OverlapMamba: A Shift State Space Model for LiDAR-Based Place Recognition

Jiehao Luo, *Student Member, IEEE*, Jintao Cheng, Qiuchi Xiang, Jin Wu, *Member, IEEE*,
Rui Fan, *Senior Member, IEEE*, Xieyuanli Chen, *Member, IEEE*, and Xiaoyu Tang, *Member, IEEE*

***Abstract*—Place recognition is the foundation for autonomous systems to achieve independent decision-making and secure operation. It is also crucial in tasks such as loop closure detection and global localization in Simultaneous Localization and Mapping (SLAM) technology. Existing LiDAR-based place recognition (LPR) methods use raw point cloud representations or multifarious point cloud representations as inputs, as well as employ convolutional neural networks or transformer architectures. However, the recently proposed Mamba deep learning model combined with State Space Models (SSMs) has enormous potential in long sequence modeling. Therefore, we have developed a novel place recognition network OverlapMamba, which represents input range images as sequences. In a novel way, we use a stochastic reconstruction method to establish shifted state space models to compress the visual representation. Extensive experiments on three public datasets demonstrate that OverlapMamba achieves competitive performance with real-time inference speed, which effectively detects loop closure even when traversing previously visited locations from different directions, indicating its strong place recognition ability and real-time efficiency.**

***Index Terms*—LiDAR-based place recognition, localization, SLAM, loop closure detection, mamba architecture.**

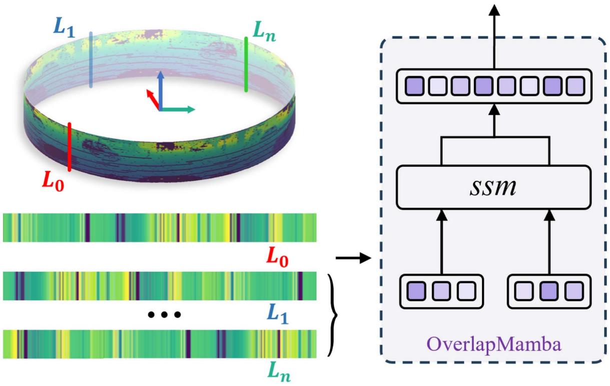

Fig. 1. Core idea of the proposed OverlapMamba model. The left parts represent range view (RV) projection and 1-D point cloud serialization. The right parts represent the overview of our novel state space models for place recognition.

I. Introduction

Place recognition (PR) is a key technology in autonomous driving and robotics that enables vehicles and robots to determine their location within complex environments. It plays a vital role in various tasks such as reliable vehicle localization, environmental mapping [1], and path planning, and effective PR systems must operate in real-time while maintaining low computational and memory requirements.

Previous methods relied heavily on hand-crafted features, which struggled with viewpoint variations, environmental changes, and failed to capture semantic relationships, leading to poor generalization. The paradigm has since shifted towards data-driven, learning-based PR models [2], [3], demonstrating remarkable performance gains. While vision-based methods [4] show promise, their limitations in challenging conditions have led researchers to explore LiDAR-based solutions [2], [3], [5], [6], which offer superior robustness across various environments. Recent investigations have explored multi-modal architectures [7] and multi-view fusion networks [8], leveraging complementary features from diverse data domains. However, despite achieving state-of-the-art performance and real-time operation, these sophisticated approaches incur significant computational overhead, conflicting with typical PR system resource constraints.

To recognize the same scenario, global descriptors generated by learning-based methods inherently converge towards yaw invariance during training for place matching. Through a

Received 18 February 2025; accepted 11 June 2025. Date of publication 20 June 2025; date of current version 10 July 2025. This article was recommended for publication by Associate Editor P. Checchin and Editor S. Behnke upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grant 62233013, in part by the Science and Technology Commission of Shanghai Municipal under Grant 22511104500, in part by the Fundamental Research Funds for the Central Universities, in part by Xiaomi Young Talents Program, and in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2024A1515012126. *(Corresponding author: Xiaoyu Tang.)*

Jiehao Luo and Qiuchi Xiang are with the School of Data Science and Engineering, Xingzhi College, South China Normal University, Shanwei 516600, China (e-mail: 20228132034@m.scnu.edu.cn; 20218131007@m.scnu.edu.cn).

Jintao Cheng and Xiaoyu Tang are with the School of Electronics and Information Engineering, Xingzhi College, South China Normal University, Foshan 528000, China (e-mail: 20172332035@m.scnu.edu.cn; tangxy@scnu.edu.cn).

Jin Wu was with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong SAR, China. He is now with the School of Intelligent Science and Technology, University of Science and Technology Beijing, Beijing 100083, China (e-mail: jin_wu_uestc@hotmail.com).

Rui Fan is with the College of Electronics & Information Engineering, Shanghai Research Institute for Intelligent Autonomous Systems, The State Key Laboratory of Intelligent Autonomous Systems, and Frontiers Science Center for Intelligent Autonomous Systems, Tongji University, Shanghai 201804, China (e-mail: rui.fan@ieee.org).

Xieyuanli Chen is with the College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410000, China (e-mail: xieyuanli.chen@nudt.edu.cn).

Our method has been implemented at http://github.com/SCNU-RISLAB/OverlapMamba.

Digital Object Identifier 10.1109/LRA.2025.3582109

detailed analysis of existing methods, we observe that many works [3], [8] tend to achieve yaw-invariant descriptors, while we aim to further improve this approach with more efficient sequence modeling capabilities provided by Mamba [9]. Furthermore, sequential data structures demonstrate superior capability in maintaining these properties compared to multi-dimensional tensors [10], [11].

