ρ	7.84	6.54	11.76
Dragon5 ρ	25	55.23	50.78
Dragon10 ρ	19.53	20.58	26.56
Dragon20 ρ	14.06	15.74	18.75
Buddha5 ρ	41.18	70.63	80.21
Buddha10 ρ	46.52	41.06	51.34
Buddha20 ρ	21.39	19.11	24.6

In Table 3, we list the Fetch-Ratio of the four models under different noise levels and compare with 3DMatch and DIP [49]. Horizontally, in each noise level, the Fetch-Ratio of our feature description network is generally greater than DIP and 3dmatch, which means that our network can make better use of the feature information contained in the point cloud. DIP is only focused on the local geometric information, which makes its descriptors more distinctive on different models. Consequently, DIP outperforms our method in model bunny and dragon. Vertically, both algorithms are inevitably affected by noise, but our method is still higher than DIP and 3dmatch at each noise level. DIP is particularly affected by noise, which means although the features extracted by DIP are rotation invariant, they are not robust and general when being applied to models with strong noise. On the whole, the descriptors we generated are more robust and discriminative than the descriptor of DIP and 3dmatch, and will also bring some improvement in the subsequent registration effect.

4.4 Registration effect

In this section, we tested the registration effects of ISS + FPFH, 3DMatch, DIP [49], Predator [48] and our algorithm under the conditions of noise-free, 0.1 ρ, 0.5 ρ noise levels and 10% outliers. The dataset is shown in Fig. 11 and their average sampling density is 0.001mm. We recorded the average angle of the normal vector and translation in the common part of the data as the initial conditions. The angle and translation in the common part of armadillo is 111°and 0.079mm, respectively.

In all experiments, the number of match points used by 3dmatch/DIP/Predator is 500(which is the default value), and the number of key points selected by our algorithm is 128. We report all the registration rate in Table 4.



(a). noise-free (b). 0.1 ρ noise (c). 0.5 ρ noise (d). 10% outliers

Fig. 11 Original data

Table 4 Coarse registration rate

Method	Noise level			
	0	0.1 ρ	0.5 ρ	10% Outlier
ISS + FPFH	69.77	66.15	56.7	21.8
3DMatch	68.20	67.07	70.45	60.56
DIP	89.67	83.54	79.21	68.14
Predator	85.88	79.88	75.43	75.98
Our	**97.08**	**95.84**	**96.84**	**82.85**

Bold indicates the highest Coarse registration rate for all methods at each noise level, indicating that our method is better than other methods

Figure 12 shows the registration effects of the three algorithms for the noise-free point cloud. It can be found that although ISS filters out the points with significant features in the point cloud, the position of the extracted feature points were changed greatly due to the resampling and noise, which makes registration results deviate greatly at the head and legs. In Fig. 12b, with the better descriptor generated by 3DMatch, the registration effect of the head and legs of the model is improved. The overlap-attention block in Predator can exchange early information between the latent encodings of the two point clouds which greatly improves registration performance. Both DIP and Predator perform better than the first two methods, except for some details like ears and hands. Our algorithm adopts curvature extreme points with better repeatability, and uses the channel attention mechanism to strengthen the discrimination of descriptors further. Therefore, the registration effect is the best of the five, and the model fitted well at the head, legs and hands.

In Fig. 13, we added Gaussian noise with amplitude of 0.1 ρ to the point cloud to verify the robustness of our algorithm. The results show that the ISS + FPFH has poor robustness, mainly because the LCS constructed by FPFH is affected by the noise, resulting in deviation of the registration. 3DMatch performs better than ISS + FPFH, but there is still a large deviation at the head of the model. DIP

(a). ISS+FPFH (b). 3DMatch (c). DIP (d). Predator (e). Ours

Fig. 12 Registration result for noise-free point cloud

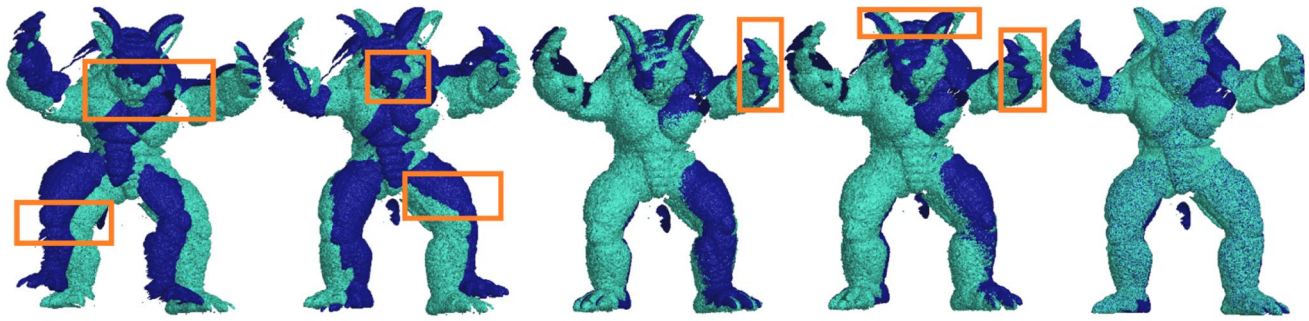


Fig. 13 Registration result of 0.1 ρ noise level

and Predator pay few attentions to noisy data when training, so they are affected and the registration rate decreases about 6%. Our feature network is trained with noisy data, so the algorithm has the best robustness in the case of noise, and the registration rate still reached 95.84%.

In the experiment in Fig. 14, we increased the noise amplitude to 0.5 ρ. The registration effect of ISS + FPFH is further deteriorated, and the hand and head of the model were deviation greatly. The registration effect of 3DMatch was also worse. As previously analyzed, DIP is vulnerable to noise and Predator is committed to improving registration effect in low-overlap scenario, which leads to a further deterioration of the effect. In contrast, our algorithm still performed well, the registration results in the parts with significant characteristics such as the head, hands and legs of the model are comparable to the case of noise-free, which means our algorithm still has high robustness in the case of strong noise.

