https://doi.org/10.5755/j02.eie.40523

BSPR-Net: Dual-Branch Feature Extraction Network for LiDAR Place Recognition in Unstructured Environments

Zhichao Zhang[1], Youchun Xu[1,*], Zhenhuan Yuan[1], Yulin Ma[2]
[1]Institute of Military Transportation, Army Military Transportation University,
Tianjin 300161, China
[2]School of Mechanical and Automotive Engineering, Anhui Polytechnic University,
Beijing Mid Rd 1, Wuhu 241000, China
zzcjake6688@163.com; [*]xu56419@126.com; Yuan785663056@163.com; mayulin@mail.ahpu.edu.cn

***Abstract*—LiDAR point cloud-based place recognition (LPR) in unstructured natural environments remains an open challenge with limited existing research. To address the limitations of unstructured environments, such as sparse structural features, uneven point cloud density, and significant viewpoint variations, we present BSPR-Net, a dual-branch point cloud feature extraction approach for point cloud place recognition, which consists of a BEV - projection rotation - invariant convolution branch and a point cloud sparse convolution branch. This design enhances the representation capability of geometric structural features while aggregating rotation-invariant characteristics of point clouds, thereby better addressing the challenge of large viewpoint disparities in reverse-revisited unstructured environments. The proposed network was tested on multiple reverse-revisited sequences of the Wild-Places data set, a benchmark for unstructured natural environment place recognition. It achieved a maximum F1 score of 85.46 %, exceeding other classical methods by more than 4 %. The ablation experiments further confirmed the effectiveness of each module in improving place recognition performance.**

***Index Terms*—Unstructured environments; LiDAR point clouds; Place recognition; Dual-branch features.**

I. INTRODUCTION

Regarding the problem of robot localisation, Elhousni and Huang [1] provide a very clear explanation of satellite-based positioning and point out that localisation problems can be divided into two categories. One type is pose-tracking-based localisation, where the robot's position is estimated and tracked over time using sequential information in the state space, such as inertial navigation and odometry-based localisation. The other type is global position retrieval-based localisation within a given area, such as re-localisation and place recognition. Place recognition is an active research field in robotics and is a key technology for solving long-term autonomous navigation in large-scale environments [2], [3]. This is an autonomous robot localisation method based on environmental perception. It works well in areas with weak or heavily interfered satellite signals, such as underground spaces and indoor areas. Place recognition mainly has two methods: vision-based methods and point cloud-based methods. In general, place recognition involves comparing the similarity between the current frame's information (such as images or 3D point clouds) obtained by the robot and the information in a map database, thus determining the robot's current location [4]. This method is crucial to solve localisation problems in areas where satellite signals are weak or severely interfered with, such as underground spaces and buildings.

Vision-based place recognition methods have been widely researched and have yielded rich results. However, under the task requirements of large-scale environments and long-term autonomous navigation, robots face complex environmental factors, such as changes in lighting, seasons, and viewpoints, which make vision-based methods less adaptable to these environmental changes. In contrast, point cloud-based methods demonstrate relatively higher robustness under these conditions. In recent years, continuous advances in LiDAR technology have led to higher hardware integration, lower costs, denser laser point clouds, and better 3D performance, which has made LiDAR-based systems more widely used in various types of robots. Research on LiDAR-based place recognition algorithms (as shown in Fig. 1) has also become more abundant [5], [6].

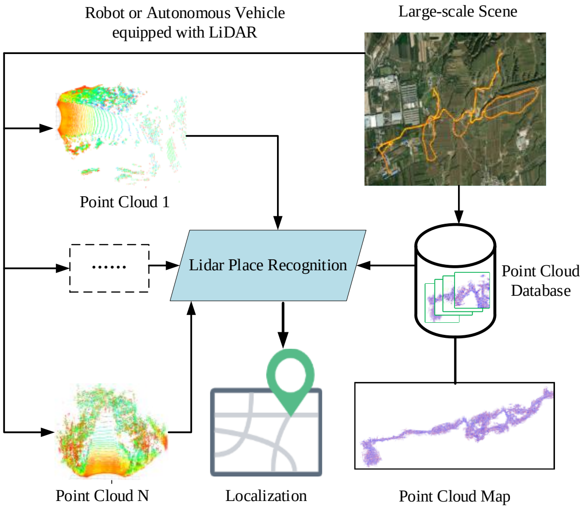


Fig. 1. LiDAR-based place recognition workflow.

Manuscript received 12 December, 2024; accepted 11 March, 2025.

However, most of these research works focus on urban environment data sets such as KITTI [7], MulRan [8], and NCLT [9], where the structured features of the data sets are relatively rich. Additionally, the target point clouds and source point clouds are often of the same origin, with similar point cloud densities at the same locations, and the data used to validate location revisiting performance mostly involve same-direction revisits with small viewpoint differences. Although many algorithms have achieved state-of-the-art (SOTA) performance on these data sets, their performance may drop significantly under conditions of high noise, large viewpoint differences, and nonhomogeneous point clouds. For example, autonomous vehicles operating in real-world scenarios face more challenges such as reverse revisits, environmental changes, and unstructured natural conditions, which make location recognition using point clouds more difficult.

Guan *et al.* [10] first established the 3D cross-source data set CS-CAMPUS3D and proposed CROSSLoc3D. CROSSLoc3D focusses on the feature representation differences between aerial and ground laser scans due to noise, viewpoint, and density pattern variations. It employs multiscale feature extraction and iterative refinement in scale space to reduce the feature representation gap between point clouds from different sources, achieving good results. Inspired by this, this paper uses U-Net based on sparse convolution [11] for feature extraction. Although it does not directly emphasise multiscale feature extraction in its design, it performs exceptionally well in capturing details at different levels. Autonomous vehicles operate on the ground and primarily move in a 2D plane. The bird's-eye view (BEV) projection of the point cloud can provide important location features and makes it easier to design rotation-invariant descriptors in the 2D plane to counter large viewpoint differences caused by reverse revisits. Therefore, this paper proposes a novel dual-branch feature extraction method for point cloud location recognition based on BEV projection and sparse convolution. To better aggregate the dual-branch features, the paper also uses a cross-attention mechanism for feature fusion.

The main contributions of this paper are threefold:

1. The design of BSPR-Net (as shown in Fig. 2), a dual-branch feature extraction point cloud location recognition network architecture based on BEV projection and sparse convolution, which effectively aggregates BEV rotation-invariant features and 3D geometric structural features in the global feature descriptor of the point cloud.
2. An improvement of the point branch of the sparse convolution U-Net to enhance its ability to capture local structural features of point clouds.
3. Validation on the KITTI data set and the Wild-Places data set for unstructured natural environments, demonstrating the effectiveness of the proposed BSPR-Net.