To address these challenges, we propose OverlapMamba, a lightweight network for LiDAR-based place recognition that generates yaw-invariant descriptors through spatial sequence modeling while maintaining computational efficiency comparable to Transformer-based methods. Our main contributions are:

1) The proposal of a lightweight network that produces high-quality, yaw-equivariant symmetric feature representations.
2) The introduction of a specialized module for LiDAR sequence processing, which maintains the linear complexity and batch processing capabilities of SSMs.
3) The implement of bidirectional modeling strategy and SHIFT strategy for better capturing important environmental feature relationships.
4) Achieve leading performance on loop closure detection and place recognition tasks on the KITTI, NCLT and Ford Campus datasets.

II. Related Work

A. LPR Based on Local Description

LPR methods using local descriptors have evolved from handcrafted approaches to learning-based techniques. These methods extract distinctive features from point clouds to identify previously visited locations. For example, Zhou et al. [12] demonstrated that local 3D deep descriptors typically offer better generalization than global features for loop closure detection in various environments. Another significant advancement came from Ye et al. [13] who introduced FPET-Net, an efficient 3D point cloud place recognition approach based on feature point extraction and transformer architecture. Most recently, Kong et al. [14] proposed an interest point-driven approach that uses LeGO-LOAM, EdgeConv, and PointNet to generate robust global descriptors with significantly improved performance. However, these methods are still susceptible to viewpoint changes and rely on substantial computing power, which face limitations in processing sparse point clouds.

B. LPR Based on Global Description

Recent methods tend to use global descriptor to describe overall scene features, providing a comprehensive view of the data. Projection-based methods typically use various representations of data as input, such as RV, BEV and spherical view. Xieyuanli Chen et al. [5] and others proposed a network that can solve the problems of loop closure detection and place recognition. This method intuitively and effectively estimates the similarity between scan pairs by overlapping their RVs. Subsequently, OverlapTransformer [3] was introduced as an enhanced version of the previous model. On the basis of OverlapTransformer, Junyi Ma et al. [8] proposed a cross-view transformer network, which fused RVs and BEVs generated from LiDAR data. Luo et al. [15] developed BEVPlace, which employs yaw-invariant group convolutions on BEV, whereas our approach focuses on RV. In point-based methods, PTC-Net [16] introduced a novel Point-wise Transformer with sparse Convolution architecture. In point-based methods, MinkLoc3D [17] pioneered a sparse voxelized point cloud representation combined with 3D convolutions. However, their high computational requirements limit the batch sizes during training.

III. Overview of the Framework

A. Preliminaries

This paper explores the integration of Mamba [9] architecture into SLAM techniques for enhanced place recognition and global localization. The structured state space model (S4) based on SSM and Mamba [9] is inspired by continuous systems, which map a 1-D function or sequence $x(t) \in \mathbb{R}$ to $y(t) \in \mathbb{R}$ through a hidden state $h(t) \in \mathbb{R}^N$. Mathematically, they are often formulated as linear ordinary differential equations (ODEs), with parameters A serves as the evolution parameter, while B and C act as projection parameters. The dimensionality N represents the size of the hidden state and determines the model's capacity to capture complex patterns in the sequence:

$$\begin{aligned} h'(t) &= Ah(t) + Bx(t) \\ y(t) &= Ch(t) + Dx(t) \end{aligned} \tag{1}$$

As continuous-time models, SSMs face significant challenges when integrated into deep learning algorithms. Discretization is necessary to overcome this obstacle, and S4 and Mamba are discretized via Zero-Order Hold, which assumes the input signal remains constant over each sampling interval. This transforms the differential equation into a difference equation where Δ represents the sampling interval. The discretization process can be expressed as:

$$\begin{aligned} \bar{A} &= e^{\Delta A}, \\ \bar{B} &= (e^{\Delta A} - I)A^{-1}B \end{aligned} \tag{2}$$

After discretization, the linear ODEs representing SSMs can be rewritten as (3)

$$\begin{aligned} h_k &= \bar{A}h_{k-1} + \bar{B}x_k, \\ y_k &= Ch_k + Dx_k, \end{aligned} \tag{3}$$

In our OverlapMamba architecture, this discretization is implemented within Algorithm 1, where Δ is generated dynamically for each token through learned projection. This data-dependent approach enables adaptive modeling of spatial relationships in LiDAR data based on region-specific features.

In the context of LiDAR-based place recognition, A captures spatial relationships between sequential elements across viewing angles; B transforms depth measurements into feature space; C projects state features to maintain place-distinctive information; and D preserves geometric details through direct