In the experiment of Fig. 15, we added 10% outliers to both models. It can be seen from the results that the ISS algorithm has poor robustness to outliers, resulting in registration failure and the point cloud model is directly flipped. 3DMatch and DIP random sample the point cloud to generate matching points, so it has a considerable chance to include outliers to the matching set, resulting in a poor registration effect. The overlap-attention block of Predator can predict salient points lying in the overlap region, which makes it achieve an unexpected good result on models with outliers. In our algorithm, when selecting curvature extreme points, a distance threshold is used to filter outliers, so the probability that an outlier point is selected is rare. At the same time, we also use a distance threshold constraint on the virtual corresponding points, even if outliers are selected, they will be filtered at this stage, which makes the final registration accuracy not be reduced too much.

As shown in Table 4, we take the coarse registration rate as our metric to evaluate the effect of our algorithm on the armadillo model. Specifically, we calculated the proportion of the number of points whose nearest neighbor distance meets the threshold (0.1 ρ) after coarse registration and fine registration, and calculated the ratio of the two. Horizontally, our algorithm performs well in all the noise cases. The registration rate reaches 97.08% for the noise-free situation, and only decreases by 1.24% and 0.24% in the case of 0.1 ρ and 0.5 ρ noise, respectively. Although the registration rate decreases by 14.23% when 10% outliers are added, it

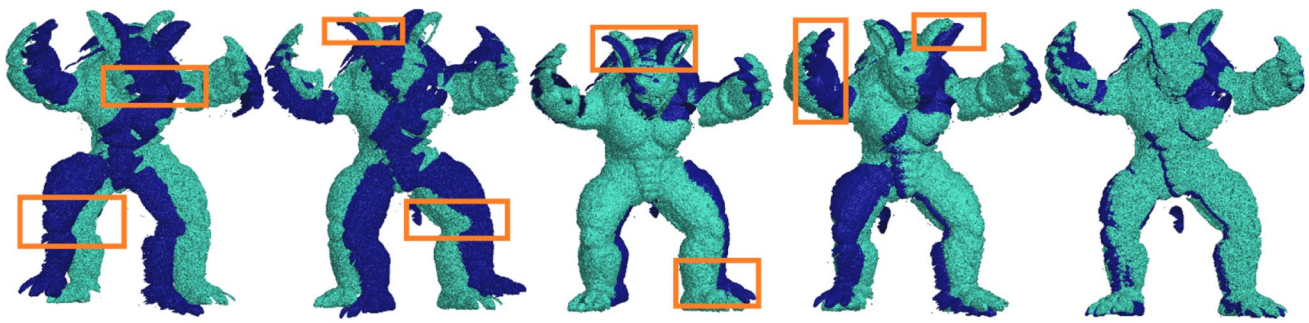


Fig. 14 Registration result of 0.5 ρ noise level

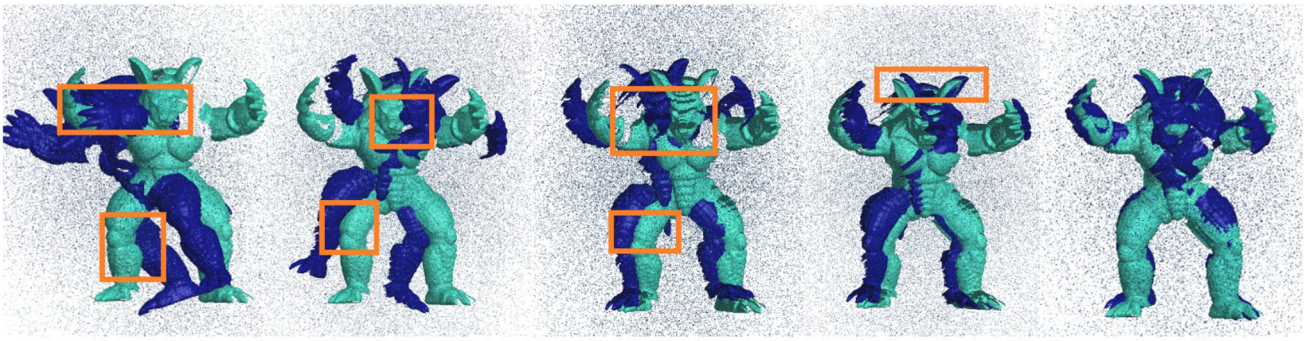


Fig. 15 Registration result of 10% outliers

is still much higher than other methods. In other words, the registration accuracy of our algorithm will not differ much with noise and outliers. However, 3DMatch performs much worse than our algorithm. The highest registration accuracy is only 70.45%, and it drops to 60.56% rapidly when outliers are added. The registration accuracy of ISS + FPFH under 0.5 ρ noise and 10% outliers drop by 23.07% and 47.97%, respectively, which is much more obvious, and the registration accuracy is only 46.7% and 21.8%. DIP and Predator seem to have a greater decline than 3DMatch when noise is added, but their overall performance is still better than first two methods. Vertically, our algorithm also performs better than other four methods in all cases, which shows that the registration accuracy of our algorithm is greatly improved compared with both deep learning algorithms and traditional algorithm.

4.5 Algorithm universality

Aim to verify the generality of our algorithm, we select five more models including the Dancing Children, the Horse, the Bunny, the Buddha and the Dragon. The original data is shown in Fig. 16 and their average sampling density is 0.001mm. For initial conditions, the average angle and translation in the common part of five models is 7°and 0.1 mm, respectively. Among them, the Dancing Children model and the Horse model are produced manually by obtaining visible point clouds from different perspectives, while the other three models are the actual scanning data from the Stanford 3D scanning repository. We also add 0.5 ρ noise to all those models to increase the challenge.

