CVTNet: A Cross-View Transformer Network for LiDAR-Based Place Recognition in Autonomous Driving Environments

Junyi Ma, Guangming Xiong, Jingyi Xu, and Xieyuanli Chen

***Abstract*—LiDAR-based place recognition (LPR) is one of the most crucial components of autonomous vehicles to identify previously visited places in GPS-denied environments. Most existing LPR methods use mundane representations of the input point cloud without considering different views, which may not fully exploit the information from LiDAR sensors. In this article, we propose a cross-view transformer-based network, dubbed CVTNet, to fuse the range image views and bird's eye views generated from the LiDAR data. It extracts correlations within the views using intratransformers and between the two different views using intertransformers. Based on that, our proposed CVTNet generates a yaw-angle-invariant global descriptor for each laser scan end-to-end online and retrieves previously seen places by descriptor matching between the current query scan and the prebuilt database. We evaluate our approach on three datasets collected with different sensor setups and environmental conditions. The experimental results show that our method outperforms the state-of-the-art LPR methods with strong robustness to viewpoint changes and long-time spans. Furthermore, our approach has better real-time performance that can run faster than the typical LiDAR frame rate does.**

***Index Terms*—Autonomous driving, LiDAR place recognition (LPR), multiview fusion, transformer network.**

I. Introduction

PLACE recognition provides the current global location of the vehicle in the previously seen environments, which is an important component of robotic simultaneous localization and mapping and global localization. During online operation, it retrieves the reference scan in the database most similar to the current query by directly regressing the similarity [1], [2] or descriptor matching [3], [4], [5]. LiDAR-based place recognition (LPR) methods [6], [7], [8] can be applied to large-scale outdoor environments due to their robustness to illumination and weather changes. Different representation forms of LiDAR data have been exploited as input for LPR methods, such as 3-D point clouds [3], [5], voxels [9], [10], [11], normal distributions transform cells [12], [13], bird's eye views (BEVs) [7], [14], [15], and range image views (RIVs) [1], [6]. However, few methods fuse them together to exploit the advances of different views. Yin et al. [16] proposed FusionVLAD combining the different input forms, but they simply used fully connected layers to fuse features, thus not yet thoroughly digging out the inner relations between different representations.

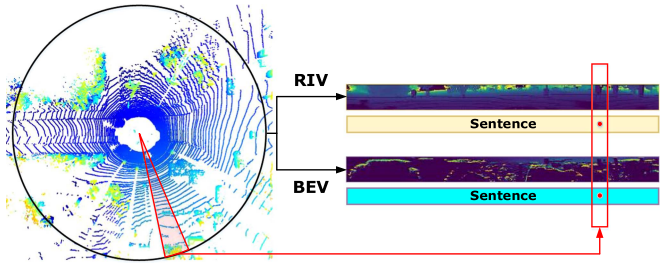

Fig. 1. Correspondence between the RIV and BEV leads to aligned feature sentences used by our proposed CVTNet.

To fill this gap, a novel cross-view transformer network named CVTNet is proposed for LPR in this work. It utilizes the combination of multilayer RIVs and BEVs generated from 3-D LiDARs mounted on autonomous vehicles. Exploiting the correspondence between the RIVs and BEVs of the same LiDAR scan as shown in Fig. 1, these two input views are first compressed into aligned sentence-like features. Then, our proposed CVTNet uses intra- and intertransformers to extract the correlation within and cross the aligned sentences from different views. Based on the proposed novel transformer network, our method generates a yaw-angle-invariant global descriptor for each place in an end-to-end manner and, in the end, achieves robust place recognition based on descriptor matching.

The main contribution of this article is an end-to-end transformer network that fuses RIVs and BEVs of LiDAR data to achieve reliable long-term place recognition. Unlike the existing LPR methods using mundane input forms, our method exploits both RIVs and BEVs of LiDAR data. Furthermore, we extend

Manuscript received 22 March 2023; revised 24 July 2023; accepted 31 August 2023. Date of publication 6 October 2023; date of current version 23 February 2024. This work was supported by the National Natural Science Foundation of China under Grant 52372404. Paper no. TII-23-0978. *(Corresponding author: Xieyuanli Chen.)*

Junyi Ma, Guangming Xiong, and Jingyi Xu are with the Beijing Institute of Technology, Beijing 100081, China (e-mail: 3120200365@bit.edu.cn; xiongguangming@bit.edu.cn; 3220200359@bit.edu.cn).