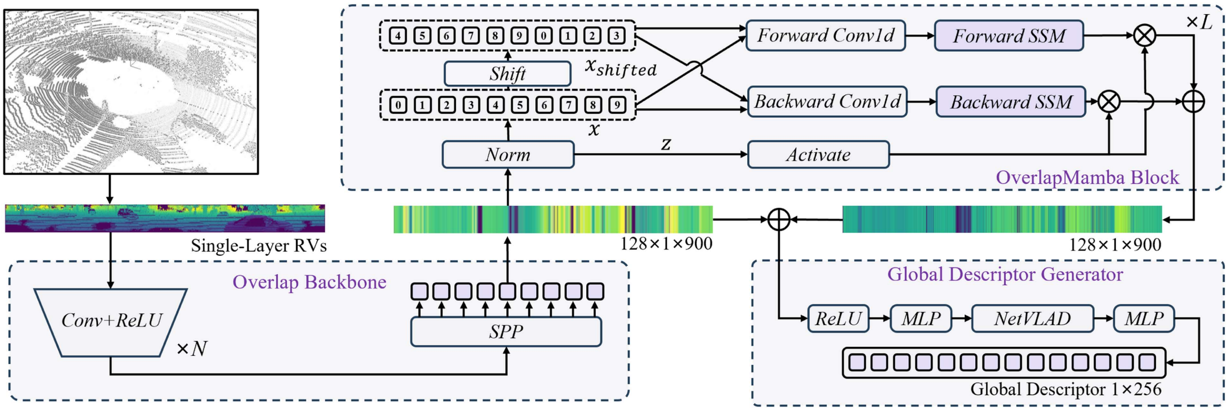

Fig. 2. Overview of the proposed OverlapMamba. The overlap backbone compresses the RVs from the LiDAR sensor information into yaw-equivariant feature sequences. The OverlapMamba block connects the feature sequences from the backbone with the multidirectionally enhanced feature sequences processed by the SSM. The global descriptor generator (GDG) utilizes a combination of multilayer peceptron (MLP) and NetVLAD to generate a 1-D global descriptor.

connections. This parameterization processes range image sequences by focusing on distinctive environmental features while handling LiDAR data sparsity.

B. Mamba-Based Place Recognition

The OverlapMamba architecture, illustrated in Fig. 2, consists of three main components: the overlap backbone, OverlapMamba block, and Global Descriptor Generator (GDG). Our model processes RV generated from raw LiDAR point cloud data. The projection transformation $\Pi : \mathbb{R}^3 \to \mathbb{R}^2$ maps each 3-D point cloud P to RV R, where the process of transforming each point $p_k = (x, y, z)$ into image coordinates (u, v) can be expressed as:

$$\begin{pmatrix} u_k \\ v_k \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\left[1 - \arctan(y_k, x_k)/\pi\right] w \\ \left[1 - \left(\arcsin\left(z_k/r_k\right) + f_{up}\right)/f\right] h \end{pmatrix} \tag{4}$$

where $r_k = ||p_k||_2$ is the distance measurement for the corresponding point p_k, $f = f_{\text{up}} + f_{\text{down}}$ is the vertical field of view of the sensor, and w, h are the width and height of the resulting RVs, respectively. Through a detailed analysis of existing methods [3], [8], we found that vertical resolution depends on the LiDAR scanner type, while a horizontal resolution of 900 pixels provides optimal balance between computational efficiency and descriptive power, capturing sufficient yaw-angle detail while maintaining real-time processing.

We employ single-channel RVs of size $1 \times h \times w$, applying vertical convolutional filters while preserving width following OverlapLeg [5], and introduce sequence pyramid pooling to address information loss and noise amplification.

Inspired by Vision Mamba [10], we leverage Mamba's efficient sequence processing. RVs are serialized into $x \in \mathbb{R}^{c\times 1\times w}$, where c denotes channel number and w represents width. Each x_{l-1} is processed through the l-th OverlapMamba encoder to generate x_l, which undergoes activation, normalization, and propagation to the GDG.

For yaw-invariant feature generation, we employ NetVLAD [18] in the GDG. Rotations of input LiDAR data translate to shifts in the distance image while maintaining descriptor consistency. The process is formulated as:

$$\begin{aligned} x_l &= Olm\left(x_{l-1}\right) + x_{l-1}, \\ n_l &= Norm\left(x_l\right), \\ \hat{g} &= GDG(n_l) \end{aligned} \tag{5}$$

where $Olm(.)$ represents the OverlapMamba block with residual connection, and $GDG(.)$ denotes the GDG that transforms the normalized sequence into the final global descriptor.

C. OverlapMamba Block

We propose the OverlapMamba block (OLM) architecture, as illustrated in Fig. 2, to effectively process spatial information in range images, which often contain empty regions from occlusions or missing returns. Mamba's selective state space model is ideal for this data through its gating mechanism that updates hidden states based on input importance, filtering uninformative regions while preserving structure. Unlike transformers that allocate equal resources to all position pairs, Mamba efficiently focuses computation on regions with discriminative features, particularly valuable for place recognition where key environmental features are unevenly distributed.

1) Bidirectional Sequence Modeling for Transformation Equivariance: For a feature sequence $Z = \{z_1, z_2, \ldots, z_w\}$ with length w, a unidirectional SSM processing tokens sequentially can be expressed as:

$$y_i = SSM(z_1, \ldots, z_i)$$

where y_i denotes the output at position i. However, when a shift operation in the sequence occurs, the transformation equivariance is broken as shifted tokens access different contextual windows, which can be expressed as:

$$y_i^s = SSM(z_{(1+s) \bmod w}, \ldots, z_{(i+s) \bmod w}) \tag{6}$$

$$y_{(i+s) \bmod w} = SSM(z_1, \ldots, z_{(i+s) \bmod w}) \tag{7}$$

$$y_i^s \neq y_{(i+s) \bmod w} \tag{8}$$

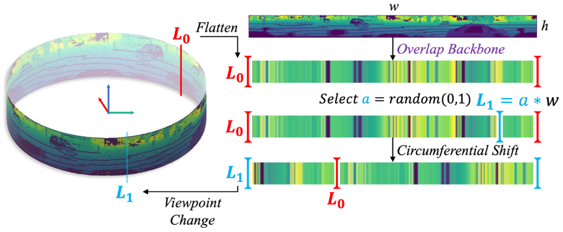

Fig. 3. The SHIFT operation process. The left part shows an example of a range image containing omnidirectional feature information. In the right part of the figure, we demonstrate the process of randomly reconstructing the feature sequence modeled along the horizontal direction for the yaw angle, where w is the width of the range image and a is a random parameter used to calculate the starting index of the reconstructed sequence.

where y_i^s denotes the shifted variant of y_i by s positions. To preserve the transformer equivariance, we design bidirectional SSM processing, which can be expressed as:

$$y_i' = SSM(z_1, \ldots, z_i) + SSM(Flip(z_w, \ldots, z_i)) \tag{9}$$

where $Flip(\cdot)$ denotes the operation of reversing the sequence order.The element-wise addition of forward and backward features ensures complete context access regardless of cyclic shifts, maintaining yaw equivariance. Based on this property, we can further enhance the feature representation by leveraging different viewing angles.