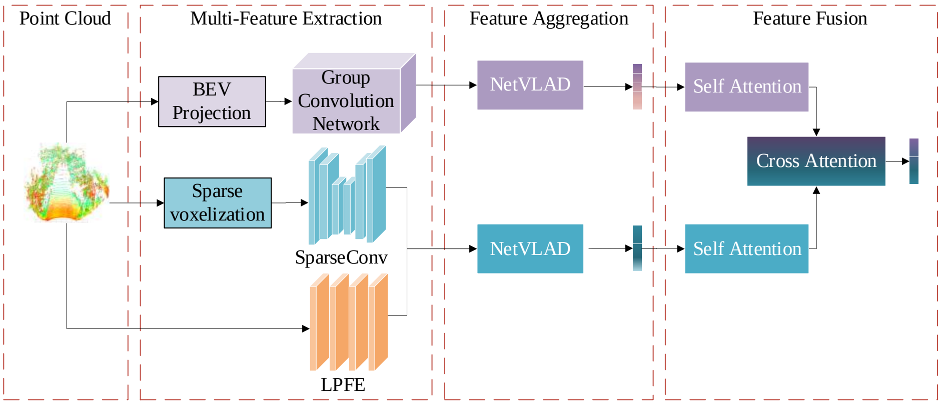


Fig. 2. BSPR-Net backbone.

II. RELATED WORKS

A. BEV Projection-based LPR

Projecting 3D point clouds onto the 2D plane of Bird's Eye View (BEV) and then extracting global descriptors has proven to be an efficient approach to place recognition. BEV-based methods are particularly robust to motion from LiDAR sensors. BVMatch [12], proposed in 2021, demonstrated that BEV images are more robust to LiDAR sensor movement because the rotation of the point cloud only results in rotational changes in the BEV image, avoiding significant appearance changes. OverlapNet [13] introduced a deep neural network that leverages overlapping information in the distance projection image and the estimation of the relative yaw angle, without requiring relative pose information. RING++ [14] proposed a point cloud feature descriptor with rotation-translation invariant representation, focussing on global localisation under large viewpoint variations and lightweight maps. Overlap Transformer [15] extended OverlapNet by incorporating the Transformer network architecture, designing a lightweight place recognition network. The authors in [16] introduced a cross-view transformer network (CVTNet) that combines distance images and BEV dual-branch point cloud projection features, enabling the online generation of orientation-invariant descriptors, demonstrating strong robustness and real-time performance in large-scale environments. AttDLNet [17] also primarily focusses on rotation-invariant descriptors. BEVPlace [18], after projecting point clouds onto BEV, uses convolutional networks to extract rotationally equivariant local features from the images, followed by global feature aggregation using NetVLAD. The aforementioned BEV projection methods mainly focus on the extraction of rotation-invariant descriptors.

B. Sparse Convolution-based LPR

In large-scale point cloud feature extraction, 3D sparse convolution reduces the computational load of point cloud processing while preserving the 3D structural features of the point cloud. It represents a trade-off between methods based on raw points and projections. MinkLoc3Dv2 [19] uses efficient 3D convolutional feature extraction and channel attention blocks, along with a differentiable loss function that approximates mean average precision. MinkLoc3D-SI [20] combines the advantages of handcrafted descriptors with the most effective 3D sparse convolution, enhancing the performance of a single 3D LiDAR scan. TransLoc3D [21] proposed a large-scale point cloud location recognition method that uses adaptive receptive fields to address challenges such as object size variation, irrelevant moving objects, and long-range contextual information. LoGG3D-Net [22] introduced a local consistency loss to guide the network in learning consistent local features during loop closure, forming more reliable global descriptors and improving 3D location recognition performance. CROSSLOC3D [10] addresses the problem of cross-source 3D location recognition between aerial and ground data by embedding features from different sources into a unified standard space through multigranularity features and iterative refinement processes, facilitating metric learning. Although sparse convolution is suitable for processing large-scale point cloud data, it may struggle to fully capture local features or maintain sufficient descriptiveness when dealing with extremely sparse or unevenly distributed point cloud data. Additionally, its design for rotational invariance is relatively complex.

III. MODELS AND METHODS

A. *Problem Description*

The problem model described in this paper can be outlined as follows. Firstly, a point cloud map database with location labels is pre-built, and a set of places is defined $\mathbf{P} = \{P_0, P_1, \ldots P_t, \ldots\}$. Each **P** represents a place location. For each place P_t, multiple observed point cloud frames $P_t = \{I_0, I_2, \ldots I_i, \ldots\}$ are defined. Through a point cloud feature extraction and aggregation model $f(.)$, a global descriptor for each observed point cloud frame is obtained by $g_i = f(I_i)$, thereby acquiring the place features for each location and constructing a point cloud place feature library as $G_t = \{g_0, g_1, \ldots \mathrm{g}_\mathrm{i}, \ldots\}$. Given a point cloud query as I_q, the global descriptor g_q of the query is similarly obtained through feature extraction and aggregation. The Euclidean distance between the query's global descriptor g_q and the elements of the point cloud place feature library is then calculated. Subsequently, the nearest neighbour search algorithm is utilised to find the most matching feature with a location label. This process ultimately achieves place recognition for the current query point cloud.

B. *Sparse Convolution Branch*

Tang *et al.* [11] proposed an efficient sparse point-voxel convolution (SPVConv) sparse convolution module, which consists of two branches: one for processing the original points and another for processing the voxelized point cloud. The original point processing branch preserves the high-resolution representation of the points and extracts features through a multilayer MLP network. The voxelized processing branch uses a sparse convolution network for feature extraction, and finally the features are fused and output.

To address the "sparse far, dense near" characteristic of vehicular point clouds (i.e., point clouds are sparse in regions far from the sensor and dense in regions close to the sensor), we have designed a hierarchical neighbourhood construction method to optimise the SPConv sparse convolution module to better balance the uneven density of point clouds and improve feature extraction performance (as shown in Fig. 3). First, a k-nearest neighbours (KNN) neighbourhood layer is constructed, where geometric position encoding is applied to the point cloud within the KNN neighbourhood, and a spherical query neighbourhood is built for each point. Then, robust principal component analysis (PCA) is used to extract point cloud features within the spherical query local neighbourhood. Next, the point cloud features extracted by both methods are weighted. Finally, the aggregated geometric position encoding features of the KNN neighbourhood points are obtained. This weighting and aggregation allow the points in the point cloud to not only gather geometric structural features, but also expand the receptive field for feature extraction. To better explain and clarify the point processing branch and the optimised SPConv sparse convolution module designed in this chapter, the point processing branch is named the local point feature extraction unit (LPFE), and the optimised SPConv sparse convolution module is referred to as the optimised-SPCNN module. These operations do not add learning parameters, but involve some mathematical computations, so they do not increase the complexity of the network.