Xieyuanli Chen is with the National University of Defense Technology, Changsha 410073, China (e-mail: xieyuanli.chen@nudt.edu.cn).

Open-source code is available online at https://github.com/BITQ1MJY/CVTNet.

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TII.2023.3313635.

Digital Object Identifier 10.1109/TII.2023.3313635

the vanilla single-channel RIVs and BEVs to multilayer formats, which enables our transformer network to reweigh the LiDAR data from different ranges and heights, thus generating more reliable and distinctive features. To the best of the authors knowledge, this is the first work to dig out the latent alignment of RIV and BEV features from LiDAR point clouds and use the property to generate distinguishable representations for places. Our proposed CVTNet is also the first network exploiting cross-attention to fuse the multiview LiDAR data and generate yaw-angle-invariant global descriptors for LiDAR place recognition. Benefiting from the devised yaw-angle-invariant architecture, CVTNet is robust to viewpoint changes and achieves reliable place recognition even when the car drives in opposite directions, which is thoroughly evaluated on the NCLT, KITTI, and self-recorded datasets. The experimental results show that our proposed method outperforms the state-of-the-art methods in terms of both loop closure detection and place recognition. The ablation studies on the intra- and intertransformer modules and the validation experiments on yaw-angle-invariance are further provided.

In sum, we make three claims that our approach is able to

1) achieve state-of-the-art performance in place recognition and loop closure detection in outdoor large-scale environments using multiview LiDAR data,
2) recognize places well despite different viewpoints based on the proposed yaw-angle-invariant architecture, and
3) achieve online operation with a runtime less than 50 ms faster than the typical LiDAR frame rate. All claims are supported by our experimental evaluation on multiple datasets.

The rest of this article is organized as follows. Section II will investigate the related works. In Section III, the architecture of our proposed CVTNet and the related mathematical derivation will be described in detail. Section IV will present the experimental setups and results as well as corresponding analysis. Finally, Section V concludes this article.

II. Related Work

Place recognition has been widely investigated in robotics and computer vision [17], [18]. In this article, we only focus on LPR.

LPR is a classic research topic, and many traditional handcrafted methods have been proposed in [7], [14], [15], [19], and [20]. Recently, more and more learning-based methods have shown significant improvements with the development of neural network techniques. Parts of existing works use the raw point cloud or its voxelized and segmented forms as the input of the model. For example, Uy et al. [21] first combined the commonly used PointNet [22] with a differentiable VLAD architecture [18], providing the basic framework for the following methods to extract global descriptors from point clouds. Liu et al. [4] proposed LPD-Net, which utilizes a graph neural network to aggregate multiple local features extracted from point clouds. SegMap proposed by Dubé et al. [23] first extracts the segments from the point clouds as the input of the devised convolution neural network. LoGG3D-Net by Vidanapathirana et al. [5] exploits sparse convolution directly on raw point clouds, and considers both local consistency loss and scene-level loss. In contrast, MinkLoc3D by Komorowski [10] combined the sparse voxelized representation and convolutions to tackle the unordered set problem. More recently, LCDNet proposed by Cattaneo et al. [8] used an adapted PV-RCNN architecture to extract features that are used to generate global descriptors for point clouds and estimate the relative poses between the query and reference scan.

Although point cloud-based LPR methods have achieved reasonable results, most of them cannot operate online, thus hindering the deployment in real-world applications. To tackle this, many other works use image-like representations of LiDAR data, i.e., RIVs or BEVs, to improve the recognition efficiency and investigate more significant properties. For example, OverlapNet [1] utilizes the range images to directly regress out the overlap values and relative yaw angles between query and reference scans. Following the overlap conception, OverlapTransformer [6] takes advantage of the naturally yaw-angle-equivariance and builds a yaw-angle-invariant architecture. SphereVLAD++ by Zhao et al. [24] also uses RIVs to extract viewpoint-invariant global descriptors. Different from RIVs, BEVs lose some fine depth information but maintain stable height information, which is useful for long-term localization. For example, Scan Context [7] and Intensity Scan Context [14] both utilize the top–down view to encode each point cloud to a feature matrix. LiDAR-Iris [20] further extends the context-based methods to the frequency domain using Fourier transform. Inspired by LiDAR-Iris, BVMatch [15] also refers to the advantages of frequency domain, and extracts keypoints and BVFT descriptors for bag-of-words-based retrieval. CCL [25] proposes to use contrastive learning to extract more distinctive descriptor of places to achieve better LPR results.