For any circular shift, our bidirectional approach maintains equivariance in our feature representation by combining complementary context from both directions, which can be demonstraed as:

$$\begin{aligned} y_i^s &= SSM(z_{(1+s) \bmod w}, \ldots, z_{(i+s) \bmod w}) \\ &\quad + SSM(Flip(z_{(w+s) \bmod w}, \ldots, z_{(i+s) \bmod w})) \\ &= SSM(z_{1 \bmod w}, \ldots, z_{i \bmod w})_{(i+s) \bmod w} \\ &\quad + SSM(Flip(z_{w \bmod w}, \ldots, z_{i \bmod w}))_{(i+s) \bmod w} \\ &= y_{(i+s) \bmod w} \end{aligned} \tag{10}$$

This equality holds because under circular shifts, the bidirectional SSM processes the same token set, just with different starting positions. The circular nature of the sequence preserves complete context in both directions, maintaining rotational equivariance.

2) SHIFT Operation and Yaw Invariance: The SHIFT operation is formally defined as a circular shifting function that reconstructs the input sequence starting from a randomly selected position:

$$Shift(Z) = \{z_{(1+s) \bmod w}, z_{(2+s) \bmod w}, \ldots, z_{(w+s) \bmod w}\} \tag{11}$$

where $s = \lfloor a \cdot w \rfloor$ is the starting index, with a being a random value sampled from the uniform distribution $[0, 1)$.

Token sequences inherently encode yaw information, with the reversed sequence representing the scene from opposite viewing directions. Due to the global nature of range images, token sequences form a cyclic pattern at different yaw angles within the same scene, as shown in Fig. 3. While the SHIFT operation introduces intentional randomness during training to improve generalization across different yaw angles, it is designed to maintain deterministic inference behavior. The random parameter used for sequence reconstruction varies during training to simulate different viewing angles, but the data is not processed through branches involving stochastic reconstruction during inference for efficiency and reliability.

Algorithm 1: OverlapMamba Block Process.

```
Input: token sequence $T_{l-1}$: (B,M,D)
Output: token sequence $T_l$: (B,M,D)
/* normalize the input sequence $T_{l-1}$ */
$T'_{l-1}$: (B,M,D) $\leftarrow$ Norm($T_{l-1}$)
$x_{forward}$, z: (B,M,E) $\leftarrow$ $Linear^x(T'_{l-1})$, $Linear^z(T'_{l-1})$
/* process with random yaw and different directions*/
$x_{shift}$: (B,M,E)$\leftarrow$ Shift($x_{forward}$)
$x_{backward}$, $x_{shift\_backward}$: (B,M,E)$\leftarrow$ Flip($x_{forward}$), Flip($x_{shift}$)
for $o$ in {forward, backward, shifted_forward,
 shifted_backward}
   $x'_o$: (B,M,E) $\leftarrow$ SiLU($Conv1d_o(x)$)
   $A_o \leftarrow \exp(Parameter_o^A)$
   $D_o \leftarrow Parameter_o^D$
   $\Delta$, $B_o$, $C_o$ $\leftarrow$ Split(Linear($x'_o$))
   $\Delta_o$: (B,M,E) $\leftarrow$ Softplus($Linear_o^{\Delta}(\Delta)$)
   $\overline{A}_o$, $\overline{B}_o$ : (B,M,E,N) $\leftarrow$ $\Delta_o \otimes A_o$, $\Delta_o \otimes B_o$
   $y_o$ : (B,M,E) $\leftarrow$ SSM($\overline{A}_o, \overline{B}_o, C_o, D_o$)($x'_o$)
end for
/* get $y'$ */
for $i$ in {forward, backward, shifted_forward,
 shifted_backward}
   $y'_i$ : (B,M,D) $\leftarrow$ $y_i \odot$ SiLU(z)
end for
/* residual connection */
$T_l$ : (B,M,D) $\leftarrow$ $Linear^T$(Sum($y'$)) + $T_{l-1}$
Return: $T_l$
```

Since NetVLAD is proved to have yaw invariance [3], the GDG process also maintains yaw invariance, which can be demonstrated as:

$$\begin{aligned} GDG(Z) &= NetVLAD(\{z_1, z_2, \ldots, z_w\}) \\ &= NetVLAD(\{z_{(1+s) \bmod w}, \ldots, z_{(w+s) \bmod w}\}) \\ &= NetVLAD(Shift(Z)) \end{aligned} \tag{12}$$

This invariance is guaranteed through NetVLAD's permutation-invariant pooling. Our bidirectional SSM ensures feature sets remain identical under circular shifts, while NetVLAD's design maintains invariance to feature ordering, producing consistent global descriptors regardless of yaw angle.

