Multi-scale Point Octree Encoding Network for Point Cloud based Place Recognition

Zhilong Tang[1], Hanjing Ye[1] and Hong Zhang[1*]

***Abstract*— Over the past decades, point cloud-based place recognition has garnered significant attention. This research paper presents a pioneering approach, denoted as the Multi-scale Point Octree Encoding Network (MPOE-Net), designed to acquire a discriminative global descriptor for efficient retrieval of places. The key element of the MPOE-Net is the point octree encoding module, which adeptly captures local information for each point by considering its nearest and farthest neighbors. Further enhancing local relationships, a multi-transformer network is introduced, utilizing a novel grouped offset-attention mechanism. To amalgamate the multi-scale attention maps into a comprehensive global descriptor, a multi-NetVLAD layer is incorporated. Through rigorous experimentation across diverse benchmark datasets, our proposed method unequivocally outperforms existing techniques in the realm of point cloud-based place recognition tasks, achieving state-of-the-art results. Our code is released publicly at https://github.com/Zhilong-Tang/MPOE-Net.**

I. INTRODUCTION

Place recognition constitutes a fundamental cornerstone in the domains of robot navigation, autonomous driving, and augmented reality. This process involves matching a given query scene, either in the form of an image or a point cloud, against a comprehensive database to determine the closest correspondences and subsequently establish the query's localization within the reference map. Notably, place recognition bifurcates into two primary categories: point cloud-based place recognition, as demonstrated by PointNetVLAD approach [1], and image-based place recognition [2]. Although numerous image-based place recognition networks have been recently proposed, they are susceptible to challenges associated with varying illumination, seasonal changes, and the confined field of view of the camera. As a viable alternative, LiDAR technology has garnered attention, as it circumvents these limitations and capitalizes on its capacity to provide precise 3D data, effectively enhancing the accuracy of place recognition processes.

Numerous studies have emerged in recent decades concerning point cloud place recognition. However, generating a robust and distinctive global descriptor for a single query remains a primary challenge. Inspired by PointNet [3] and NetVLAD [4], Mikaela *et al.* [1] proposed PointNetVLAD, a pioneering work for learning-based point cloud place recognition. PointNetVLAD employs PointNet to extract local features from 3D point cloud data and then uses NetVLAD as the global descriptor generator. PCAN was proposed by Zhang *et al.* [5] in 2019. PCAN extracts local point features by PointNet and produces an attention map that estimates a weight for each point based on the contextual information. Inspired by PointNet++, PCAN uses ball query search based on different query radii to extract multi-scale features. Nonetheless, Both methods use PointNet as a point local features generator, which does not consider the local geometric structure of the point cloud. Liu *et al.* [6] proposed a large-scale place description network (LPD-Net). Instead of only using the original point cloud coordinate, they extract local features as the network input. They propose a graph-based aggregation module in both feature space and Cartesian space to further reveal the spatial distribution of local features and inductively learn the structure information of the whole point cloud. Self-attention and orientation encoding network (SOE-Net) [7] was proposed by Xia *et al.* in 2021. They use a novel point cloud orientation encoding module named point orientation encoder (PointOE) to extract the point cloud orientation feature. PointOE considers the spatial relationship between eight different orientations for a single point. Le *et al.* [8] proposed a pyramid point transformer network (PPT-Net), which uses a pyramid transformer structure to extract local features. Instead of using query ball, which PointNet++ proposed, they employ k-neighbor-nearest (KNN) as the graph embedding encoder. This method uses KNN, which does not consider the orientation information.

Previous research has underscored the significance of local neighbor information in effective place recognition. Drawing inspiration from PointSIFT [9], we propose a point octree encoding module. Unlike prior approaches that solely consider the nearest neighbor, our proposed module takes both the nearest and the farthest neighbors into account, effectively merging features from both. Additionally, a multi-transformer module is presented to calculate the attention map of the point cloud, enhancing the discriminative capabilities of the network. To enhance computational efficiency while maintaining accuracy, a grouped offset-attention module is devised. This module optimizes the computation process, resulting in improved overall accuracy. Lastly, the attention maps obtained are integrated into the multi-NetVLAD layer, enabling the generation of a discriminative global descriptor, crucial for robust place recognition.