All the abovementioned LPR methods only use one single input format of point clouds without combining different formats to further enhance the representative ability of the extracted features and descriptors. FusionVLAD by Yin et al. [16] is the first work that considers the inputs from different views of raw point clouds. However, it only applies fully connected networks to combine the extracted features, which cannot fully exploit different information from multiviews. Transformer [26] has recently been deployed to better extract features for place recognition [6], [9], [13]. However, there is no work successfully using transformers to fuse the features from different input forms or views. In this article, we propose a novel transformer network to extract the correlation within and between RIVs and BEVs. We explicitly align the RIV and BEV features from the same LiDAR data, thus better finding the relations between the sentence-like features from the dual views using cross-transformer and extracting more distinct global descriptors for more accurate place recognition.

III. Proposed Method

The overview of our proposed method is shown in Fig. 2. CVTNet consists of three modules, a multiview multilayer generator (MMG), an aligned feature encoder (AFE), and a multiview fusion module (MVF). MMG first splits the

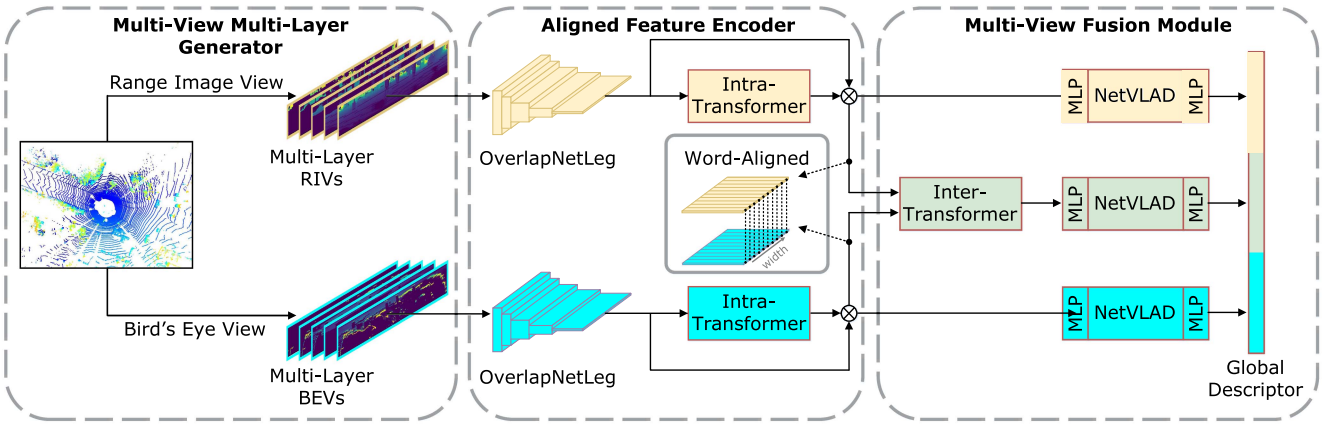

Fig. 2. Pipeline overview of our proposed CVTNet. It first generates multilayer RIV and BEV images for the two separate branches, and encodes the aligned sentence-like features, respectively, using the combination of OverlapNetLeg and the intratransformer module. Then, the intertransformer module is utilized to extract the inner correlation cross views. In the end, NetVLAD with MLP is exploited on three branches, respectively, to generate the final yaw-angle-invariant global descriptors for fast retrieval of place recognition.

region by range and height intervals, and conducts spherical and top–down projection, respectively, to generate multilayer input data (see Section III-A). The following AFE takes the multilayer RIVs and BEVs as inputs and uses a fully convolutional encoder to generate sentence-like features. It then applies an intratransformer to extract the inner correlation of the compressed features of each view (see Section III-B). Finally, the enhanced feature volumes are fed to the MVF module. It first uses an intertransformer to fuse the aligned features from different views and then applies NetVLAD with multilayer perceptrons (MLPs) on both single-view and fused features to generate the final 1-D global descriptor (see Section III-C). The final descriptor is invariant against yaw-angle rotation (see Section III-D). We train our network using the triplet loss with overlap labels to better distinguish the positive and negative examples (see Section III-E).