As shown in Figs. 17, 18, 19, 20, 21, ISS + FPFH and 3DMatch perform poorly in several models. For example, in Fig. 17, they deviations a lot at the head and base of the second child, and there are also large deviations in the early faucet in Fig. 21. DIP and Predator achieve an excellent result in Figs.17, 18, 19. However, they perform not so well in models with rich surface feature information like Figs. 20 and 21. The registration effect of our algorithm has significant improvements over ISS + FPFH and 3DMatch. When facing models with smooth surface, we can do as well as

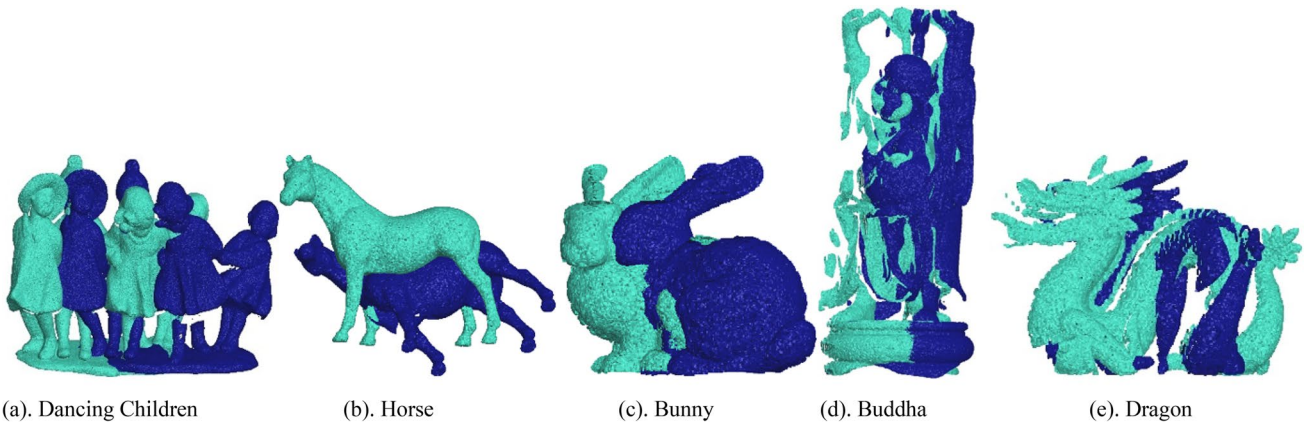


Fig. 16 Original data

Fig. 17 Registration result of the Dancing Children model

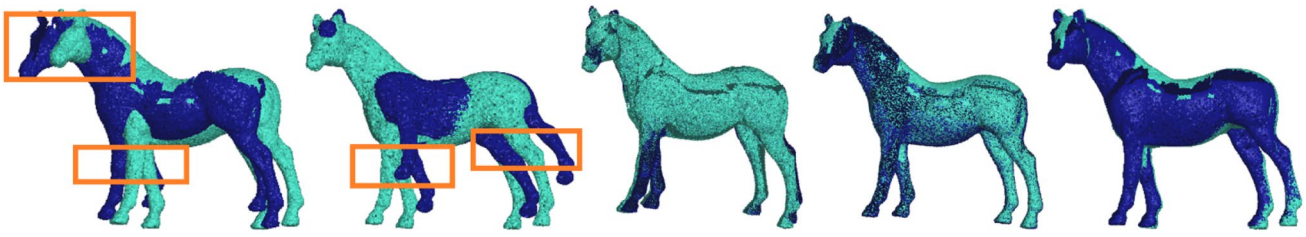


Fig. 18 Registration result of the Horse model

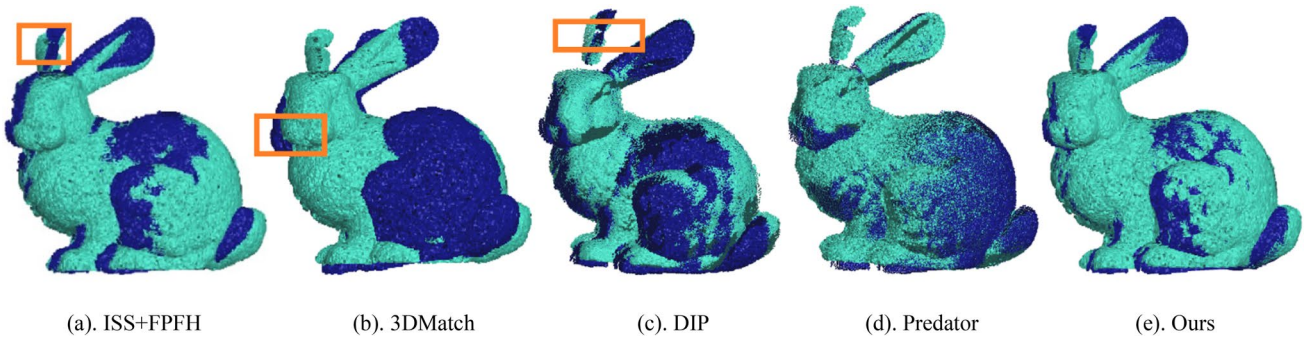


Fig. 19 Registration result of the Bunny model

DIP and Predator. And, our algorithm can still maintain a high level in Figs. 20 and 21 where other methods achieve unsatisfactory results.

We report all the coarse registration rate on five models in this section. We can see from Table 5 that although our algorithm cannot get best results on every model, the maximum gap between us and the best method is less than 2%. Moreover, our average registration rate has reached 90%, which means the generalization ability of our algorithm.