The contributions of this paper are as follows:

- We propose a new point cloud local descriptor extraction method that considers both the nearest and farthest points within a particular radius in an octree. This

*Corresponding author (hzhang@sustech.edu.cn)

Zhilong Tang, Hanjing Ye and Hong Zhang are with Shenzhen Key Laboratory of Robotics and Computer Vision, Southern University of Science and Technology (SUSTech), and the Department of Electrical and Electronic Engineering, SUSTech, Shenzhen 518055, China

This work was supported by the Shenzhen Key Laboratory of Robotics and Computer Vision (ZDSYS20220330160557001).

method makes use of more local information of each point to generate a more significant local descriptor.
- We propose a multi-transformer layer that enhances the local spatial relationship using a novel grouped offset-attention module.

II. Related Work

Point cloud based place recognition is converted to a feature matching problem. The 3D descriptor has a significant impact on performance. Various point cloud descriptor extractors are proposed, which can be divided into two categories: 3d local descriptor and 3d global descriptor.

A. 3D Local Descriptor

The goal of a 3D local descriptor is to provide a discriminative and robust representation of a local neighborhood in a 3D point cloud, allowing for efficient comparison and matching between different neighborhoods. There have been several approaches to 3D local descriptors in recent years, including hand-crafted and deep learning-based descriptors. Spin image [10] uses the idea of projecting the local surface geometry of a point cloud onto a 2D plane and describing it using a histogram. Geometry histogram [11] represents the local geometry of a point cloud as a set of the histogram which is based on regional shape context. Point feature histogram (PFH) [12] describes the local geometry information of a point cloud by capturing the shape and geometry information. Fast point feature histogram (FPFH) [13] is a fast and efficient variant of PFH, which uses a more straightforward and efficient method to compute the histograms. Nonetheless, these methods based on the histogram need to be more robust for large-scale place recognition due to sensitivity to noisy and incomplete data acquired by sensors.

Recently, some learning-based 3d local descriptor extraction methods have been proposed. Inspired by convolutional neural networks (CNNs), volumetric-based point cloud deep learning feature extraction methods are proposed like 3D SharpNets [14], volumetric CNN [15], OctNet[16] for 3D object classification. 3DMatch [17], which jointly learns geometric feature presentation and associated metric functions from real-world data. In addition, to represent the point cloud as a voxel, multi-view projection is also an excellent method to extract the point cloud feature. Multi-view convolutional neural network (MVCNN) [18] projects point clouds into different views and uses CNNs to extract features from every view. PointNet [3] is a pioneering work for processing point cloud point-wise. It takes the original point cloud data as input and extracts features using CNNs. Based on PointNet, point pair feature network (PPF-Net) is proposed by using CNN to extract and match point pair features that describe relative orientation and distance between two points in a point cloud. The author of PPF-Net also proposes PPF-FoldNet [19], which adds a folding operation to capture the global context in the point clouds.

B. 3D Global Descriptor

The mainstream methods of 3d point cloud place recognition generate a global discriminative scene descriptor. Various global descriptor-generating algorithms have been proposed, which can be divided into two categories: hand-crafted and learning-based. [20] propose M2DP, a hand-crafted method which generates the global descriptor by projecting the 3d point cloud into different 2d planes and generating a density signature for each plane. DELIGHT [21] leverages intensity information to generate a novel descriptor of LiDAR intensities. The descriptor encodes the distributed histograms of the intensity of the surroundings, which are compared using chi-squared tests. [22] propose a robust place recognition algorithm that adopts Bearing Angle (BA) to convert the 3D point cloud to images. Oriented fast and rotated brief (ORB) features are extracted from the images for scene matching. PointNetVLAD [1] is the first learning-based 3D global descriptor-generating network for point cloud place recognition. It utilizes PointNet to extract local features from point cloud data and employs NetVLAD to generate the global descriptor for the place recognition task. Compared to PointNetVLAD, PCAN [5] adds a point contextual attention network to generate multi-scale contextual information by ball query search. Large-scale place recognition descriptor (LPD-Net) [6] utilizes original point cloud data and local features extracted from the original point cloud as the input. They propose a Graph-based Neighborhood Aggregation to learn the spatial distributed relationship of the scene. SOE-Net [7] extracts features by PointSIFT [9], which encodes point cloud through 8 orientation information. PPT-Net [8] proposes a pyramid point transformer module to learn the local relationship adaptively. A pyramid VLAD layer is designed to aggregate multi-scale feature maps into a global descriptor.