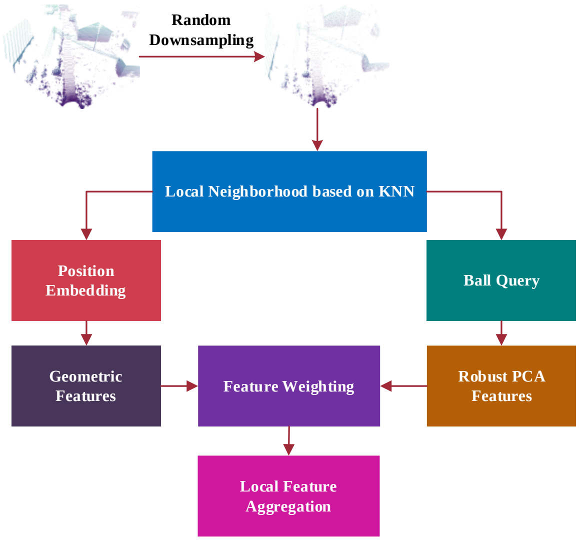


Fig. 3. Point branch of the sparse convolution.

Ball Query: To address the uneven density of vehicular point clouds, a radius-based spherical query method is first used to construct local neighbourhoods. This method defines a radius range (r = 1 m) around each point and finds neighbouring points within this range. In sparse regions, the spherical query ensures that even with sparse point clouds, sufficient neighbourhood information can still be obtained. On the basis of this, robust principal component analysis (robust PCA) is applied to extract features from the local point cloud. Given a point cloud ***P*** within the spherical query neighbourhood, robust PCA effectively removes noise and outliers, extracting robust initial features f_c

$$f_c = \text{RobustPCA}(\boldsymbol{P}), \boldsymbol{P} \in \mathbb{R}^{N\times 3}. \qquad (1)$$

Position Embedding: Positional encoding enables the establishment of topological or spatial connections among different elements, which can be regarded as a generalised graph model. Point clouds inherently exist in space. Despite the fact that the points within a point cloud are unordered, they possess a stable spatial structure in terms of relative positions. In this chapter, positional encoding is incorporated into the geometric structural characteristics of point clouds, which enables the network module to learn such spatial geometric structural characteristics in a more efficient manner. For each point $p_i = (x_i, y_i, z_i) \in \mathbb{R}^{1\times d}, d = 3$ in a point cloud, provided that the data dimension of its positional encoding is denoted as d_{pe}. Positional encoding ought to be conducted, respectively, on the x, y, and z dimensions of the point cloud, and the corresponding encoding features are f_i^x, f_i^y, and f_i^z. Taking f_i^x as an instance:

$$f_i^x(x_i, 2m) = \sin\left(\alpha x_i / \beta^{\frac{2*d*m}{d_{pe}}} \right),\ m \in [0, \frac{d_{pe}}{2*d}], \qquad (2)$$

$$f_i^x(x_i, 2m+1) = \cos\left(\alpha x_i / \beta^{\frac{2*d*m}{d_{pe}}} \right),\ m \in [0, \frac{d_{pe}}{2*d}], \qquad (3)$$

$$\boldsymbol{f}_{PE} = (f_i^x, f_i^y, f_i^z) \in \mathbb{R}^{1\times d_{pe}}. \qquad (4)$$

where α and β are two control variables that can adjust the learning distance between elements, capturing different spatial geometric structure information. $\boldsymbol{f}_{pe}$ is the geometric position encoding feature of the point.

Local Neighbourhood Feature Aggregation: As shown in Fig. 4, through the above steps, we first construct the local neighbourhood structure of the KNN. Taking point p_{query} as an example, points p_k^0, p_k^1, p_k^2 are its nearest neighbours found through the KNN search. Each of these points extracts preliminary features of the spherical neighbourhood through spherical querying and robust principal component analysis (PCA). These features are then aggregated onto the local features of point p_{query}, which expands the receptive field of p_{query} and aggregates robust point features. Taking point p_{query} as the centre, its robust principal component analysis (PCA) feature is denoted as f_c, and the robust PCA features of its neighbouring points are denoted as f_k^j. First, the features are merged to obtain the following

$$\boldsymbol{f}_{ck} = \left\{ f_c \oplus f_k^0, f_c \oplus f_k^1, \ldots, f_c \oplus f_k^k \right\}^T, j \in (0,1,\ldots k). \qquad (5)$$

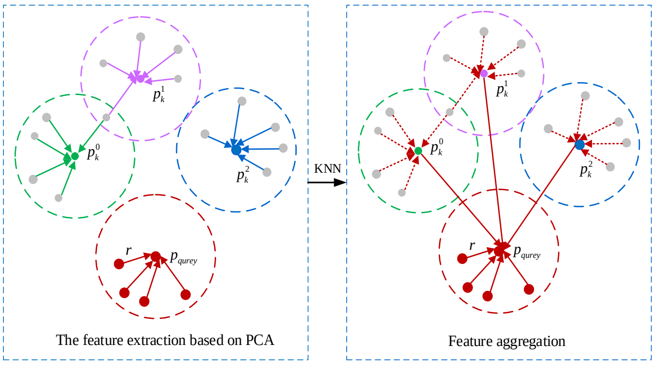


Fig. 4. Process of aggregation of local neighbourhood features.

During the aggregation of nearest neighbour features, similar to the method in [4], this paper employs a mechanism akin to attention pooling. First, a set of weight coefficients $W = (w_0, w_1, \ldots, w_{d_{RPCA}+d_{PE}})$ is initialised to learn the significance of features in each dimension. The geometric position encoding features $\boldsymbol{f}_{pe}$ are then combined with the robust PCA features $\boldsymbol{f}_{ck}$ and normalised using SoftMax to obtain the weight matrix for the neighbourhood features.

The weight matrix is multiplied element-wise by the feature matrix, and feature aggregation is performed through an MLP network. The final local neighbourhood features (as is shown in Fig. 5) are obtained as follows:

$$\boldsymbol{f}_{ck}^{w} = \text{SoftMax}(W \bullet \{f_{ck}^0 \oplus f_{pe}^0, f_{ck}^1 \oplus f_{pe}^1, \ldots, f_{ck}^k \oplus f_{pe}^k\}^{\mathrm{T}}), \qquad (6)$$

$$\boldsymbol{f}_c^{\mathrm{A}} = \text{MLP}(\boldsymbol{f}_{ck}^{w} \odot \boldsymbol{f}_{ck}). \qquad (7)$$

Finally, the framework of the optimised SPVConv sparse convolution module proposed in this paper is shown in Fig. 6.

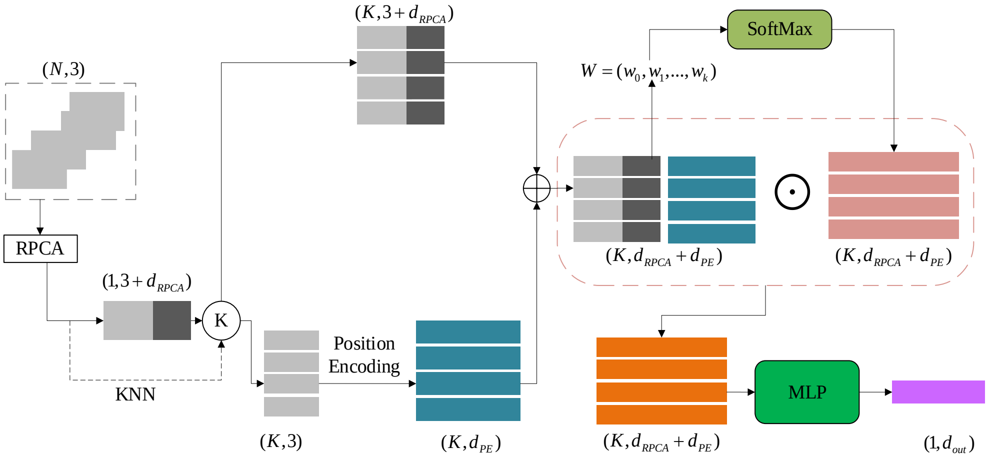


Fig. 5. Local point feature extraction (LPFE).