4.6 Registration on more dataset

Recently, more and more papers [38, 41–45, 49] focus on indoor scenes, ModelNet40 [58] and KITTI outdoor LiDAR dataset [59]. In order to verify the university of our algorithm further, we select several models from 3DMatch [20], NDT [57], ModelNet40 and KITTI to compare with two state-of-the-art methods. Specifically, Fig. 22 is from 3DMatch and its angle and translation in the common part is 13 and 0.15mm, respectively; Fig. 23 is from NDT and its angle and translation in the common part is 5°and 0.037 mm, respectively; Figs. 24 and 25 is from KITTI; Figs. 26, 27, 28 is from ModelNet40.

Indoor scene point cloud data is usually characterized by structural occlusion and self-similarity. These factors have brought some difficulties to the registration of indoor scene registration. ModelNet40 is usually used to evaluate the performance of 3D shape classification algorithms.

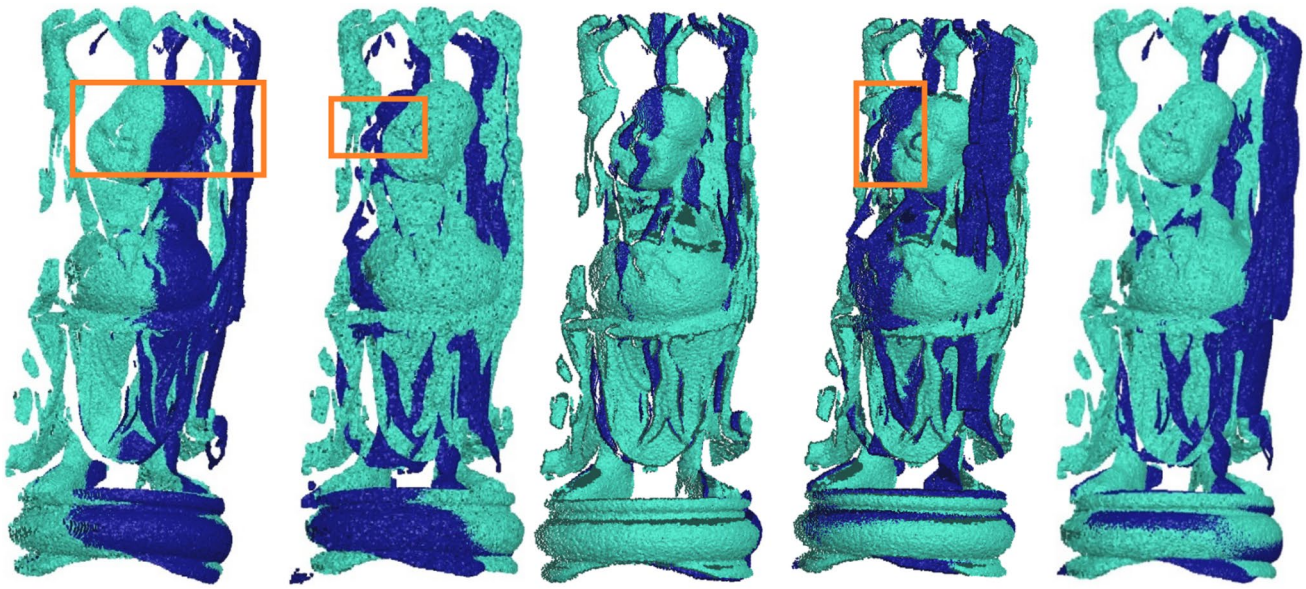


Fig. 20 Registration result of the Buddha model

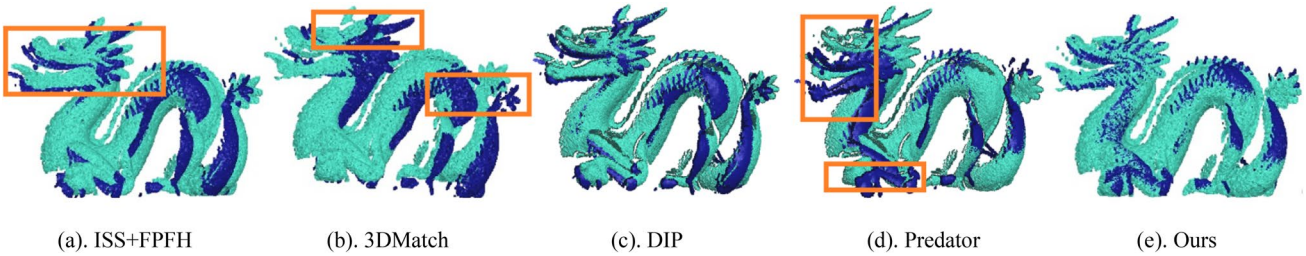


Fig. 21 Registration result of the Dragon model

Table 5 Coarse registration rate on dataset of generality test

Method	Model				
	Dragon	Children	Horse	Bunny	Buddha
ISS + FPFH	86.25	78.54	76.91	86.15	79.24
3DMatch	81.3	81.88	75.7	84.29	82.13
DIP	89.74	88.54	88.69	89.16	**90.44**
Predator	77.62	**91.23**	**90.05**	92.35	80.17
Our	90.13	90.81	89.46	**94.58**	88.54