III. Method

Fig. 1 shows the architecture of our neural network MPOE-Net. Original point cloud will be used as the only input, and four point octree encoding (PointOE) modules and four offset-attention modules are designed to extract local descriptors, which will be delivered to a multi NetVLAD layer to generate a discriminative global descriptor.

Given a query point cloud as the input with coordinates donated as $Q = p_1, \cdots, p_N \in \mathbb{R}^{N\times 3}$, a PointOE module is designed to extract local descriptors for each point. We also design a multi-transformer layer to enhance the local neighbor relationship. A multi-NetVLAD network is proposed to take the feature maps generated from every transformer module to produce a discriminative global descriptor by utilizing multi-scale information.

A. Local Descriptor Extraction

Previous work [9] [7] about point orientation encoding utilize the nearest neighbors of each point in eight direction, which can not accurately perceive the distribution of neighborhood information. We propose a new point octree encoding (PointOE) to utilize the nearest neighborhood and

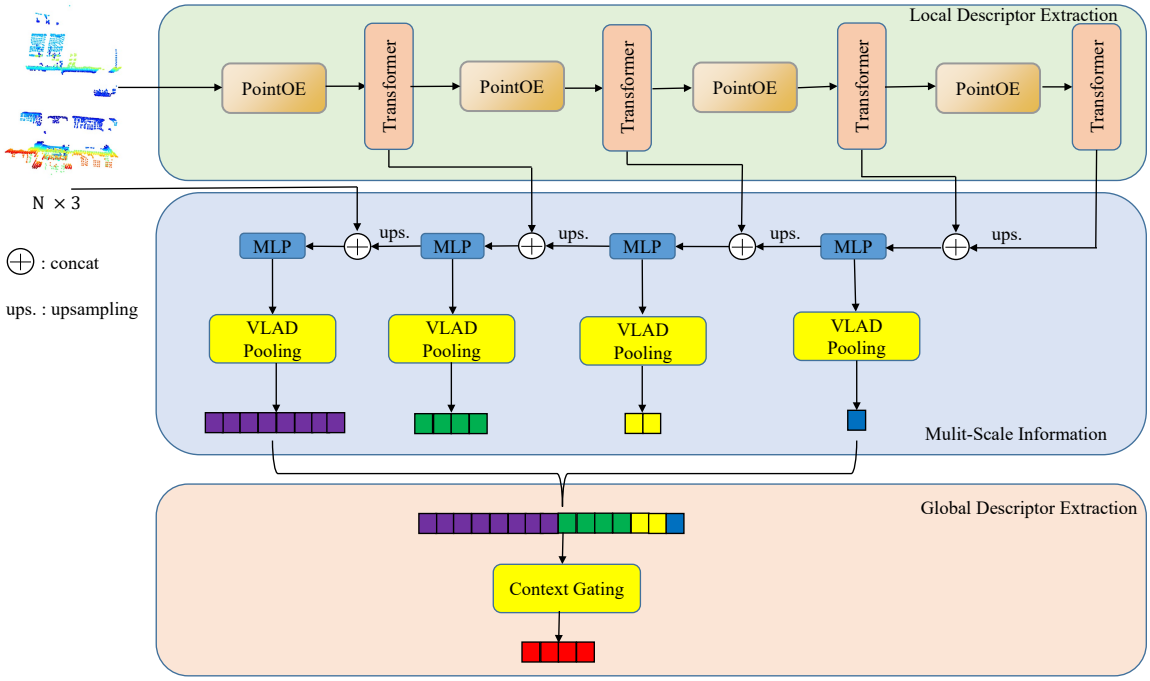


Fig. 1. Overview of our MPOE-Net architecture. Our network takes point cloud as input data. Four PointOE are employed to extractor local descriptors. Four feature maps are feed into multi-NetVLAD layer to generate a discriminative global descriptor

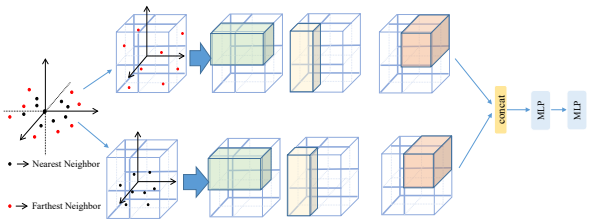


Fig. 2. Illustration of our PointOE module

farthest neighbors of each point to extract the local descriptor. The nearest and farthest neighbors are defined as the nearest and farthest distance of two points in Euclidean space. The space can be divided into eight blocks, and it means eight different directions for each point.