3) OverlapMamba Algorithm: The OverlapMamba (OLM) block integrates multidirectional sequence modeling for place recognition, with operations detailed in Algorithm 1. Parameterized by module stacking number L, hidden state dimension D, extended state dimension E, and SSM dimension

N, the block processes input sequences through these key steps:

Normalization and projection: Input token sequence T_{l-1} is normalized to stabilize training before being projected through parallel linear transformations into primary features x and gating features z.

Directional sequence generation: Four processing paths are created by applying forward/backward operations and stochastic shifts to x, simulating different yaw angles during training.

Parameter generation: Each directional sequence undergoes convolution, activation, and linear projections to generate SSM parameters. Softplus ensures positive time-steps Δ for stability, while tensor broadcasting $\otimes$ efficiently applies parameters across sequence positions.

Selective filtering: The SSM outputs are modulated by z through SiLU activation, enabling selective feature emphasis based on input relevance.

Integration: All directional outputs are combined and projected back to the original feature space, with a residual connection preserving gradient flow.

This multidirectional approach enhances model generalization by simulating features under different viewing angles.

D. Sequential Pyramid Pooling in the Backbone

The range image processing divides the image into H sequences of length W along the horizontal dimension. Our overlap backbone uses vertical convolutional filters to compress these images into $c \times 1 \times w$ feature sequences, preserving critical yaw information along the width dimension. To handle object distortions and noise interference in range images, we propose a simplified Spatial Pyramid Pooling (SPP) module inspired by [19].

While traditional SPP operates on 2D feature maps to capture multi-scale information through pooling at different grid resolutions, our SPP design employs two convolutional layers along the horizontal dimension to compress input or expand intermediate states, performing three consecutive max-pooling operations followed by channel compression. The pooling operations apply sequentially rather than in parallel branches, with each operation building upon the previous output, effectively enhancing position invariance while mitigating noise-induced feature loss.

E. Improved Triplet Loss With Hard Mining

Our ImTrihard Loss builds upon the conventional overlap-based supervision triplet loss [5], [8]. However, it exhibits limitations in capturing subtle feature distinctions. This limitation primarily stems from the non-uniform distribution of training samples, where random sampling leads to easily distinguishable sample pairs that provide limited learning signals. To address this issue, we propose a modified triplet loss that emphasizes the most discriminative samples.

For a given query descriptor g_q, we compute the distances to all positive samples $\{g_p\}$ and all negative samples $\{g_n\}$ in the batch. Unlike standard triplet loss that averages these distances, ImTrihard Loss focuses on the hardest positive sample (maximum distance to query) and the hardest negative sample (minimum distance to query):

$$\begin{aligned}\mathcal{L}_T\left(g_q,\left\{g_p\right\},\left\{g_n\right\}\right) &= \lambda d\left(g_q, g_p\right) \\ &\quad + k_p\left(\alpha+\max_p\left(d\left(g_q, g_p\right)\right)\right) \\ &\quad - k_n \min_n\left(d\left(g_q, g_n\right)\right) \qquad (13)\end{aligned}$$

where α denotes the margin, λ denotes a compression coefficient, $d(\cdot)$ computes the squared Euclidean distance, and k_p and k_n are normalization factors based on the number of positive and negative samples respectively.

We strictly follow the sample selection strategy as [3], constructing training triplets. During training, scan pairs with overlap scores above 0.3 are designated as positive samples, while those below this threshold are negative samples. The additional term $\lambda d(g_q, g_p)$ ensures consistent feature learning by maintaining an absolute distance constraint between query and positive samples.

IV. Experiments

A. Experimental Setup

We evaluate our method on KITTI [20], Ford Campus [21], and NCLT [22] datasets. KITTI provides urban, rural, and highway scenes with varying numbers of vehicles and pedestrians. Ford Campus dataset is collected from an autonomous vehicle platform with multiple loop closures. NCLT contains indoor and outdoor campus trajectories across different times and seasons. We using every LiDAR frame, adopting the same RV projection strategy and parameters setting as [3], [8]. For the NCLT dataset, we select sequence 2012-01-08 for training and constructing database, while the sequences 2012-02-05 is used for querying. For the KITTI Ford Campus datasets, we use the same training and evaluation strategy as [3] to ensure the fairness of the comparison experiments.

We employ AUC, F1max, Recall@1, and Recall@1% metrics for consistency with prior research. These metrics have inherent limitations as the overlap threshold selection directly influences true/false positive classifications and consequently affects F1max scores. Furthermore, these standard metrics may favor methods optimized for moderate viewpoint changes while inadequately evaluating performance in extreme rotation scenarios.

For OverlapMamba implementation, we employ a single layer with embedding dimension $d_{model} = 256$. The processed and unprocessed sequences are summed one-to-one for random yaw augmentation. The SPP module uses a pooling kernel size of 5 with appropriate padding. We configure the NetVLAD module following [3]. This configuration leverages NetVLAD's inherent permutation invariance property, which ensures that reordering input features, corresponding to yaw rotations in LiDAR scans, does not affect the final descriptor, thereby achieving robust yaw invariance. Training uses Adam optimizer with initial learning rate 5×10^{-6} for 20 epochs, using only LiDAR point cloud data without fine-tuning. For comparative analysis, some results are referenced from OverlapTransformer, and to ensure fair comparison, we maintain identical experimental configurations.