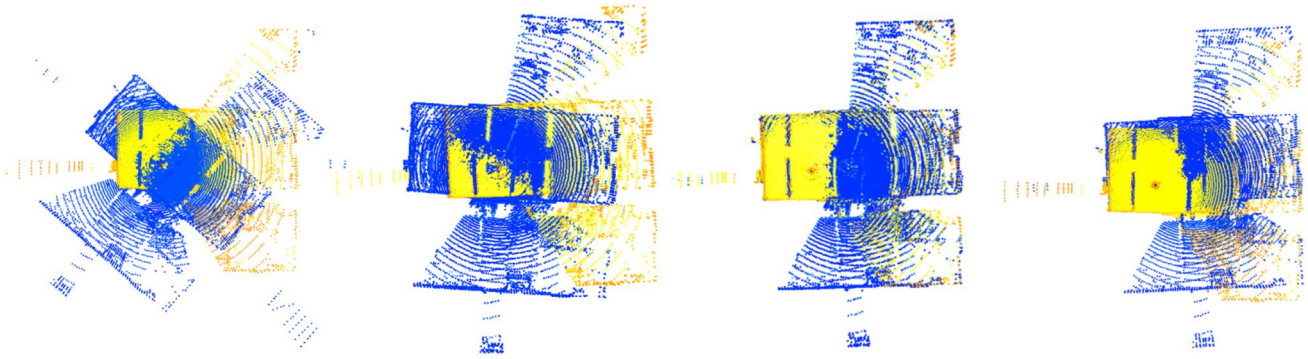
KITTI includes multi-view images taken by autonomous vehicles and some lidar data. It is a great challenge to perform well on these datasets. As shown in Figs. 22, 23, 24, 25, 26, 27, 28, with highly differentiated descriptor, all DIP, Predator and our algorithm achieve an excellent result on all datasets. According to statistics, the average registration rate of three methods is higher than 85%, which also shows that we are able to complete the task of registration on these challenging datasets.

4.7 Registration with two matching points

[48] points out that although the low-overlap regime is very relevant for practical applications, the registration performance of some state-of-the-art methods deteriorates rapidly when the overlap between the two point clouds is very low (< 30%).When the overlapping area of two point clouds is too small, it will bring great challenges to alignment, because it will lead to that only few feature points can be searched in the overlapping area. Also, the feature points in non-overlapping areas will bring more interference to feature point matching in that situation. Figures 29 and 30 show such scenarios. In the overlapping region of

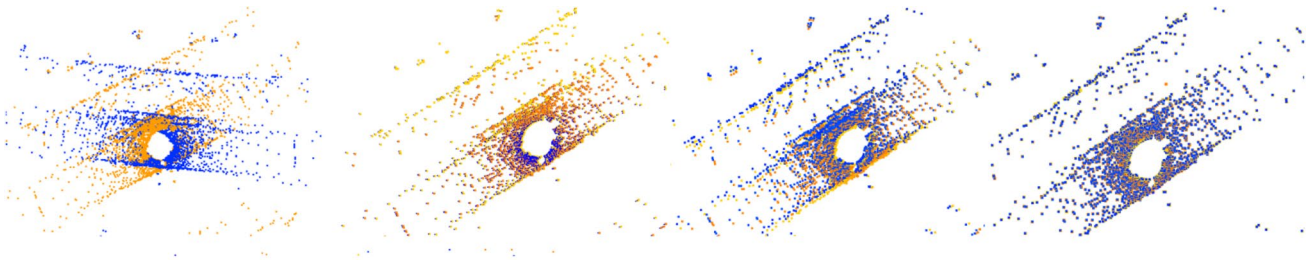
(a). Original data (b). DIP (c). Predator (d). Ours

Fig. 22 Registration result of the indoor scene model (1)



(a). Original data (b). DIP (c). Predator (d). Ours

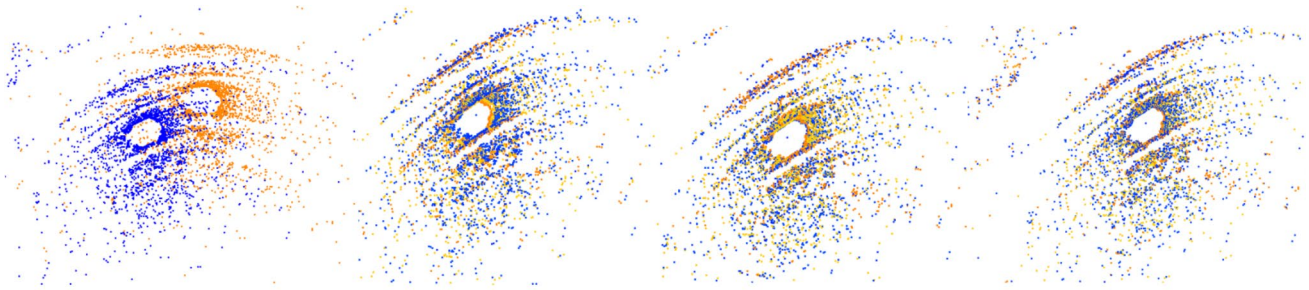
Fig. 23 Registration result of the indoor scene model (2)



(a). Original data (b). DIP (c). Predator (d). Ours

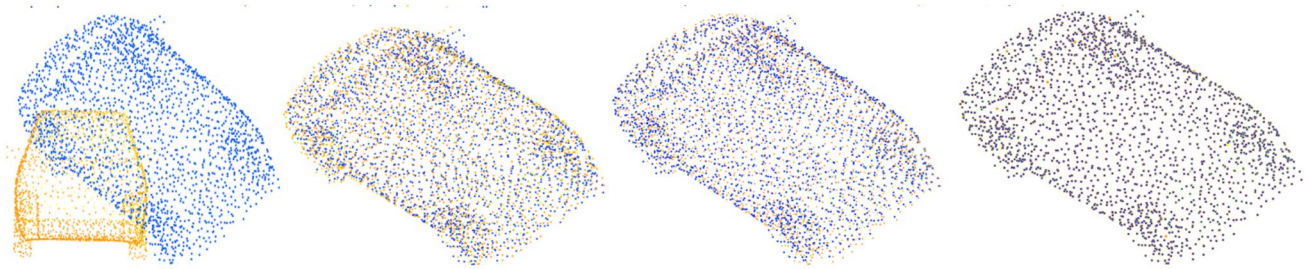
Fig. 24 Registration result of KITTI (1)

the two models in the figure, only three matching points located in the overlapping region can be found in armadillo and two in dragon. Therefore, the traditional methods cannot stably calculate the registration parameters. However, the overlap-attention block can greatly improve performance in the low-overlap scenario. We introduced the concept of virtual feature points, so more virtual matching points can be constructed by offsetting the matched points along its normal vector. Finally, enough corresponding points can still be got to calculate the registration parameters. It can be seen from the figure that our method outperforms Predator because the overlapping region is too small for Predator to extract good enough features for registration. Thanks to our proposed virtual matching points, two models coincide well after registration and our registration rate reaches 86%, which is higher comparison than Predator's 79%.