We also propose a multi-transformer module to enhance the local spatial relationship. More specifically, we have shown in Fig. 1 the multi-transformer module that we have incorporated. At the end of PointOE, we added a transformer module to enhance the spatial contextual relationships.

Consider a $N \times C$ matrix as the input of our PointOE module, where N is the number of points, and C is the feature dimension of each point. First, we employ a farthest point sampling (FPS) to downsampling the input matrix and a new feature matrix $(N^{'} \times C)$. Our module then adopt two stacked 8-neighborhood search (S8N): one for the nearest neighbors $(N^{'} \times 8 \times C)$ and another for the farthest points $(N^{'} \times 8 \times C)$ of the new feature matrix within a particular radius. As we show in fig. 2, black points mean the nearest neighbor of each directions, and the red points mean the farthest points within a particular radius. A three-stage convolution is used to extract features from the neighbors and the farthest points where the convolutional kernel size is $(1, 2)$ and the stride is $(1, 2)$:

$$V_x = g(Conv(W_x, V))$$

$$V_{xy} = g(Conv(W_y, V_x))$$

$$V_{xyz} = g(Conv(W_z, V_{xy}))$$

After three-stage convolution, we obtain two feature maps $(N^{'} \times C \times 1)$, and we concatenate these two new feature maps into one new feature map $(N^{'} \times C \times 2)$. A new convolution is applied to this new feature map and generates a new feature map $(N^{'} \times C \times 1)$. Finally, we put this new feature map into our grouped offset-attention network to learn the local relationship between different regions. More details will be introduced in the next section.

B. Grouped Offset Attention

The self-attention mechanism has demonstrated excellent performance in the fields of speech and image processing. However, its drawback lies in the high computational cost required for its implementation. [8] proposes a grouped self-attention module to reduce the computation of the attention module. In addition, inspired by graph convolutional networks [23], [24] designs an offset-attention network to replace the original self-attention network. The offset-attention network is designed to adaptively learn the local spatial relationship between different point cloud regions. We propose a novel grouped offset-attention network to generate feature maps. Feature maps generated from the attention

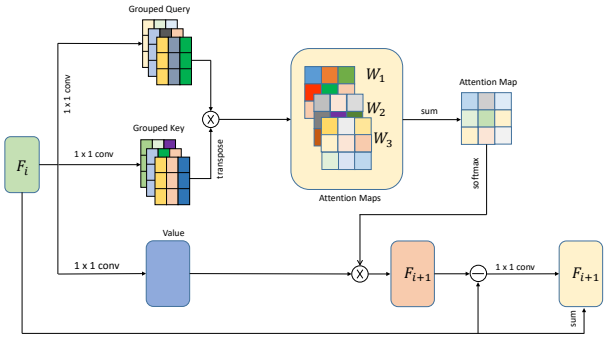


Fig. 3. Illustration of grouped offset-attention module

network will be delivered to the NetVLAD layer to produce a global descriptor. Fig. 2 shows the architecture of our offset-attention network.

Local features $F_i \in \mathbb{R}^{m\times C}$ are extracted from i-th stage PointOE module. Two group-wise 1×1 convolution networks are applied to local features F_i to generate the query map $Q \in \mathbb{R}^{m\times C}$ and the key map $K \in \mathbb{R}^{m\times C}$. Another 1×1 convolution network is used to generate value map $V \in \mathbb{R}^{m\times C}$. The query map and key map can be divided into G groups along the channel side, donated by $\{Q_g \in \mathbb{R}^{m\times C/G} | g = 1, \cdots, G\}$, $\{K_g \in \mathbb{R}^{m\times C/G} | g = 1, \cdots, G\}$. In the g-th group, attention map $W_g \in \mathbb{R}^{m\times m}$ can be calculated as:

$$W_g = Q_g \cdot K_g^T$$

The final attention map $W \in \mathbb{R}^{m\times m}$ can be calculated by adding all G attention maps as follow:

$$W = \sum W_g$$

Multiplying the value map V with the attention map W followed by a $softmax$ function, and can get the original output of attention layer $F_{i+1} \in \mathbb{R}^{m\times C}$ which can be formulated by:

$$F_{i+1} = softmax(\frac{W}{\sqrt{C}}) \cdot V$$

where C is the dimension of query map. Graph neural networks [23] show the advantage of using a Laplacian matrix $L = D - E$ to replace the adjacency matrix E, where D is a diagonal degree matrix. Inspired by it, We use an offset-attention module to replace the self-attention module where the final output can be calculated as:

$$F_{i+1} = Conv(F_{i+1} - F_i) + F_i$$

The final attention map will be sent to next stage encoder to extract local descriptors.

C. Multi NetVLAD

Previous work [1] [5] [7] for point cloud place recognition uses only a single NetVLAD module to generate a global descriptor. These work does not make use of multi-scale information. In our network, four grouped offset-attention modules are designed to generate different scale feature maps, which are crucial to producing a discriminative global descriptor. We propose a multi-NetVLAD to utilize multi-scale feature maps to generate the global descriptor. The multi NetVLAD layer is shown in Fig. 1.

Our multi-NetVLAD layer utilizes four multi-scale feature maps as the input. The feature map, which is generated from the different receptive fields, has a different ability to represent the local information of a single point. The low-level features with smaller receptive fields may only work partially, and the high-level features are used to combine with low-level features to enhance the representation ability. Inspired by [25], we use the point feature propagation module to upsample the feature. The input feature maps can be formulated as: F_1, F_2, F_3 and F_4, and the new feature maps can be formulated as:

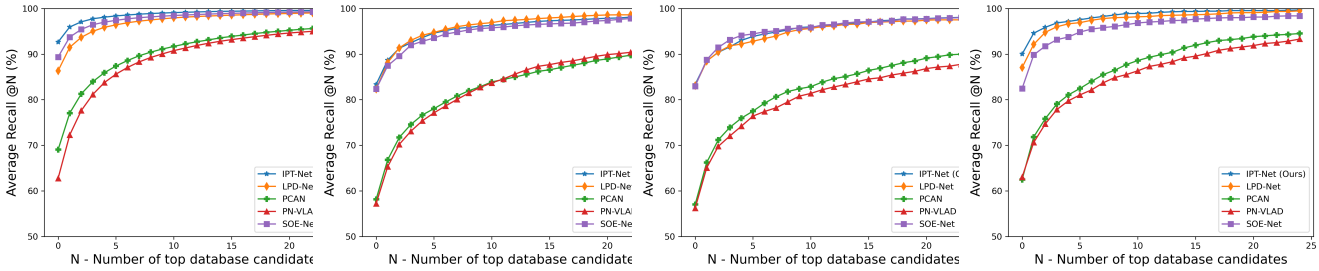
$$F_i^{'} = MLP(F_i \oplus \mathcal{L}(F_{i+1}^{'}))$$

where $i \in \{0, 1, 2, 3\}$ and $F_0^{'}, F_1^{'}, F_2^{'}$ and $F_3^{'}$ are the new feature maps. $\oplus$ means channel-wise concatenation and $\mathcal{L}$ means the point feature propagation. We generate the multi-scale global descriptors by NetVLAD [4] based on the new feature maps. This layer can learn k_i visual words for each feature map which can be denoted as $\{d_i^j \in \mathbb{R}^C \mid j = 1, \cdots, k_i\}$, and creates a $(C \times K_i)$ dimensional vector $u_i = [u_i^1, \cdots, u_i^{k_i}]$. After this, we generate four global descriptors at four resolutions. To obtain a more discriminative global descriptor, we utilize the context gating mechanism. Finally, the discriminative descriptor can be generated by the context gating module.