A. Multiview Multilayer Generator

It is important for LPR methods to extract discriminative features from different scans. Existing works obtain RIVs or BEVs using single spherical or top–down projection leads to data loss, thus degrading retrieving performance. To fully exploit all information from LiDAR data while maintaining fast speed, we propose to use multiple projections at different ranges/heights and form multilayer inputs based on space splits. By doing that, our proposed network learns to weigh information from different ranges/heights differently, thus extracting more representative features from scans.

More specifically, we propose the multiview multilayer generator. For RIVs in spherical coordinate systems, it first discretizes the region by preset range intervals $\{s_0, s_1, s_2, s_3, \ldots, s_q\}$, obtaining the split space $E^r = \{E_1^r, E_2^r, E_3^r, \ldots, E_q^r\}$. Then, it applies the spherical projection on laser points of the input point cloud $\mathcal{P}$ within different range intervals to generate multilayer range images. Similarly, for BEVs in the Euclidean coordinates, it discretizes space by height intervals $\{t_0, t_1, t_2, t_3, \ldots, t_q\}$ to get $E^b = \{E_1^b, E_2^b, E_3^b, \ldots, E_q^b\}$ and generates multilayer BEV images for all the intervals.

The correspondence between a LiDAR point $\boldsymbol{p}_k \in \mathcal{P}, \boldsymbol{p}_k = (x_k, y_k, z_k)$ and the pixel coordinates (u_k^r, v_k^r) in the corresponding RIV $\mathcal{R}_i$ can be represented as follows:

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

where $r_k = ||\boldsymbol{p}_k||_2 \in [s_{i-1}, s_i]$ is the range measurement of the corresponding point, $\mathrm{f} = \mathrm{f}_{\mathrm{up}} + \mathrm{f}_{\mathrm{down}}$ is the vertical field-of-view of the sensor, and w and h are the width and height of the resulting range image, respectively.

We project the same LiDAR points to BEVs by the top–down projection. The correspondence between the same LiDAR point $\boldsymbol{p}_k \in \mathcal{P}$ and the pixel coordinates (u_k^b, v_k^b) in the corresponding BEV image $\mathcal{B}_j$ can be represented as follows:

$$\begin{pmatrix} u_k^b \\ v_k^b \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\left[1 - \arctan(y_k, x_k)/\pi\right] w \\ \left[r_k'/\mathrm{f}'\right] h \end{pmatrix} \tag{2}$$

where $r_k' = ||(x_k, y_k)||_2 \in [t_{j-1}, t_j]$ and f' is the maximum sensing range.

In this work, we let the RIV and BEV images have the same width and height for better alignment, leading to $u_k^r = u_k^b$. This leads to the fact that the columns of the RIV and BEV images with the same index are both from the same spatial sector, as shown in Fig. 1. Finally, MMG generates the multilayer RIVs $\mathbb{R} = \{\mathcal{R}_i\}$ and multilayer BEVs $\mathbb{B} = \{\mathcal{B}_j\}$, as shown in Fig. 3.

B. Aligned Feature Encoder

To better fuse different representations of LiDAR data, our CVTNet utilizes an aligned feature encoder to extract coarse features of each representation and align them for later fusion. It first concatenates all RIVs $\mathbb{R}$ and BEVs $\mathbb{B}$ along the channel dimension obtaining $\mathcal{R}$ and $\mathcal{B}$ with the size of $q \times h \times w$. Then, it applies two OverlapNetLeg [1], [6] modules with the

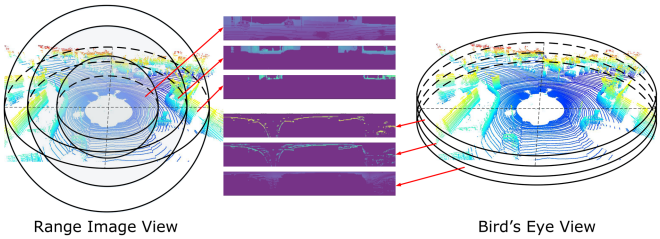

Fig. 3. Multilayer generation for RIVs (left-hand side) and BEVs (right-hand side).

same architecture to compress inputs into sentence-like features. The OverlapNetLeg is a full convolutional encoder only to compress RIVs and BEVs in the vertical dimension without changing the width dimension to avoid the discretization error to the yaw equivariance. The output coarse features from each branch are then enhanced by an intratransformer. We denote the enhanced features as $\mathcal{A}_0^r = \mathrm{AFE}_r(\mathcal{R})$ for the RIV branch, and $\mathcal{A}_0^b = \mathrm{AFE}_b(\mathcal{B})$ for the BEV branch, both with the size of $c \times 1 \times w$, where c refers to the feature channel number.