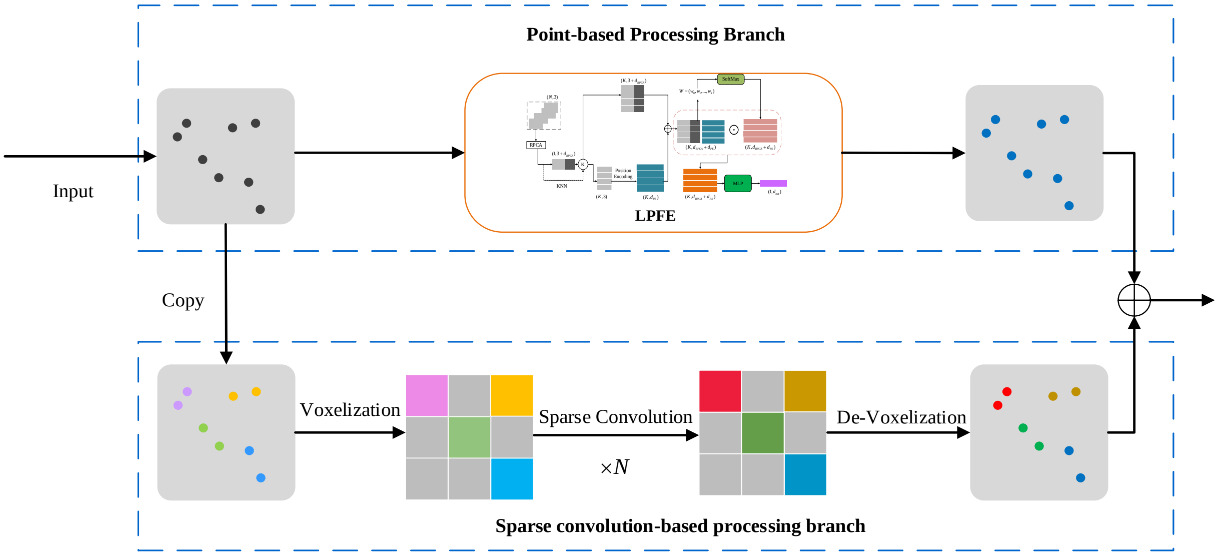


Fig. 6. Optimised-SPCNN.

C. *Feature Aggregation Design*

NetVLAD [23] has strong global description ability and can effectively utilise local features for aggregation to generate global features. In the BSPR-Net designed in this paper, NetVLAD is used to perform feature aggregation for both the BEV projection branch and the sparse convolution branch.

Feature Aggregation in the BEV Projection Branch:

f_i^b is the local features of the BEV projection branch, $i \in 0,1,...,N$

$$F^b = \{f_0^b, f_1^b, ..., f_N^b\} \in \mathbb{R}^{H\times W\times C}, N = H\times W, \tag{8}$$

where *H* and *W* are the size of the feature map, and *C* is the number of feature channels.

We randomly select a set of cluster centres as

$$C^b = \{c_0, c_1, ..., c_K\} \in \mathbb{R}^{K\times C}. \tag{9}$$

The global descriptor of the BEV projection branch:

$$f_g^b(k) = \sum_{i=0}^{N} a(f_i^b)(f_i^b - c_k), \tag{10}$$

$$\alpha(f_i^b) = \frac{e^{-\alpha\|f_i^b - c_k\|^2}}{\sum_{k'} e^{-\alpha\|f_i^b - c_{k'}\|^2}}, \tag{11}$$

$$\boldsymbol{F}_g^b = \{f_g^b(1), f_g^b(2), ..., f_g^b(K)\}^T. \tag{12}$$

Sparse Convolution Branch Feature Aggregation:

f_j^s is the local features of the sparse convolution branch, $j \in 0,1,...,M$

$$F^s = \{f_0^s, f_1^s, ..., f_M^s\} \in \mathbb{R}^{M\times C}, \tag{13}$$

where *M* is the number of local features output by the optimised-SPCNN module and *C* is the number of feature channels.

We randomly select a set of cluster centres as

$$C^s = \{c_0, c_1, ..., c_K\} \in \mathbb{R}^{K\times C}. \tag{14}$$

The global descriptor of the sparse convolution module branch is given by:

$$f_g^s(k) = \sum_{i=0}^{M} a(f_j^s)(f_j^s - c_k), \tag{15}$$

$$\alpha(f_j^s) = \frac{e^{-\alpha\|f_j^s - c_k\|^2}}{\sum_{k'} e^{-\alpha\|f_j^s - c_{k'}\|^2}}, \tag{16}$$

$$\boldsymbol{F}_g^s = \{f_g^s(1), f_g^s(2), ..., f_g^s(K)\}^T. \tag{17}$$

D. *Feature Fusion*

The role of the feature fusion module is to interactively aggregate the global descriptors $\boldsymbol{F}_g^b$ and $\boldsymbol{F}_g^s$ from the BEV projection branch and the sparse convolution module. We use the self-attention mechanism and the cross-attention mechanism from the transformer model for feature fusion of the two branches.

Self-Attention Computation: Enhanced self-attention global descriptor of $\boldsymbol{F}_g^s$ is V_{SA}^s

$$\boldsymbol{V}_{SA}^s = \boldsymbol{F}_g^s + \lambda \bullet SoftMax\{\boldsymbol{F}_g^s(\boldsymbol{F}_g^s)^T\}^T \boldsymbol{F}_g^s, \tag{18}$$

where λ is a learnable parameter. Similarly, the enhanced self-attention global descriptor of $\boldsymbol{F}_g^b$ is V_{SA}^b.

Cross-Attention Computation:

The two branches designed in this paper represent different point cloud characteristics. If the features of the two branches are directly concatenated during fusion, it cannot fully capture the inherent relationship between the different branches. Inspired by the work in [6], this paper uses cross-attention to capture the relationships between different channels of the global descriptors of the two branches, thus improving the model's performance.

Through self-attention computation, we have obtained the global enhanced descriptors from both the BEV projection branch and the sparse convolution module.