TABLE I
COMPARISON OF LOOP CLOSURE DETECTION PERFORMANCE IN KITTI AND FORD CAMPUS DATASET

Dataset	Approach	AUC	F1max	Recall @1	Recall @1%
KITTI	Scan Context [23]	0.836	0.835	0.820	0.869
	PointNetVLAD [24]	0.856	0.846	0.776	0.845
	OverlapNet [5]	0.867	0.865	0.816	0.908
	OverlapTransformer [3]	0.907	0.877	**0.906**	**0.964**
	MinkLoc3D [17]	0.894	0.869	0.876	0.920
	CVTNet [8]	0.911	0.880	-	-
	BEVPlace [15]	0.908	0.875	0.889	0.953
	OverlapMamba(Ours)	**0.934**	**0.890**	0.898	0.959
Ford Campus	Scan Context [23]	0.903	0.842	0.878	**0.958**
	PointNetVLAD [24]	0.872	0.830	0.862	0.938
	OverlapNet [5]	0.854	0.843	0.857	0.932
	OverlapTransformer [3]	0.923	0.856	**0.914**	0.954
	MinkLoc3D [17]	0.871	0.851	0.878	0.942
	OvelapMamba(Ours)	**0.929**	**0.871**	0.909	0.957

Following OverlapTransformer [3], we trained on KITTI sequences 03∼10 and evaluated on sequences 00 and 02. Loop closures were determined using an overlap threshold of 0.3, with a maximum of 6 positive and negative samples each.

B. Evaluation for Loop Closure Detection

The experimental results validate the effectiveness of our approach in large-scale outdoor LiDAR-based localization and loop closure detection, demonstrating strong generalization across different environments. The quantitative evaluation results are presented in Table I. OverlapMamba, utilizing only depth range images, achieves an AUC of 0.934 and F1max of 0.890 on the KITTI dataset. Compared to CVTNet with its dual RV and BEV inputs, our single-branch method demonstrates enhanced performance with a 2.3% improvement in F1max, while also outperforming BEVPlace (which uses only BEV input) with improvements of 2.6% in AUC and 1.5% in F1max. Furthermore, our approach surpasses the current state-of-the-art OverlapTransformer by 1.3% in F1max, while achieving competitive recall@1 of 0.898 and recall@1% of 0.959. The slightly lower recall metrics compared to OverlapTransformer (underlined in the table) suggest that while our method excels at overall place recognition accuracy, it occasionally ranks the single most similar place with slightly less precision. This trade-off likely stems from our focus on optimizing the global descriptor's discriminative power across varying viewpoints rather than maximizing single-candidate ranking precision.

To verify cross-dataset generalization, we evaluate the model trained on KITTI sequences 03-10 directly on the Ford dataset. OverlapMamba achieves an F1max of 0.871, exceeding the previous best results from OverlapTransformer by 2.1%.

The performance improvements across experiments likely stem from the SSM architecture's efficiency in modeling long-range dependencies in range images and our bidirectional strategy with SHIFT operations that enhances yaw invariance. These architectural innovations, rather than dataset-specific optimizations, explain our consistent performance gains across diverse environments.

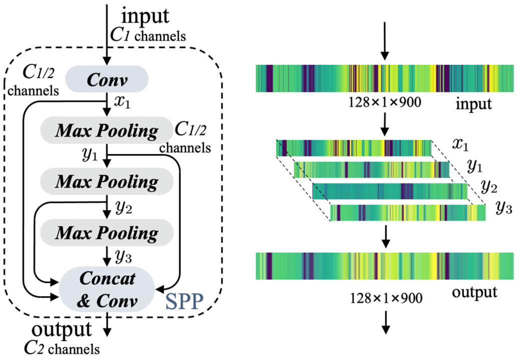

Fig. 4. The structure of the SPP module. Three consecutive 1-D pooling operations are performed in the block, and the intermediate states are concatenated along the vertical dimension and processed by a convolution layer to obtain the output.

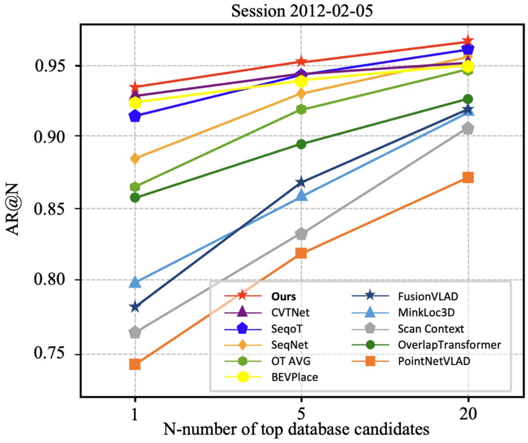

Fig. 5. Place recognition results using the NCLT dataset session 2012-02-05 as the query and 2012-01-08 as the database.

While we evaluate place recognition and loop closure detection in isolation, OverlapMamba's lightweight architecture enables seamless integration with complete SLAM systems without significant computational overhead. Our global descriptors directly interface with standard pose graph optimization frameworks to correct accumulated drift.

C. Evaluation for Place Recognition

The experimental results on the NCLT dataset further validate our method's effectiveness in place recognition tasks. Following the protocol established in OverlapTransformer [3], we train our model using the database from 2012-01-08 and evaluate on query sequences from 2012-02-05.