As proved in [6], the features from the RIV branch are yaw-angle-equivariant

$$\mathcal{A}_0^r C = \mathrm{AFE}_r\left(\Pi_r\left(R_\theta \mathcal{P}\right)\right) \tag{3}$$

where $\Pi_r(\cdot)$ is the spherical projection detailed in (1), C represents the column shift of the feature by matrix right multiplication, θ is the yaw angle change, and R_θ is the yaw rotation matrix of the point cloud $\mathcal{P}$.

In this article, we further show that our proposed BEV branch is also yaw-angle-equivariant. According to the top–down projection $\Pi_b(\cdot)$ detailed in (2), a yaw rotation of a point corresponds to a specific horizontal shift on BEV as follows:

$$\begin{pmatrix} u_k^{b\prime} \\ v_k^{b\prime} \end{pmatrix} = \begin{pmatrix} u_k^b + s \\ v_k^b \end{pmatrix}, \quad s = \frac{1}{2}\left[1 - \theta\pi^{-1}\right] w \tag{4}$$

$$\mathcal{B}C = \Pi_b\left(R_\theta \mathcal{P}\right) \tag{5}$$

where C and R_θ are the same column shift and yaw rotation.

As the OverlapNetLeg and intratransformer have proved to be yaw-angle-equivariant [6], for the BEV branch, we therefore also obtain the yaw-angle-equivariant features as follows:

$$\mathcal{A}_0^b C = \mathrm{AFE}_b\left(\Pi_b\left(R_\theta \mathcal{P}\right)\right). \tag{6}$$

It is crucial to align the features from different modalities before using the cross-transformer to fuse the different views and extract the cross-modal correlations [27], [28]. To this end, we first make sure the inputs of intertransformer from the two different views are spatially aligned. We let the input RIV and BEV images have the same width and height, and thus $u_k^r = u_k^b$ holds as clarified in Section III-A. The same laser point is projected to the same column of the two images, leading to one space sector corresponding to the same column index of the two images. To keep the alignments, we further design our network architecture only to compress the height dimension and maintain the information from one space sector aggregated to the same column of the two output features. We concatenate the output feature of the OverlapNetLeg and the feature enhanced by intratransformer along the channel dimension. Therefore, the output features $\mathcal{A}_0^r$ and $\mathcal{A}_0^b$ of AFE are devised to be word-aligned along the width dimension.

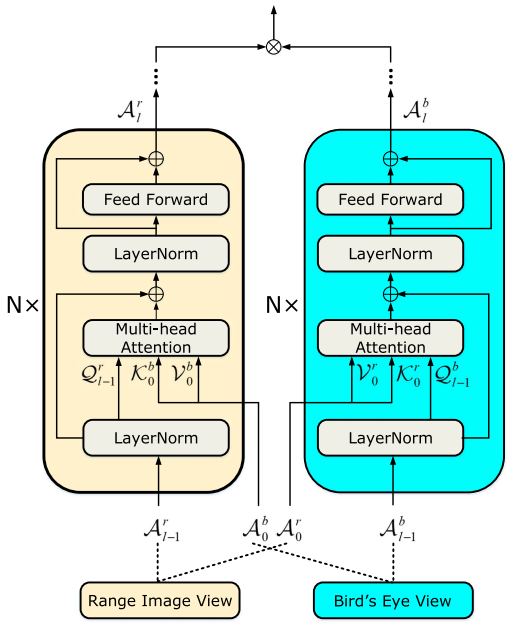

Fig. 4. Intertransformer module.

C. Multiview Fusion Module

Our CVTNet fuses the multiview features in the devised multiview fusion module. It consists of an intertransformer and three NetVLAD-MLPs combos [18] as shown in Fig. 2. It first uses the proposed intertransformer module to fuse the features from two branches of different views, as detailed in Fig. 4. In the intertransformer module, there are also two branches with l stacked transformer blocks to extract the correlation of different views by using the query feature from one view, and the key and value features from the other view.