We obtain the global feature descriptor $V_g \in \mathbb{R}^{2K\times 1}$ through the cross-attention computation, and the calculation process is as follows:

$$V_g^b = V_{SA}^b + softMax\{\frac{V_{SA}^b (V_{SA}^s)^T}{\sqrt{K}}\} V_{SA}^s, \tag{19}$$

$$V_g^s = V_{SA}^s + softMax\{\frac{V_{SA}^s (V_{SA}^b)^T}{\sqrt{K}}\} V_{SA}^b, \tag{20}$$

$$V_g = \{V_g^b, V_g^s\}^T. \tag{21}$$

The global feature descriptor V_g^b and V_g^s integrates the significant attention features of the other branch, and they are concatenated to obtain the global feature descriptor.

E. Loss Function

Local Feature Consistency Loss: We adopt the idea of local feature consistency loss from [17] to guide network training, enhancing the accuracy and efficiency of cross-source point cloud location recognition. $D_{coord}(P_1^i, P_2^i)$ is the Euclidean distance of the spatial coordinates of the corresponding points in the source point cloud and the target point cloud; $D_{feat}(F_1^i, F_2^i)$ is the Euclidean distance of the features of the corresponding points in the source point cloud and the target point cloud:

$$D_i = \sqrt{D_{\text{coord}}\left(P_1^i, P_2^i\right)^2 + \sigma D_{\text{feat}}\left(F_1^i, F_2^i\right)^2}, \tag{22}$$

$$L_{local} = \frac{1}{2} \sum_{\left(P_1^i, P_2^i\right) \subset C_{1\leftarrow 2}^{+}} \left(D_i^2\right) + \frac{1}{2} \sum_{\left(P_1^3, P_2^j\right) \subset C_{1\leftarrow 2}^{-}} \max\left(0, m - D_j\right)^2, \tag{23}$$

where σ is a hyperparameter used to control the weight relationship between spatial coordinates and features, $C_{1\leftarrow 2}^{+}$ is the positive sample set, $C_{1\leftarrow 2}^{-}$ is the negative sample set, and *m* is a hyperparameter that represents the margin.

Global Feature Loss: The global feature loss uses a triplet loss to reduce the distance between the anchor point and the positive sample, while increasing the distance between the anchor point and the negative sample. The mathematical formula for the triplet loss function is as follows

$$L_{\text{triplet}} = \frac{1}{N}\sum_{i=1}^{N}\left[\max\left(0, \left\|V_g^A - V_g^P\right\|_2^2 - \left\|V_g^A - V_g^N\right\|_2^2 + \gamma\right)\right], \tag{24}$$

where *N* is the number of triplets in the training set, V_g^A is the global descriptor of the anchor point, V_g^P is the global descriptor of the positive sample, V_g^N is the global descriptor of the negative sample, $||\bullet||_2$ represents the Euclidean distance, and γ is a hyperparameter (Margin).

Joint Loss Function: Taking into account both the local consistency loss and the global descriptor loss, the joint loss function in this paper is as follows

$$L = L_g + \delta L_{local}. \tag{25}$$

Here, δ is a learnable hyperparameter, with an initial value set to 1.0.

IV. EXPERIMENTAL EVALUATION

To verify the effectiveness of the method proposed in this paper, we first evaluated the algorithm on the publicly available urban environment data set KITTI Odometry, and then evaluated it on the publicly available unstructured natural environment data set Wild-Places [24].

A. Data Sets and Evaluation Metrics

KITTI Data Set: The KITTI data set contains 11 data sequences, which were scanned using Velodyne HDL-64E LiDAR in real outdoor scenes. Each sequence includes a series of point clouds in the scene along with corresponding position and pose information. In the research experiment, we used these 11 sequences for training. Specifically, sequences 00, 02, and 08 provide a large number of point cloud revisit query frames, fulfilling the requirement for location recognition verification. In particular, queries for sequence 08 were collected by the vehicle driving in the opposite direction, so the performance of the model on sequence 08 better reflects its generalisability and effectiveness.

Wild-Places Data Set: The Wild-Places data set was collected in Brisbane National Park, Australia, using a handheld Velodyne VLP-16 LiDAR sensor over a 14-month period. The data cover more than 33 kilometers and include 63,319 LiDAR submaps. The data set includes eight LiDAR sequences collected from two environments, Venman and Karawatha, with diversity covering terrain features with sparse and dense vegetation. It is the first large-scale LiDAR point cloud data set used for studying point cloud-based location recognition in unstructured natural environments. The laser point cloud sequences 02 and 01 from the two environments were collected in reverse directions, sequence 03 was extended, and sequence 04 is almost the same as sequence 01, except for a later collection time. Each sequence contains multiple forward and reverse direction location revisits, supporting location re-identification evaluation. The data set includes 19,096 training submaps and 16,037 query submaps.

Evaluation Metrics: The maximum F1 score is the harmonic mean of precision and recall, and it is an important metric for evaluating the performance of classification models.

B. Comparative Experiment

Comparative Experiment on the KITTI Data Set: To evaluate the effectiveness and fairness of the BSPR-Net proposed in this paper, experiments were conducted on the KITTI data set, comparing it with four classic algorithms: ScanContext [25], PointNetLAD [26], Locus [27], LoGG3D-Net, and NFC-reLoc [28]. ScanContext is a handcrafted descriptor method that converts point clouds from Cartesian coordinates to a polar coordinate system after BEV projection. It divides the space into sector grids for encoding and has achieved efficient location recognition in urban environment data sets. It is commonly used in simultaneous localisation and mapping (SLAM) research for loop closure detection and serves as a benchmark for location recognition tasks. PointNetVLAD combines the PointNet and NetVLAD deep learning architectures, representing a milestone in point cloud-based location recognition research using raw points. Locus utilises multilevel feature descriptions of the scene and incorporates spatial and temporal information into the feature description stage, resulting in robust location recognition performance. LoGG3D-Net employs sparse convolution U-Net modules for feature extraction, introduces local consistency loss, and combines the second-order pooling idea from Locus, achieving outstanding performance in location recognition tasks. NFC-reLoc is an end-to-end place recognition network and has been a representative method in recent times. The maximum F1 metrics for the ScanContex, PointNetLAD, Locus, LoGG3D-Net, and NFC-reLoc methods are from their respective papers. Experiments on the proposed BSPR-Net method were conducted on an NVIDIA GFORCE GTX4090. The results are shown in Table I.

TABLE. I. PLACE RECOGNITION EXPERIMENTAL RESULTS ON THE KITTI DATA SET (MAX F1).

Method	00	02	08	Average
ScanContext	96.60	87.10	61.10	81.60
PointNetLAD	90.90	63.70	43.70	66.10
Locus	**98.30**	76.20	93.10	89.20
LoGG3D-Net	95.30	88.80	84.30	89.47
NFC-reLoc	95.10	89.90	83.60	89.53
BSPR-Net	97.30	**94.22**	**93.36**	**94.96**

In terms of the maximum F1 score, BSPR-Net achieves an average F1 value of 94.96 %, the highest among all methods, far surpassing the others. Particularly in data sets 02 and 08 sequences, the performance of BSPR-Net is especially remarkable, reaching 94.22 % and 93.36 %, respectively, indicating its strong recognition ability in complex scenarios. In comparison, NFC-reLoc, although also performing well with an average F1 value of 91 %, slightly lags behind BSPR-Net in the 02 sequence of certain data sets. ScanContext and PointNetLAD have average F1 values of 81.60 % and 66.10 %, significantly lower than BSPR-Net, with their performance notably deteriorating in the 08 sequence.