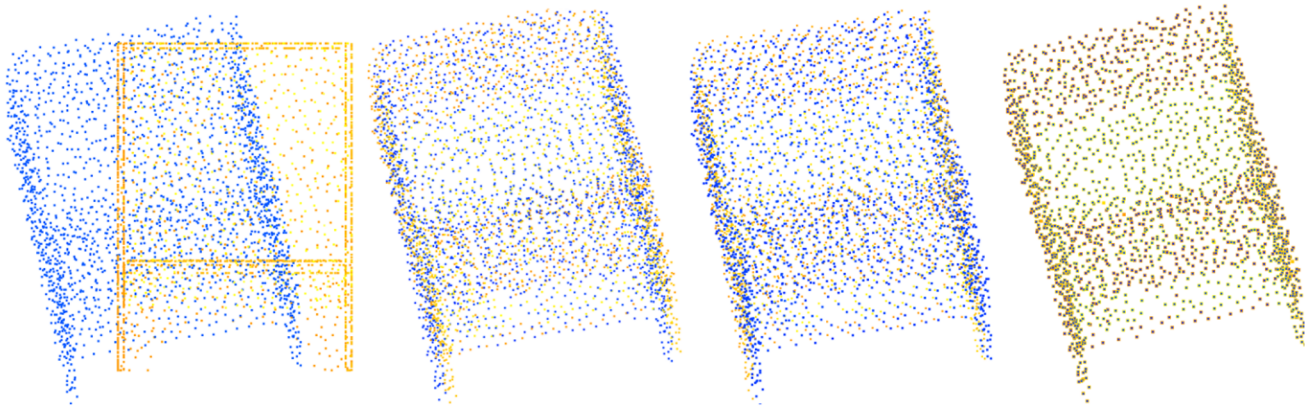
(a). Original data (b). DIP (c). Predator (d). Ours

Fig. 25 Registration result of KITTI (2)



(a). Original data (b). DIP (c). Predator (d). Ours

Fig. 26 Registration result of ModelNet40 (1)

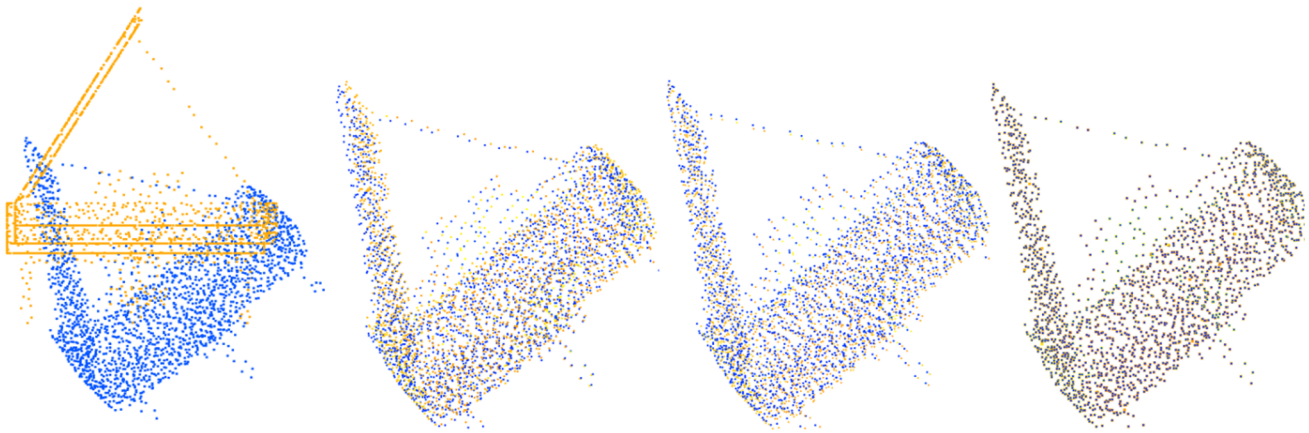


(a). Original data (b). DIP (c). Predator (d). Ours

Fig. 27 Registration result of ModelNet40 (2)

5 Conclusion

In this paper, a coarse point cloud registration algorithm based on local extremum feature and depth descriptor matching is proposed. The algorithm identifies the curvature extreme points as feature points through a lightweight network, which improves the repeatability and robustness of feature detection; meanwhile, it generated highly differentiated descriptors through a lightweight attention-based

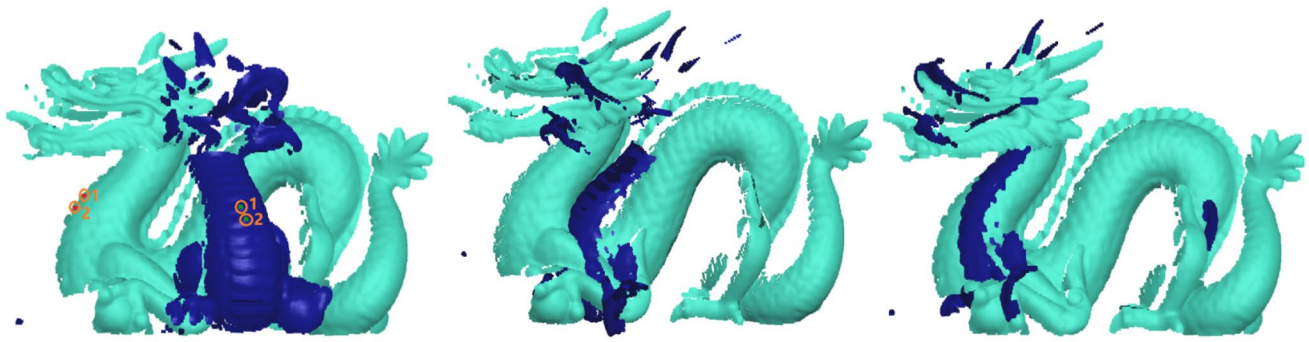


(a). Original data (b). DIP (c). Predator (d). Ours

Fig. 28 Registration result of ModelNet40 (3)