For the RIV branch, we denote the feature volume extracted by the multihead selfattention (MHSA) of the lth transformer block as $\mathcal{A}_l^r$, and the cross-attention mechanism of the transformer can then be formulated as follows:

$$\bar{\mathcal{A}}_l^r = \mathrm{Attention}\left(\mathcal{Q}_{l-1}^r, \mathcal{K}_0^b, \mathcal{V}_0^b\right) = \mathrm{Softmax}\left(\frac{\mathcal{Q}_{l-1}^r \mathcal{K}_0^{bT}}{\sqrt{d_k}}\right)\mathcal{V}_0^b \tag{7}$$

where $\mathcal{Q}_{l-1}^r$ is substantially the query split of the layer-normalized $\mathcal{A}_{l-1}^r$ from the RIV, and $\mathcal{A}_0^r$ is the output feature from the AFE in the RIV branch. $\mathcal{K}_0^b, \mathcal{V}_0^b$ are the key and value splits of the layer-normalized $\mathcal{A}_0^b$, which is the output feature from the AFE in the BEV branch. d_k represents the dimension of splits. $\bar{\mathcal{A}}_l^r$ is then fed into the feed-forward network (FFN) and layer normalization (LN) to generate the attention-enhanced feature $\mathcal{A}_l^r$ of the lth transformer block, which can be formulated as

follows:

$$\mathcal{A}_l^r = \text{FFN}\left(\text{LN}\left(\bar{\mathcal{A}}_l^r + \text{LN}\left(\mathcal{A}_{l-1}^r\right)\right)\right) + \text{LN}\left(\bar{\mathcal{A}}_l^r + \text{LN}\left(\mathcal{A}_{l-1}^r\right)\right). \tag{8}$$

For the BEV branch, we conduct the same operations symmetrically as follows:

$$\bar{\mathcal{A}}_l^b = \text{Attention}\left(\mathcal{Q}_{l-1}^b, \mathcal{K}_0^r, \mathcal{V}_0^r\right) = \text{softmax}\left(\frac{\mathcal{Q}_{l-1}^b \mathcal{K}_0^{rT}}{\sqrt{d_k}}\right)\mathcal{V}_0^r \tag{9}$$

$$\mathcal{A}_l^b = \text{FFN}\left(\text{LN}\left(\bar{\mathcal{A}}_l^b + \text{LN}\left(\mathcal{A}_{l-1}^b\right)\right)\right) + \text{LN}\left(\bar{\mathcal{A}}_l^b + \text{LN}\left(\mathcal{A}_{l-1}^b\right)\right). \tag{10}$$

We then concatenate $\mathcal{A}_l^r$ with $\mathcal{A}_l^b$ from the last transformer block and obtain the intermediate fused feature $\mathcal{A}_l^f$.

In the end, our MVF uses the NetVLAD-MLPs combos on both the RIV and BEV features $\mathcal{A}_0^r$ and $\mathcal{A}_0^b$, as well as the cross-fused feature $\mathcal{A}_l^f$. The NetVLAD-MLPs has been widely used in LPR [4], [8], [13] to generate the global descriptor. We also exploit three NetVLAD-MLPs combos to compress the features from two intratransformers and one cross-view intertransformer into the global descriptors $[g^r, g^b, g^f]$.

D. Yaw-Angle Invariance

We further mathematically prove that the multiview fused descriptor generated by CVTNet is yaw-angle-invariant, which makes our method very robust to viewpoint changes. We have shown in Section III-B that the outputs of the aligned feature encoder are yaw-angle-equivariant in (3) and (6). As introduced in [3], NetVLAD is permutation invariant, and the column shift of its input caused by the yaw rotation of the raw point cloud can be regarded as the reorder of a set of 1-D vectors over the channel dimension [6]. Therefore, the outputs of NetVLAD in the RIV and BEV branches are not affected by the yaw rotation, and the global descriptors g^r and g^b are yaw-angle-invariant. In the rest, we prove that the fused cross-view feature g^f is also yaw-angle-invariant.

Note that $\mathcal{A}_0^r$ and $\mathcal{A}_0^b$ share the same shift C caused by one yaw rotation of one LiDAR data, which has been introduced in Section III-B. Therefore, for the first transformer block in the intertransformer after rotation, the (7) becomes

$$\begin{aligned}\widetilde{\mathcal{A}}_1^r &= \text{Attention}\left(C\mathcal{Q}_0^r, C\mathcal{K}_0^b, C\mathcal{V}_0^b\right) \\ &= \text{Softmax}\left(\frac{C\mathcal{Q}_0^r (C\mathcal{K}_0^b)^T}{\sqrt{d_k}}\right) C\mathcal{V}_0^b \\ &= C\,\text{sozftmax}\left(\frac{\mathcal{Q}_0^r \mathcal{K}_0^{bT}}{\sqrt{d_k}}\right)\mathcal{V}_0^b = C\bar{\mathcal{A}}_1^r. \end{aligned} \tag{11}$$