In terms of robustness, as shown in Fig. 7, BSPR-Net performs very stably across different data sequences (such as 00, 02, 08). Regardless of significant changes in the data set, BSPR-Net maintains a high recognition accuracy. This robustness gives BSPR-Net a significant advantage in diverse and complex scenarios, especially in practical applications. On the contrary, PointNetLAD performs poorly on the 08 sequence with an F1 score of only 66.10, showing weak overall stability. This limits its practical application.

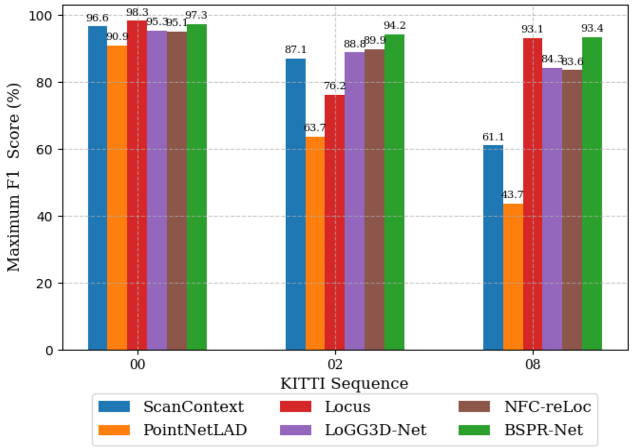


Fig. 7. Different methods on the KITTI data set

Overall, the BSPR-Net proposed in this paper outperforms all the compared methods, consistently maintaining high performance across multiple data sets, particularly in the 02 and 08 sequences. This makes BSPR-Net a reliable model for efficiently performing location recognition tasks in different environments.

Comparison Experiment on the Wild-Places Data Set: The Wild-Places data set is the first data set collected in unstructured natural environments specifically for location recognition research. The evaluation of different location recognition methods in the Wild-Places data set has practical significance for autonomous vehicles operating in off-road environments. To better assess the performance of BSPR-Net, this paper conducts experimental evaluations based on the comparison requirements and evaluation metrics proposed by Wild-Places. The methods compared in this experiment include ScanContext, TransLoc3D [21], MinkLoc3Dv2 [20], and LoGG3D-Net. ScanContext and LoGG3D-Net have been introduced earlier, so they will not be repeated here.

TransLoc3D improves point cloud location recognition performance by using a self-attention mechanism and multiscale feature fusion. MinkLoc3Dv2 uses a feature pyramid network (FPN) architecture to extract local features and applies a simple generalised mean pooling layer to obtain global features. It is a representative work utilising MinkowskiEngine sparse convolution. Wild-Places proposed an intersequence evaluation method. In this data set, the 03 and 04 sequences are duplicates of the 01 and 02 sequences, with a time gap of 6 and 14 months, respectively. Therefore, training is only done in sequences 01 and 02, while sequences 03 and 04 can be used to test the model's ability to generalise to environmental changes.

As shown in Table II, BSPR-Net performs excellently in terms of the F1 score, especially achieving high scores across multiple data sets. The average F1 score on the Wild-Places data set is 74.43, significantly higher than other methods. Compared to LoGG3D-Net, MinkLoc3Dv2, TransLoc3D, and ScanContext, the F1 score of BSPR-Net shows a significant advantage, indicating its superior accuracy and performance in location recognition tasks.

TABLE II. PLACE RECOGNITION EXPERIMENTAL RESULTS ON THE WILD-PLACES DATA SET (MAX F1).

Method	**Venman**		**Karawatha**		**Average**
	03	**04**	**03**	**04**	
ScanContext	1.01	13.00	13.22	32.26	14.87
TransLoc3D	17.09	60.83	47.22	66.99	48.03
MinkLoc3Dv2	49.78	82.19	51.33	80.00	65.83
LoGG3D-Net	53.94	80.42	64.30	84.54	70.08
NFC-reLoc	40.45	78.09	62.30	74.16	63.75
BSPR-Net	**60.26**	**82.25**	**69.75**	**85.46**	**74.43**

In terms of robustness, as shown in Fig. 8, BSPR-Net demonstrates strong robustness. In comparison, although LoGG3D-Net and MinkLoc3Dv2 perform well in some scenarios, their performance fluctuates significantly in other data sets. ScanContext, TransLoc3D, and NFC-reLoc exhibit poor robustness, especially in more complex extended scenarios such as Venman 03, where their scores drop significantly, showing lower adaptability and stability.

Comprehensive analysis indicates that the ScanContext method exhibits a performance disadvantage compared to deep learning approaches within the unstructured natural scene data set Wild-Places. The limitations of this method lie primarily in its inability to effectively handle nonlinear features in complex scenes and dynamic environmental variations. In sharp contrast, the BSPR-Net method demonstrates exceptional performance stability across all test sequences. This study successfully reproduced the NFC-reLoc method and conducted a systematic comparative analysis with BSPR-Net. Figure 9 clearly illustrates the precision-recall (PR) curves of both methods. Notably, on the Venman 04 sequence, the proposed BSPR-Net achieves a maximum F1 score of 85.46 %, demonstrating a significant 15.23 % performance advantage over NFC-reLoc.