IV. Experiments

A. Benchmark Datasets

We utilize the benchmark datasets proposed by [1] to train and evaluate our network. The benchmark datasets include four different scenes: the Oxford RobotCar outdoor dataset [26] and three in-door datasets of the university sector (U.S.), residential area (R.A.), and business district (B.D.). These datasets are created by a LiDAR sensor which is mounted on a car that travels through four regions repeatedly at different times, traversing a 10km, 10km, 8km, and 5km route. The collected LiDAR data is used for a reference map which can be used to construct a database of submaps. GPS/INS readings are used to build the reference map with respect to the UTM coordinate frame. The ground planes are removed in submaps processing. The final point cloud is downsampled to 4096 points. It is also rescaled and shifted to zero means and inside the range of $[-1, 1]$. To generate a training tuple, if the distance between two point clouds is less than 10m, they will be regarded as positive and, if the distance is larger than 50m, they will be regarded as negative. In the test case, distance less than 25m will be regarded as a positive match. We will provide a baseline version, and a refined version result. We first train the baseline model only using the Oxford RobotCar dataset. Then, we train the refined model by adding the U.S. and R.A. datasets to improve the generalizability of our network. Tab. II shows the details of baseline datasets and



(a) Oxford (b) Business (c) Residential (d) University

Fig. 4. Evaluation results (AR@ top N) of different methods trained on the **baseline dataset**

TABLE I

THE AVERAGE RECALL AT TOP 1 AND TOP 1% IN FOUR DATASETS FOR DIFFERENT METHODS FOR **BASELINE MODEL**

	Oxford		U.S.		R.A.		B.D.	
	AR@1%	AR@1	AR@1%	AR@1	AR@1%	AR@1	AR@1%	AR@1
PN_VLAD	81.0	62.8	77.8	63.0	69.8	56.2	65.3	57.2
PCAN	83.8	69.1	79.1	62.5	71.2	57.0	66.8	58.1
LPD-Net	94.9	86.3	96.0	87.0	**90.5**	83.1	**89.1**	82.3
SOE-Net	96.4	89.4	93.2	82.5	91.5	82.9	88.5	83.3
MPOE-Net (OURS)	**97.7**	**92.7**	**96.8**	**90.0**	90.4	**83.3**	88.7	**83.3**

TABLE II

THE SPLIT OF TRAINING AND TEST DATASETS

	Training		Test	
	Baseline	Refine	Baseline	Refine
Oxford	21711	21711	3030	3030
Indoor	-	8442	4542	1766

refined datasets. We have 21711 Oxford submaps for baseline training. 21711 Oxford submaps and 8442 indoor submaps are prepared for refine training. To evaluate our baseline model, we choose 3030 Oxford submaps and 4542 indoor submaps. 3030 Oxford submaps and 1766 indoor submaps are prepared for refine model test.

B. Implementation Details

We implement our network by the PyTorch framework, and our network is trained on a single Nvidia RTX 3090 GPU with 24G memory. The number of points in an input point cloud is 4096. We use a four-stage PointOE module to extract the local descriptors. The number of points after downsampling is 2048, 1024, 512, and 256. The radii of the PointOE module are 0.1, 0.1, 0.25, and 0.5. The number of groups in grouped offset-attention is 8. The output dimension of multi-NetVLAD layer is 256. The initial learning rate is 0.0001, and it is halved every 5 epochs. We use Adam as the optimizer and train the model for 30 epochs. The batch size is 1, and we choose 2 positive clouds and 14 negative point clouds to calculate the loss. We choose the hardest positive and hardest negative quadruplet loss [7] as the loss function.

C. Baseline Network

We compare our baseline network with some excellent methods, including PointNetVLAD [1], PCAN [5], LPD-Net [6], SOE-Net [7]. These methods are all learning-based methods. We use the same training datasets and test datasets. We denote PointNetVLAD as PN_VLAD. We use average recall at top N and average recall at top 1% as the evaluation metrics.

Tab. I shows the average recall@1% (AR@1%) and average recall@1 (AR@1) of different methods trained on the baseline datasets and tested on different datasets. As we can see, our MPOE-Net achieves the best result in most test datasets. Our MPOE-Net achieves the best performance at both top 1% and top 1 on Oxford, and U.S. datasets. We also obtain good results at R.A. and B.D., where our performance is exceeded only by LPD-Net, which relies on ten handcrafted features. Fig. 4 shows the recall curves of the top 25 retrieval results of PN_VLAD, PCAN, LPD-Net, SOE-Net, MPOE-Net at four test datasets. The results show that our methods improve the performance at Oxford and U.S. datasets, and our method also has good results in B.D. and R.A. datasets. These results indicate that our network can extract significant local features and generate discriminative global descriptors.