As shown in Fig. 6, OverlapMamba demonstrates superior performance with only RV input, improving AR@1 by 1.30% and AR@20 by 4.13% compared to CVTNet [8], which utilizes both RV and BEV inputs, and consistently outperforming BEVPlace across all N values on both NCLT sessions despite

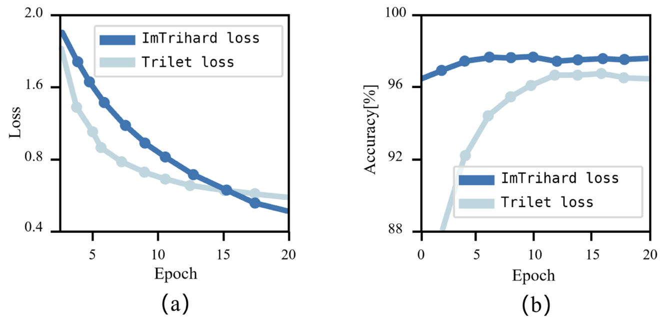

Fig. 6. Comparison of two loss functions. (a) shows the change in the loss value and (b) shows the evaluation on the sequence 00 of the KITTI dataset.

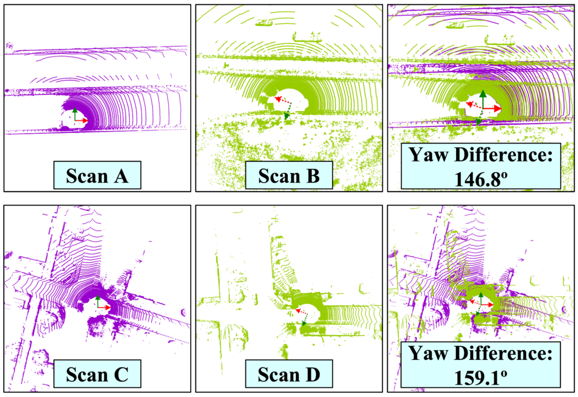

Fig. 7. LiDAR scan pairs showing the same locations from dramatically different viewing angles. Top row: Scan A (purple) and Scan B (green) of the same location with a yaw difference of 146.8°. Bottom row: Scan C (purple) and Scan D (green) from another location with an even larger yaw difference of 159.1°. Right column shows the overlapped scans with coordinate axes indicating viewing directions. OverlapMamba successfully identifies these as matching locations despite the extreme perspective differences.

its specialized BEV representation. Our model achieves the highest AR@1 and AR@20 scores across all baselines including methods using single BEV input and combined RV-BEV representations, demonstrating that our architecture can effectively capture spatial-temporal dependencies even with a simpler input representation. This validates the effectiveness of our approach in challenging place recognition scenarios where viewpoint and environmental conditions vary significantly.

OverlapMamba excels in challenging scenarios with extreme viewpoint variations. As shown in Fig. 7, our method successfully identifies matching locations despite yaw differences exceeding 140°. This viewpoint-invariant capability derives from our bidirectional sequence modeling with SHIFT operations, enabling robust place recognition regardless of observation angle.

D. Ablation Study on Mamba Modules

We conduct comprehensive ablation studies to validate the effectiveness of each proposed component in OverlapMamba, with results shown in Table II. Using OverlapTransformer as baseline, Mamba-based descriptor processing achieves superior performance with linear complexity (setting i). Each additional component further enhances the system performance (setting ii), while their combinations (setting iii) validate the necessity of each module.

TABLE II
Ablation Experiments With Proposed Modules on KITTI Dataset

Methods	Component			AUC	F1max
	Shift	SPP	ImTrihard Loss		
Baseline				0.891	0.842
OverlapMamba (i)	-	-	-	0.898	0.843
OverlapMamba (ii)	✓	-	-	0.901	0.848
	-	✓	-	0.882	0.845
	-	-	✓	0.881	0.850
OverlapMamba (iii)	✓	✓	-	0.930	0.872
	✓	-	✓	0.913	0.858
	-	✓	✓	0.926	0.857
	✓	✓	✓	**0.934**	**0.890**

TABLE III
Comparison of Different Numbers of OverlapMamba Blocks on the KITTI Dataset

Number	Runtime(ms)	AUC	F1max
1	5.1	0.934	0.890
2	7.8	0.848	0.803
3	10.4	0.822	0.782

TABLE IV
Comparison of Convergence Speed in the Training Process of OverlapMamba on Two Loss Functions

	Trilet loss (original)		ImTrihard loss	
Epoch	Loss	F1max	Loss	F1max
1	1.231	0.776	1.925	0.872
5	0.803	0.826	1.431	0.880
10	0.667	**0.844**	1.003	0.888
20	0.571	0.832	0.557	**0.890**

Further analysis on model depth, as shown in Table III, which reports the metrics and module inference time including forward pass and post-processing, reveals that a single OverlapMamba module achieves optimal results with an AUC of 0.934 and F1max of 0.890, while deeper architectures increase runtime from 5.1 ms to 10.4 ms without performance gains. This finding suggests that for RV sequences, a single layer effectively captures spatial relationships. The SHIFT operation enhances the model's ability to handle viewpoint variations by simulating different yaw angles, which complements the SSM's sequential modeling capabilities. Unlike transformers where depth helps establish global context, SSMs can model long-range dependencies even with a single layer, while additional layers appear to disrupt the delicate balance between feature discrimination and generalization for place recognition tasks.

E. Study on ImTrihard Loss

We validate the effectiveness of our proposed ImTrihard loss function through experiments on the KITTI dataset, with results shown in Table IV and Fig. 6. The ImTrihard loss demonstrates remarkable convergence speed, achieving an F1max score of 0.872 in the first epoch. As illustrated in Fig. 6(b), the accuracy on KITTI sequence 00 reaches 96.43% after just one epoch. While the traditional triplet loss shows signs of overfitting with a 1.2% decrease in F1max from epoch 10 to epoch 20, ImTrihard

TABLE V
COMPARISON OF RUNTIME WITH STATE-OF-THE-ART METHODS

	Approach	Descriptor Extraction [ms]	Searching [ms]
Hand crafted	Scan Context [23]	57.95	492.63
Learning based	PointNetVLAD [24]	13.87	1.43
	OverlapNet [5]	4.85	3233.30
	MinkLoc3D [17]	15.94	8.10
	OverlapTransformer [3]	1.37	0.44
	Ours	**0.49**	**0.35**

loss maintains stable performance throughout training. Fig. 6(a) visualizes the loss convergence patterns, where ImTrihard loss, despite initially higher values due to hard sample selection, exhibits consistent and stable optimization. These results validate that ImTrihard loss not only accelerates training convergence but also enhances model generalization.