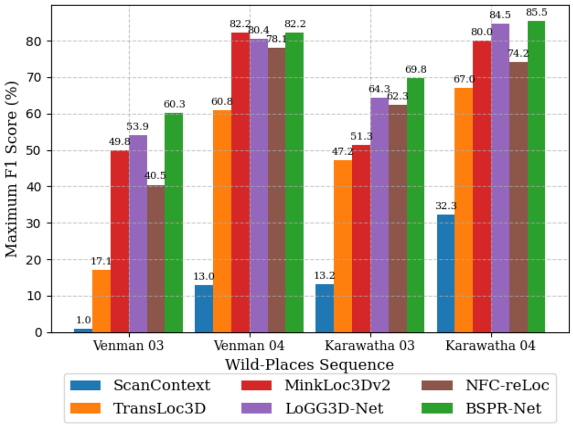
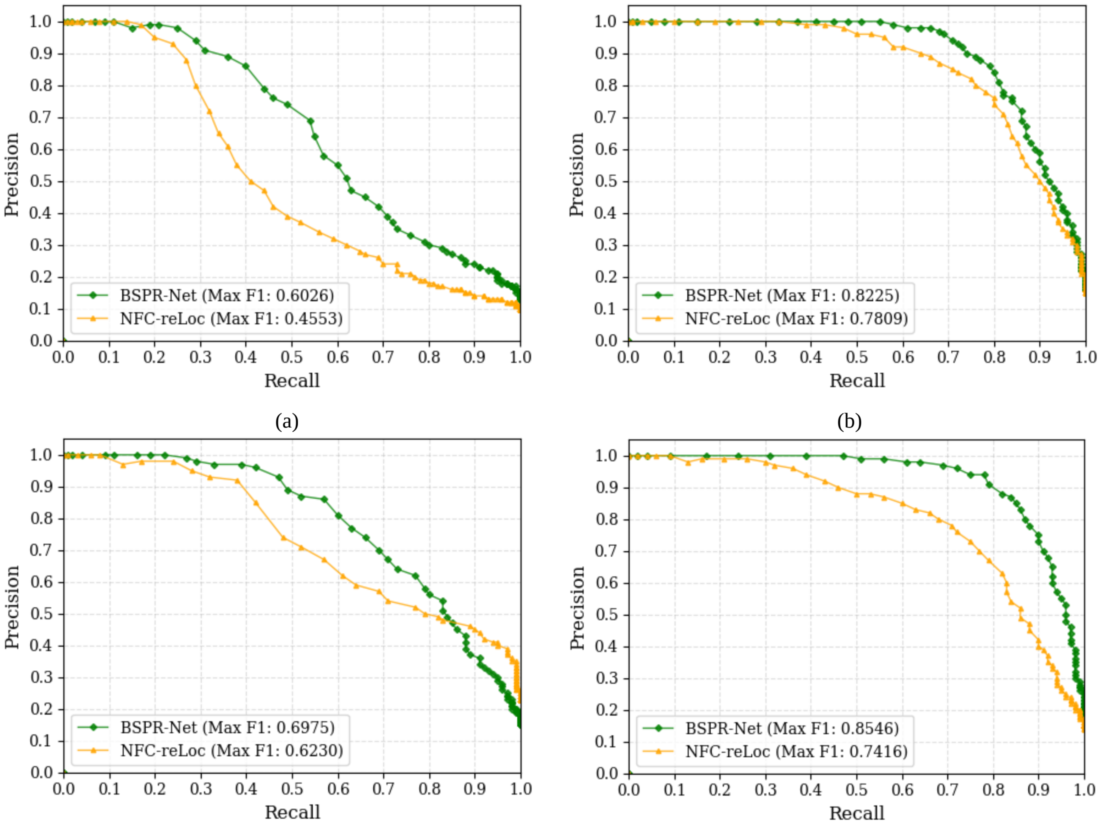


Fig. 8. Different methods on the Wild-Places data set.



(a)

(b)

(c)

(d)

Fig. 9. Precision-Recall curves for evaluation on selected sequences of the Wild-Places data set: (a) Venman 03; (b) Venman 04; (c) Karawatha 03; (d) Karawatha 04.

Comparison Experiment Summary: The KITTI data set mainly focusses on urban environments, which have rich structured features. ScanContext, LoGG3D-Net, and BSPR-Net can all extract significant features relatively well and maintain high performance in location recognition for the structured KITTI data set. The Wild-Places data set, on the other hand, focusses on natural outdoor environments, which lack structured features, making feature extraction more complex. As a result, the performance of all methods generally declines, especially the nonlearning method ScanContext, which significantly deteriorates and fails to effectively address the challenges of unstructured environments.

C. *Ablation Experiment*

To validate the effectiveness of the proposed modules, this section performs ablation experiments on the improvements made in the BSPR-Net network structure. The first ablation removes the BEV projection branch, and the network is labelled BSPR-without-BEV. The second ablation removes the sparse convolution point branch and replaces it with the original MLP network, resulting in the network being labelled as BSPR-without-LPFE. The final ablation removes the attention-based feature fusion module, directly concatenating the global descriptors of the two branches, and this network is labelled BSPR-without-Attention.

The experimental results for the KITTI data set are shown in Table III. Removing the BEV module leads to a performance drop across all tasks, especially in task 08, where the decrease exceeds 5 percentage points. The average F1 score drops to 86.12, which is about 5.6 percentage points lower than BSPR-Net (91.76). This indicates that BEV features play a significant role in feature enhancement for structured scenes. The impact of removing the LPFE module is milder compared to the BEV module, with the average F1 score dropping slightly to 86.26. Although the performance improved in task 05 (by about 5 percentage points), the performance decreased in other tasks. The LPFE module probably affects feature extraction and processing details, especially in various tasks, where LPFE provides the capability to handle details and complexity. The removal of the attention module had a smaller impact on performance compared to the removal of BEV and LPFE, but the results show that the attention module also plays a positive role in enhancing overall performance.

TABLE III. ABLATION EXPERIMENT RESULTS ON THE KITTI DATA SET (F1).

Method	00	02	08	Average
Without-BEV	89.36	85.28	82.21	86.12
Without-LPFE	87.75	82.25	85.45	86.26
Without-Attention	91.21	88.85	88.41	90.12
BSPR-Net	**97.30**	**94.22**	**93.36**	**94.96**

The experimental results on the Wild-Places data set are shown in Table IV. In unstructured scenes, the impact of removing BEV is more pronounced, especially in tasks Venman 03 and Karawatha 03, where the F1 scores decreased by approximately four and two percentage points, respectively. The average score drops to 69.81, which is 6.61 % lower than BSPR-Net. This indicates that BEV also plays an important role in complex natural scenes. The LPFE module enhances the ability to extract local structural features from point clouds in unstructured scenes. Removing it affects the model's ability to process these complex features, especially in Venman 03, where the performance drop is more significant. The average score is 67.11. Removing the attention module causes a more noticeable performance decline, especially in Karawatha 04, where the drop reaches approximately 10 %. The overall average is 66.44, which is about 12.02 % lower than BSPR-Net, highlighting the more prominent role of the attention module in unstructured scenes.

TABLE IV. ABLATION EXPERIMENT RESULTS ON THE WILD-PLACES DATA SET (F1).

Method	Venman		Karawatha		Average
	03	04	03	04	
Without-BEV	57.17	74.83	65.30	81.95	69.81
Without-LPFE	54.33	75.82	63.58	74.71	67.11
Without-Attention	55.29	75.24	62.14	73.11	66.44
BSPR-Net	**60.26**	**82.25**	**69.75**	**85.46**	**74.43**

V. CONCLUSIONS

In this paper, we propose a point cloud place recognition network, BSPR-Net, for unstructured natural environments. First, we adopt a dual-branch structure for global feature extraction from point clouds. A BEV projection branch is used to address the rotational invariance of the 2D motion plane in autonomous vehicles, while a 3D sparse convolution module branch captures spatial geometric structure information of the point clouds. Then, the two branch features are deeply fused through an attention mechanism to form a global descriptor.