Similarly, for the following stacked cross-transformer blocks we also have the following:

$$\begin{aligned}\widetilde{\mathcal{A}}_l^r &= \text{Attention}\left(C\mathcal{Q}_{l-1}^r, C\mathcal{K}_0^b, C\mathcal{V}_0^b\right) \\ &= \text{Softmax}\left(\frac{C\mathcal{Q}_{l-1}^r (C\mathcal{K}_0^b)^T}{\sqrt{d_k}}\right) C\mathcal{V}_0^b \\ &= C\,\text{Softmax}\left(\frac{\mathcal{Q}_{l-1}^r \mathcal{K}_0^{bT}}{\sqrt{d_k}}\right)\mathcal{V}_0^b = C\bar{\mathcal{A}}_l^r \end{aligned} \tag{12}$$

which means the outputs of the MHSA of the cross-view transformer blocks are yaw-angle-equivariant. Note that in (11) and (12) the shift C is left-multiplied since the first dimension of the input feature is width dimension and the second one is embedding dimension. Because FFN and LN are also yaw-angle-equivariant towards features [6], and the last concatenation is operated on the channel dimension, the intertransformer outputs yaw-angle-equivariant cross-view feature volumes. Then, the NetVLAD with MLP is used to generate the yaw-angle-invariant global descriptor g^f. Since the three parts are all yaw-angle-invariant, our final global descriptor is yaw-angle-invariant.

E. Network Training

Following [1] and [6], we use more suitable overlap between pairs of scans to supervise the network rather than the ground truth distances. For each training tuple, we utilize one query descriptor $\mathcal{G}_a$, k_p positive descriptors $\{\mathcal{G}_p\}$, and k_n negative descriptors $\{\mathcal{G}_n\}$ to compute lazy triplet loss as follows:

$$\begin{aligned}&\mathcal{L}_T\left(\mathcal{G}_q, \{\mathcal{G}_p\}, \{\mathcal{G}_n\}\right) = \\ &k_p\left(\alpha + \max_p\left(d\left(\mathcal{G}_q, \mathcal{G}_p\right)\right)\right) - \sum_{k_n}\left(d\left(\mathcal{G}_q, \mathcal{G}_n\right)\right)\end{aligned} \tag{13}$$

where α is the margin to avoid negative loss and $d(\cdot)$ is the squared Euclidean distance. In our work, A pair of scans with overlap larger than 0.3 is regarded as a positive sample, otherwise a negative sample. In addition, we set $k_p = 6$, $k_n = 6$, and $\alpha = 0.5$ for the triplet loss. We use the triplet loss to make the distance between the query and all sampled negative global descriptors much larger than the distance between the query and the hardest positive global descriptors.

IV. Experimental Evaluation

The experimental evaluation is designed to showcase the performance of our approach and to support the claims that our approach is able to:

1) achieve state-of-the-art performance in place recognition and loop closure detection in outdoor large-scale environments using multiview LiDAR data,
2) recognize places well despite different viewpoints based on the proposed yaw-angle-invariant architecture, and
3) achieve fast operation with a runtime less than 50 ms.

A. Implementation and Experimental Setup

To evaluate localization performance of our proposed method, we utilize three datasets with different environmental conditions and sensor setups, including publicly available NCLT dataset [29] and KITTI odometry sequences [30], as well as our selfrecorded datasets in urban driving environments. Our selfrecorded dataset is collected by an AGV equipped with a LiDAR sensor (HESAI PandarXT, 32-beam), a wide-angle camera (SENSING SG2, HFOV 106°, VFOV 56°), an RTK

Fig. 5. Our self-recorded dataset collected by an autonomous vehicle in the urban environments.

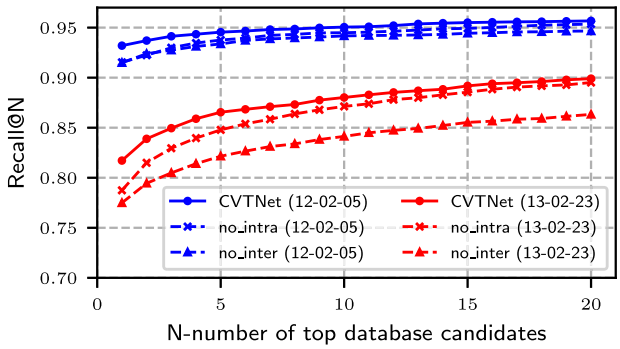

Fig. 6. Ablation study on intra- and intertransformer mudules.