Furthermore, training experiments reveal OverlapMamba's excellent stability with no gradient instability, showing consistent convergence regardless of initialization strategy. This stability stems from the natural alignment between SSM and the circular structure of range views.

F. Runtime

We evaluate the inference efficiency of our method using the same configuration as [3]. As shown in Table V, OverlapMamba achieves superior efficiency among all methods. This performance advantage in searching time may stems from the efficient SSM scan and the generation of descriptive descriptors by the Mamba architecture, which demonstrates significantly faster inference compared to similarly sized transformers or MinkLoc3D [17].

V. CONCLUSION

In this paper, we propose a LPR network integrating Mamba module into the architecture. Extensive experiments prove that OverlapMamba outperforms other SoTA algorithms on three public datasets in accuracy, complexity and speed with simple information inputs. The integration of SSMs advances LiDAR-based place recognition with potential applications in semantic scene understanding and multi-modal perception. Despite strong performance, like other single-modality approaches, our method may face challenges in environments where depth information alone is insufficient compared to multi-modal methods.

REFERENCES

[1] K. Muravyev, A. Melekhin, D. Yudin, and K. Yakovlev, "PRISM-TopoMap: Online topological mapping with place recognition and scan matching," *IEEE Robot. Automat. Lett.*, vol. 10, no. 4, pp. 3126–3133, Apr. 2025.

[2] J. Ma, X. Chen, J. Xu, and G. Xiong, "SeqOT: A spatial–temporal transformer network for place recognition using sequential LiDAR data," *IEEE Trans. Ind. Electron.*, vol. 70, no. 8, pp. 8225–8234, Aug. 2023.

[3] 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 Robot. Automat. Lett.*, vol. 7, no. 3, pp. 6958–6965, Jul. 2022.

[4] J. Zhao et al., "Learning sequence descriptor based on spatio-temporal attention for visual place recognition," *IEEE Robot. Automat. Lett.*, vol. 9, no. 3, pp. 2351–2358, Mar. 2024.

[5] X. Chen et al., "OverlapNet: A siamese network for computing lidar scan similarity with applications to loop closing and localization," *Auton. Robots*, pp. 1–21, 2022.

[6] X. Xu et al., "RING++: Roto-translation-invariant gram for global localization on a sparse scan map," *IEEE Trans. Robot.*, vol. 39, no. 6, pp. 4616–4635, Dec. 2023.

[7] B. Liu, T. Yang, Y. Fang, and Z. Yan, "MICL: Mutual information guided continual learning for LiDAR place recognition," *IEEE Robot. Automat. Lett.*, vol. 9, no. 11, pp. 10463–10470, Nov. 2024.

[8] 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 Trans. Ind. Inform.*, vol. 20, no. 3, pp. 4039–4048, Mar. 2024.

[9] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," 2023, *arXiv:2312.00752*.

[10] L. Zhu, B. Liao, Q. Zhang, X. Wang, W. Liu, and X. Wang, "Vision mamba: Efficient visual representation learning with bidirectional state space model," 2024, *arXiv:2401.09417*.

[11] J. Ruan and S. Xiang, "VM-UNet: Vision mamba UNet for medical image segmentation," 2024, *arXiv:2402.02491*.

[12] Y. Zhou, Y. Wang, F. Poiesi, Q. Qin, and Y. Wan, "Loop closure detection using local 3D deep descriptors," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 6335–6342, Jul. 2022.

[13] T. Ye, X. Yan, S. Wang, Y. Li, and F. Zhou, "An efficient 3-D point cloud place recognition approach based on feature point extraction and transformer," *IEEE Trans. Instrum. Meas.*, vol. 71, 2022, Art. no. 5023109.

[14] D. Kong et al., "Explicit points-of-interest driven siamese transformer for 3D LiDAR place recognition in outdoor challenging environments," *IEEE Trans. Ind. Informat.*, vol. 19, no. 10, pp. 10564–10577, Oct. 2023.

[15] L. Luo et al., "BEVPlace: Learning LiDAR-based place recognition using bird's eye view images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 8700–8709.

[16] L. Chen, H. Wang, H. Kong, W. Yang, and M. Ren, "PTC-Net: Point-wise transformer with sparse convolution network for place recognition," *IEEE Robot. Automat. Lett.*, vol. 8, no. 6, pp. 3414–3421, Jun. 2023.

[17] J. Komorowski, "MinkLoc3D: Point cloud based large-scale place recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 1790–1799.

[18] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1437–1451, Jun. 2018.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

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

[21] G. Pandey, J. R. McBride, and R. M. Eustice, "Ford campus vision and lidar data set," *Int. J. Robot. Res.*, vol. 30, no. 13, pp. 1543–1552, 2011.

[22] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan north campus long-term vision and lidar dataset," *Int. J. Robot. Res.*, vol. 35, no. 9, pp. 1023–1035, 2016.

[23] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 4802–4809.

[24] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4470–4479.