D. Refine Network

In addition to train on the baseline dataset, we also train our network on the refine dataset. We report the AR top 1 of different methods on four datasets. As we can see in table III, our method achieves better performance than other methods. According to the performance of the baseline network and refine network, we know our network can work well for point cloud place recognition tasks. Compared to other methods, our approach also exhibits superior generalization capability.

TABLE III

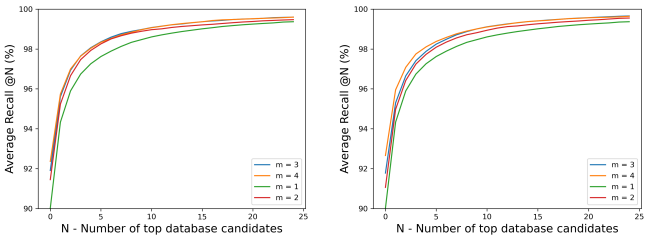
EVALUATION RESULTS OF DIFFERENT METHODS TRAINED ON **REFINE DATASET.**

	Oxford	U.S.	R.A.	B.D.
PN_VLAD	63.3	86.1	82.7	80.2
PCAN	70.9	84.3	82.9	80.2
SOE-Net	89.3	91.8	90.2	89.0
MPOE-Net (OURS)	**93.2**	**97.6**	**95.4**	**90.9**

TABLE IV

THE EAVLUATION RESULT OF DIFFERENT MODULE

	AR@1%	AR@1
N.P.	96.4	91.1
N.S.P	97.4	91.7
F.P	95.5	87.3
MPOE	**97.7**	**92.7**



(a) Multi-Transformer (b) Multi-NetVLAD

Fig. 5. The evaluation result of multi-transformer and multi-NetVLAD module

V. DISCUSSION

A. Ablation Study

Ablation studies verify the availability of our proposed modules, including the PointOE module, multi-transformer module, grouped offset-attention module, and multi-VLAD module. We evaluate our different modules on the Oxford RobotCar dataset.

PointOE. In our PointOE module, we use the nearest and farthest points to replace the nearest point. These experiments are designed to validate the effectiveness of our PointOE. MPOE in the table. IV means our proposed network with the PointOE module. We use other local features to compare with our method, such as the nearest point (N.P.) only, nearest and secondary nearest points (N.S.P.), and the farthest point (F.P.) only. We use AR@1% and AR@1 as the evaluation metrics.

Compared with N.P., our PointOE module sees an improvement of 1.3% at AR@1% and 1.6% at AR@1. It means our PointOE module proved to be an effective module in improving performance. It is due to richer context information of each point. Results in tab. IV shows our nearest and farthest neighbors strategy works better than N.S.P. and F.P. Farthest point does not work well independently.

Grouped offset-attention. To evaluate the effectiveness of our grouped offset-attention module (MPOE), we design these experiments. We change our grouped offset-attention module with self-attention module (S.A.), offset-attention module (O.A.), and grouped self-attention module (G.S.A). We use AR@1% and AR@1 as the evaluation metrics.

TABLE V

THE EAVLUATION RESULT OF DIFFERENT MODULE

	AR@1%	AR@1
S.A	97.2	89.9
O.A	97.5	91.8
G.S.A	97.4	91.6
MPOE	**97.7**	92.7

TABLE VI

THE EAVLUATION RESULT OF DIFFERENT OUTPUT DIMENSIONS

	AR@1%	AR@1
D = 128	97.2	91.0
D = 256	97.7	92.7
D = 512	**97.8**	**92.9**

Compared to other attention methods, our grouped offset-attention achieves the best performance. It seems that all methods perform well at AR@1, but our method can achieve better performance at AR@1.

Multi-transformer and Multi-NetVLAD. We reduce the number of transformer modules and NetVLAD modules to validate the effectiveness of our multi-transformer and multi-NetVLAD module. We conduct the experiments on the Oxford RobotCar dataset. We choose AR@25 to be the evaluation metrics.

Fig. 5 shows the evaluation results. With the increment of the number of transformer modules, the AR@1 increases. This is due to the local spatial relationship being enhanced by our transformer module. The more the transformer modules, the better the result. Fig. 5(b) shows AR@25 with the change of the number of NetVLAD modules. When we use four NetVLAD layers, we obtain the best result. It is because we use the richest multi-scale feature maps to generate the discriminative global descriptor.