(a). Original data (b). Predator (c). Ours

Fig. 29 Registration with virtual feature points (armadillo)



(a). Original data (b). Predator (c). Ours

Fig. 30 Registration with virtual feature points (dragon)

feature description network. We also propose a method to generate virtual corresponding points based on feature points and their normal vectors, which reduces the minimum number of feature points required for registration from 4 to 2. Experiments show that the proposed feature extraction, description and registration methods have more advantages than traditional methods, and have achieved good registration results in various challenging scenes.

In the future, we will do more research from the following aspects: first of all, when testing Fetch-Ratio, we found that although the performance of our network is still better than 3dmatch and DIP, the drop of Fetch-Ratio of our network is relatively large when the noise level reaches a very high level. How to further improve the robustness of the network in high noise environment is one of our research directions. Secondly, we find that the distance along the normal vector needs to be turned manually when generating virtual matching points. In future studies, we will further improve the algorithm.

Acknowledgements We would like to thank all the reviewers for their valuable comments. We also thank the AIM@Shape Repository and the Stanford Repository for providing the models used in this paper. This work is partly supported by the NSFC (No. 61802204).

Author contributions Haotian Lu: Data curation, Writing-Original draft preparation, Software. Jianhui Nie: Conceptualization, Methodology, Writing-Original draft preparation

Data availability The datasets generated or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Jean-Emmanuel Deschaud.: Imls-slam: scan-to-model matching based on 3d data. In Proceedings of the International Conference on Robotics and Automation. pp. 2480–2485. (2018)
2. Jay, M.W., Vincent, K., Tiffany, L., Syler, W., Gian, L.M., Abraham, S., Lei, H., Rahul, C., Mitchell, H., David, M.S.J., Jimmy, W., Bolei, Z., Antonio, T.: Segicp.: Integrated deep semantic segmentation and pose estimation. In Proceedings of the International Conference on Intelligent Robots and Systems.pp. 5784–5789 (2017)
3. Thomas Probst, Danda Pani Paudel, Ajad Chhatkuli, Luc Van Gool.: Unsupervised learning of consensus maximization for 3d vision problems. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 929–938 (2019)
4. Besl, P.J., Mckay, N.D.: A method for registration of 3-D shapes. Proc SPIE Int Soc Opt Eng **14**(3), 239–256 (1992)
5. Yang, J., Li, H., Campbell, D., et al.: Go-ICP: a globally optimal solution to 3D ICP Point-Set registration. IEEE Trans. Pattern Anal. Mach. Intell.Intell. **38**(11), 2241–2254 (2016)
6. Zhou, Q.Y., Park, J., Koltun, V.: Fast global registration. European Conference on Computer Vision, pp. 766–782. Springer International Publishing, Amsterdam (2016)
7. Zhou, Q.-Y., Park, J., Koltun, V.: Fast global registration. In: Proceedings of the European conference on computer vision, pp. 766–82. Springer, New York (2016)
8. Shen, C., Wu, Y., Cai, G.: Multiple views Lidar point cloud registration for buildings based on Quaternion constraint. J Jimei Univ. **24**(5), 393–400 (2019)
9. Birdal, T., Ilic, SA.: (2017) point sampling algorithm for 3d matching of irregular geometries. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Doi: 10.1 109/IROS.2017.8206609
10. Harris, CG., Stephens, M.: A combined corner and edge detector. Proceedings of Fourth Alvey Vision Conference. pp. 147–151 (1988)
11. Zhong, Y.: Intrinsic shape signatures: a shape descriptor for 3d object recognition. Proceedings of 2009 IEEE 12th International Conference on Computer Vision Workshops. pp. 689–696 (2009)
12. Steder, B., Rusu, RB., Konolige, K., et al.: Point feature extraction on 3D range scans taking into account object boundaries. IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9–13 May 2011. IEEE. (2011)
13. You Y., et al.: Keypointnet: A large-scale 3d keypoint dataset aggregated from numerous human annotations. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2020)
14. You, Y., et al.: UKPGAN: A General Self-Supervised Keypoint Detector. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2022)
15. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Trans. Pattern Anal. Mach. Intell.Intell. **21**(5), 433–449 (1999)
16. Zai, D., Li, J., Guo, Y., Cheng, M., Huang, P., Cao, X., Wang, C.: Pairwise registration of tls point clouds using covariance descriptors and a non-cooperative game. ISPRS J Photogram Remote Sens. **134**, 15–29 (2017)
17. Guo, Y., Sohel, F., Bennamoun, M., Lu, M., Wan, J.: Rotational projection statistics for 3d local surface description and object recognition. IntJ Comput Vision. **105**(1), 63–86 (2013)
18. Tombari, F, Salti, S. and Di Stefano, L.: Unique signatures of histograms for local surface description. In: European conference on computer vision, Springer, pp. 356–369. (2010)
19. Li L, Zhu S, Fu H, et al.: End-to-end learning local multi-view descriptors for 3D point clouds. IEEE (2020)
20. Zeng A,Song S, M. Nießner, et al.: 3DMatch: learning local geometric descriptors from RGB-D reconstructions. IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society. pp.199–208 (2017)
21. Aiger, D., Mitra, N.J., Cohen-or, D.: 4-points congruent sets for robust pairwise surface registration. Acm Trans Graphics **27**(3), 1–10 (2008)
22. Mellado, N., Aiger, D., Mitra, N.J.: Super4PCS: fast global pointcloud registration via smart indexing. Comput Graphics Forum **33**(5), 205–215 (2015)
23. Mohamad, M., Ahmed, MT., Rappaport, D., et al.: Super generalized 4PCS for 3D registration. International Conference on 3D Vision (3DV). IEEE Computer Society. (2015)
24. Huang, J., Kwok, TH., Zhou, C.: V4PCS: volumetric 4PCS algorithm for global registration. ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. (2017)
25. Xu, Z., Xu, E., Zhang, Z., et al.: Multiscale sparse features embedded 4-points congruent sets for global registration of TLS point clouds. IEEE Geosci. Remote Sens. Lett.Geosci. Remote Sens. Lett. **16**(2), 286–290 (2018)