On the KITTI data set, the proposed BSPR-Net achieves an average maximum F1 score of 94.96 %, surpassing other state-of-the-art methods by at least approximately 5 %. On the Wild-Places data set, it attains an average value of 74.43 %, outperforming existing leading methods by around at least 4 %. This shows the effectiveness of BSPR-Net and superiority over other advanced methods, especially in complex natural environments. Therefore, the proposed method offers a novel solution for place recognition or the global localisation of unmanned vehicles or robots operating in large-scale wild environments, demonstrating significant potential for widespread application.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] M. Elhousni and X. Huang, "A survey on 3D LiDAR localization for autonomous vehicles", in *Proc. of 2020 IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 1879–1884. DOI: 10.1109/IV47402.2020.9304812.

[2] P. Yin *et al.*, "ALITA: A large-scale incremental dataset for long-term autonomy", 2022. DOI: 10.48550/arXiv.2205.1073.

[3] P. Yin, A. Abuduweili, S. Zhao, L. Xu, C. Liu, and S. Scherer, "BioSLAM: A bioinspired lifelong memory system for general place recognition", *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4855–4874, 2023. DOI: 10.1109/TRO.2023.3306615.

[4] Z. Du, S. Ji, and K. Khoshelham, "3-D LiDAR-based place recognition techniques: A review of the past ten years", *IEEE Transactions on Instrumentation and Measurement*, vol. 73, art no. 2529024, pp. 1–24, 2024. DOI: 10.1109/TIM.2024.3403194.

[5] H. Yin *et al.*, "A survey on global LiDAR localization: Challenges, advances and open problems", *International Journal of Computer Vision*, vol. 132, no. 8, pp. 3139–3171, 2024. DOI: 10.1007/s11263-024-02019-5.

[6] K. Luo *et al.*, "3D point cloud-based place recognition: A survey", *Artificial Intelligence Review*, vol. 57, art. no. 83, 2024. DOI: 10.1007/s10462-024-10713-6.

[7] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite", in *Proc. of 2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361. DOI: 10.1109/CVPR.2012.6248074.

[8] G. Kim, Y. S. Park, Y. Cho, J. Jeong, and A. Kim, "MulRan: Multimodal range dataset for urban place recognition", in *Proc. of 2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 6246–6253. DOI: 10.1109/ICRA40945.2020.9197298.

[9] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan North Campus long-term vision and lidar dataset", *The International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2015. DOI: 10.1177/0278364915614638.

[10] T. Guan *et al.*, "CrossLoc3D: Aerial-ground cross-source 3D place recognition", in *Proc. of 2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 11301–11310. DOI: 10.1109/ICCV51070.2023.01041.

[11] H. Tang *et al.*, "Searching efficient 3D architectures with Sparse Point-Voxel Convolution", in *Proc. of Computer Vision – ECCV 2020: 16th European Conference*, 2020, pp. 685–702. DOI: 10.1007/978-3-030-58604-1_41.

[12] L. Luo, S.-Y. Cao, B. Han, H.-L. Shen, and J. Li, "BVMatch: Lidar-based place recognition using bird's-eye view images", *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 6076–6083, 2021. DOI: 10.1109/LRA.2021.3091386.

[13] X. Chen *et al.*, "OverlapNet: Loop closing for LiDAR-based SLAM", in *Proc. of the Conference Robotics: Science and Systems 2020*, 2020, pp. 1–10. DOI: 10.15607/RSS.2020.XVI.009.

[14] X. Xu *et al.*, "RING++: Roto-translation invariant gram for global localization on a sparse scan map", *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4616–4635, 2023. DOI: 10.1109/TRO.2023.3303035.

[15] J. Ma, J. Zhang, J. Xu, R. Ai, W. Gu, and X. Chen, "OverlapTransformer: An efficient and yaw-angle-invariant transformer network for LiDAR-based place recognition", *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6958–6965, 2022. DOI: 10.1109/LRA.2022.3178797.

[16] J. Ma, G. Xiong, J. Xu, and X. Chen, "CVTNet: A cross-view transformer network for LiDAR-based place recognition in autonomous driving environments", *IEEE Transactions on Industrial Informatics*, vol. 20, no. 3, pp. 4039–4048, 2024. DOI:

10.1109/TII.2023.3313635.

[17] T. Barros, L. Garrote, R. Pereira, C. Premebida, and U. J. Nunes, "AttDLNet: Attention-based deep network for 3D LiDAR place recognition", in *ROBOT2022: Fifth Iberian Robotics Conference. ROBOT 2022. Lecture Notes in Networks and Systems*, vol. 589. Springer, Cham, 2023, pp. 309–320. DOI: 10.1007/978-3-031-21065-5_26.

[18] L. Luo *et al.*, "BEVPlace: Learning LiDAR-based place recognition using bird's eye view images", in *Proc. of 2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8666–8675, 2023. DOI: 10.1109/ICCV51070.2023.00799.

[19] J. Komorowski, "Improving point cloud based place recognition with ranking-based loss and large batch training", in *Proc. of 2022 26th International Conference on Pattern Recognition (ICPR)*, 2022, pp. 3699–3705. DOI: 10.1109/ICPR56361.2022.9956458.

[20] K. Żywanowski, A. Banaszczyk, M. R. Nowicki, and J. Komorowski, "MinkLoc3D-SI: 3D LiDAR place recognition with sparse convolutions, spherical coordinates, and intensity", *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1079–1086, 2022. DOI: 10.1109/LRA.2021.3136863.

[21] T. Xu, Y.-C. Guo, Z. Li, G. Yu, Y.-K. Lai, and S.-H. Zhang, "TransLoc3D: Point cloud based large-scale place recognition using adaptive receptive fields", *Communications in Information and Systems*, vol. 23, no. 1, pp. 57–83, 2023. DOI: 10.4310/CIS.2023.v23.n1.a3.

[22] K. Vidanapathirana, M. Ramezani, P. Moghadam, S. Sridharan, and C. Fookes, "LoGG3D-Net: Locally guided global descriptor learning for 3D place recognition", in *Proc. of 2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2215–2221. DOI: 10.1109/ICRA46639.2022.9811753.

[23] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1437–1451, 2018. DOI: 10.1109/TPAMI.2017.2711011.

[24] J. Knights, K. Vidanapathirana, M. Ramezani, S. Sridharan, C. Fookes, and P. Moghadam, "Wild-places: A large-scale dataset for lidar place recognition in unstructured natural environments", in *Proc. of 2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 11322–11328. DOI: 10.1109/ICRA48891.2023.10160432.

[25] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map", in *Proc. of 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4802–4809. DOI: 10.1109/IROS.2018.8593953.

[26] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition", in *Proc. of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4470–4479. DOI: 10.1109/CVPR.2018.00470.

[27] K. Vidanapathirana, P. Moghadam, B. Harwood, M. Zhao, S. Sridharan, and C. Fookes, "Locus: LiDAR-based place recognition using spatiotemporal higher-order pooling", in *Proc. of 2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5075–5081. DOI: 10.1109/ICRA48506.2021.9560915.

[28] K. Kang, M. Lee, and H. Yu, "Necessity feature correspondence estimation for large-scale global place recognition and relocalization", 2023. DOI: 10.48550/arXiv.2303.06308.

]