B. Output Dimension Analysis

In this section, we evaluate the performance of the global descriptor with different dimensions. We use AR@1% and AR@1 to evaluate our method. All the experiments are conducted on the Oxford RobotCar dataset. We try the dimensions at 128, 256, and 512.

Tab. V shows the result. When the output dimension decrease from 256 to 128, AR@1 declines by 1.7%. When we increase the output dimension from 256 to 512, AR@1 only increases by 0.2%, and AR@1% only increases by 0.1%. This validates the robustness of our method against different output dimensions.

VI. CONCLUSIONS

We propose a novel multi-transformer, multi-NetVLAD network with a point octree encoding module in this paper. Compared to the original PointSIFT module, we utilize more local information around a point to extract more significant

local descriptors. A multi-transformer module is designed to enhance the local relationship between different regions using a grouped offset-attention module. The grouped offset-attention module can simplfy the network and improve the performance of our network. Finally, we design a multi-NetVLAD layer to generate a discriminative global descriptor by using multi-scale feature maps. Extensive experiments on the test datasets show that our network can achieve the state-of-the-art in the point cloud based place recognition task.

REFERENCES

[1] M. A. Uy and G. H. Lee, "Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4470–4479.

[2] H. Ye, W. Chen, J. Yu, L. He, Y. Guan, and H. Zhang, "Condition-invariant and compact visual place description by convolutional autoencoder," *Robotica*, vol. 41, no. 6, p. 1718–1732, 2023.

[3] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[4] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[5] W. Zhang and C. Xiao, "Pcan: 3d attention map learning using contextual information for point cloud based retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 436–12 445.

[6] Z. Liu, S. Zhou, C. Suo, P. Yin, W. Chen, H. Wang, H. Li, and Y.-H. Liu, "Lpd-net: 3d point cloud learning for large-scale place recognition and environment analysis," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2831–2840.

[7] Y. Xia, Y. Xu, S. Li, R. Wang, J. Du, D. Cremers, and U. Stilla, "Soe-net: A self-attention and orientation encoding network for point cloud based place recognition," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2021, pp. 11 348–11 357.

[8] L. Hui, H. Yang, M. Cheng, J. Xie, and J. Yang, "Pyramid point cloud transformer for large-scale place recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6098–6107.

[9] M. Jiang, Y. Wu, T. Zhao, Z. Zhao, and C. Lu, "Pointsift: A sift-like network module for 3d point cloud semantic segmentation," *arXiv preprint arXiv:1807.00652*, 2018.

[10] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 21, no. 5, pp. 433–449, 1999.

[11] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part III 8*. Springer, 2004, pp. 224–237.

[12] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *2008 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2008, pp. 3384–3391.

[13] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *2009 IEEE international conference on robotics and automation*. IEEE, 2009, pp. 3212–3217.

[14] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1912–1920.

[15] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view cnns for objecwt classification on 3d data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5648–5656.

[16] G. Riegler, A. Osman Ulusoy, and A. Geiger, "Octnet: Learning deep 3d representations at high resolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3577–3586.

[17] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3dmatch: Learning the matching of local 3d geometry in range scans," in *CVPR*, vol. 1, no. 2, 2017, p. 4.

[18] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 945–953.

[19] H. Deng, T. Birdal, and S. Ilic, "Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 602–618.

[20] L. He, X. Wang, and H. Zhang, "M2dp: A novel 3d point cloud descriptor and its application in loop closure detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 231–237.

[21] K. P. Cop, P. V. Borges, and R. Dubé, "Delight: An efficient descriptor for global localisation using lidar intensities," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3653–3660.

[22] F. Cao, Y. Zhuang, H. Zhang, and W. Wang, "Robust place recognition and loop closing in laser-based slam for ugvs in urban environments," *IEEE Sensors Journal*, vol. 18, no. 10, pp. 4242–4252, 2018.

[23] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," *arXiv preprint arXiv:1312.6203*, 2013.

[24] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "Pct: Point cloud transformer," *Computational Visual Media*, vol. 7, pp. 187–199, 2021.

[25] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[26] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.