26. Hussnain, Z., Elberink, S.O., Vosselman, G.: Automatic feature detection, description and matching from mobile laser scanning data and aerial imagery. Int Arch Photogramm Remote Sens Sci. **XLI-B1**, 609–616 (2016)
27. Li, R., Man Yang, Yu., Tian, Y.L., Zhang, H.: Point cloud registration algorithm based on the ISS feature points combined with improved ICP algorithm. Laser Optoelectron Progr **54**(11), 111503 (2017)
28. Rusu, RB., Blodow, N., Marton, ZC., et al.: Aligning point cloud views using persistent feature histograms . In IEEE/RS J International Conference on Intelligent Robots and Systems. (2008)
29. Rusu, RB., Bradski, GR., Thibaux, R., et al.: Fast 3D recognition and pose using the Viewpoint Feature Histogram. 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, October 18–22, 2010, Taipei, Taiwan. IEEE. (2010)
30. Van Blokland, B.I., Theoharis, T.: Radial intersection count image: a clutter resistant 3D shape descriptor. Comput. Graph.. Graph. **91**(1), 18–28 (2020)
31. Darom, T., Keller, Y.: Scale-invariant features for 3D mesh models. IEEE Trans. Image Process. **21**(5), 2758–2769 (2012)
32. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int J Comput Vis. **60**(2), 91–110 (2004)
33. Chems-Eddine, H., et al.: PCEDNet : A Neural Network for Fast and Efficient Edge Detection in 3D Point Clouds. (2020).
34. Wu, N.Z., Song, S., Khosla, A., et al.: 3D ShapeNets: a deep representation for volumetric shapes. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2015)
35. Le. T., Ye, D.: PointGrid: A deep network for 3D shape understanding. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE. (2018)
36. Elbaz, G., Avraham, T., Fischer, A.: 3D Point Cloud Registration for Localization Using a Deep Neural Network Auto-Encoder[C]// Computer Vision & Pattern Recognition. IEEE Computer Society. (2017)
37. Qi, CR., Su, H., Mo, K., et al.: PointNet: deep learning on point sets for 3D classification and segmentation.2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE. (2017)
38. Wang, Y., Justin M.S.: Deep closest point: learning representations for point cloud registration. Proceedings of the IEEE/CVF international conference on computer vision. (2019)
39. Choy, C., Dong, W., Koltun, V.: Deep global registration. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 2511–2520. (2020)
40. Vaswani, A., et al.: Attention is all you need. In: Proc. Adv. Neural Inf. Process. Syst. pp. 5998–6008. (2017)
41. Yew, ZJ, Gim HL.: Rpm-net: Robust point matching using learned features. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. (2020)
42. Lu, W., et al.: Deepvcp: An end-to-end deep neural network for point cloud registration. Proceedings of the IEEE/CVF International Conference on Computer Vision. (2019)
43. Wang, Y., Solomon, J.: PRNet: Self-supervised learning for partial-to-partial registration. Mach Learn. **32**(23318422), 8814–8826 (2019)
44. Zhang, Z., et al.: End-to-end learning the partial permutation matrix for robust 3D point cloud registration. Proc AAAI Conf Artif Intell. **36**(3), 3399–3407 (2022)
45. Zhang, Z., et al. VRNet: learning the rectified virtual corresponding points for 3D point cloud registration. IEEE Transactions on Circuits and Systems for Video Technology. (2022)
46. Ao S., et al.: Spinnet: Learning a general surface descriptor for 3d point cloud registration. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2021)
47. Bai, X., et al.: Pointdsc: Robust point cloud registration using deep spatial consistency. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2021)
48. Huang, S., et al.: Predator: Registration of 3d point clouds with low overlap. Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. (2021)
49. Poiesi, F., Davide B.: Distinctive 3D local deep descriptors. 2020 25th International conference on pattern recognition (ICPR). IEEE. (2021)
50. Itti, L, Koch, C, Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. In: IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI). (1998)
51. Rensink, R.A.: The dynamic representation of scenes. In: visual cognition 7.1–3. (2000)
52. Corbetta, M, Shulman, G.L: Control of goal-directed and stimulus-driven attention in the brain. In: Nature reviews neuroscience 3.3. (2002)
53. Wang, F., et al.: Residual attention network for image classification. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017). https://doi.org/10.1109/cvpr.2017.683
54. Hu, J., et al.: Squeeze-and-excitation networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
55. Woo, S., Park, J., Lee, J.Y., et al.: CBAM: convolutional block attention module. Springer, Cham (2018)
56. Fleishman, S., Cohen-Or, D., Silva, C., et al.: Robust moving least-squares fitting with sharp features. ACM Trans Graphics **24**(3), 544–552 (2005)
57. Biber, P., Strasser, W.: The normal distributions transform: A new approach to laser scan matching. Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2003)
58. Zhirong, W., et al.: 3D ShapeNets: A deep representation for volumetric shapes. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, pp. 1912–1920 (2015). https://doi.org/10.1109/CVPR.2015.7298801
59. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, pp. 3354–3361 (2012). https://doi.org/10.1109/CVPR.2012.6248074