GNSS (Asensing INS570D, 0.1° error in roll/pitch, 0.2° error in yaw, 2 cm + 1 ppm in RTK position), a mini computer (Intel Xeon E, 3.5 GHz, 80 W), and a Nivida Tesla T4 (16-GB memory, 70 W), as shown in Fig. 5 .

1) RIV Inputs: For RIV inputs of NCLT and selfrecorded datasets with 32-beam LiDAR data, we set range intervals as $\{[0\,\text{m}, 15\,\text{m}], [15\,\text{m}, 30\,\text{m}], [30\,\text{m}, 45\,\text{m}], [45\,\text{m}, 60\,\text{m}]\}$ to generate four-layer range images. We also generate the range image using all the laser points within 60 m, and concatenate it with four-layer range images to get the final RIV input with the size of $5 \times 32 \times 900$. For KITTI sequences, we set the range intervals as $\{[0\,\text{m}, 15\,\text{m}], [15\,\text{m}, 30\,\text{m}], [30\,\text{m}, 45\,\text{m}], [45\,\text{m}, 80\,\text{m}]\}$ to keep the same maximum range as existing works [1], [6] for fair comparisons.

2) BEV Inputs: For the BEV inputs of the NCLT dataset and selfrecorded dataset, we set height intervals to $\{[-4\,\text{m}, 0\,\text{m}], [0\,\text{m}, 4\,\text{m}], [4\,\text{m}, 8\,\text{m}], [8\,\text{m}, 12\,\text{m}]\}$ to generate four-layer BEV images, which are also concatenated with the BEV image projected from all the laser points within 60˜m. We set the height intervals of KITTI sequences to $\{[-3\,\text{m}, -1.5\,\text{m}], [-1.5\,\text{m}, 0\,\text{m}], [0\,\text{m}, 1.5\,\text{m}], [1.5\,\text{m}, 5\,\text{m}]\}$, adapted to its data distribution. The multilayer BEVs have the same size as the multilayer RIVs.

3) CVTNet Configuration: Our CVTNet uses the same OverlapNetLeg as OverlapTransformer [6]. For all the transformer modules, we set the embedding dimension $d_{\text{model}} = 256$, the number of heads $n_{\text{head}} = 4$, and the intermediate dimension of the feed-forward layer $d_{\text{ffn}} = 1024$. We use one selfattention transformer block in each intratransformer module following [6], and two cross-transformer blocks in the intertransformer module, which is validated as the best configuration in the following ablation study. For all the NetVLAD modules, we set the intermediate feature dimension $d_{\text{inter}} = 1024$, the output feature dimension $d_{\text{output}} = 256$, and the number of clusters $d_K = 64$. The output descriptor of CVTNet is a vector of size 1×768. For evaluation conducted on each dataset, our proposed CVTNet is trained for 30 epochs, using the ADAM optimizer to update the weights of the network with an initial learning rate of 5e-6 and a weight decay of 0.9 applied every five steps. Other related training configuration has been clarified in Section III-E. The test setups follow the previous work [31], which will also be introduced in the following sections. We will release the implementation of CVTNet as open source to provide more network details and evaluation configurations.

4) Baseline Configuration: In the following experiments, we compare our proposed CVTNet with both state-of-the-art methods PointNetVLAD [21] and Scan context [7] without shift matching for real-time retrieval, MinkLoc3D [10], OverlapTransformer [6], and the other multiview LPR method FusionVLAD [16]. The descriptors generated by all the abovementioned learning-based baseline methods are with the size of 1×256 except for FusionVLAD which generates the fused 1×2048 descriptor, all following their original configurations. The feature matrix generated by Scan Context is set to 20×60. Note that we reimplement FusionVLAD according to its original paper since it is not open source, and the other parameter configurations of all the baseline methods in the experiments also follow their original papers.

B. Evaluation for Place Recognition

The first experiment supports our claim that our approach achieves state-of-the-art place recognition and loop closure detection in outdoor large-scale environments using multiview LiDAR data.

Following [6], we use the NCLT dataset for evaluating long-term place recognition, where sequence 2012-01-08 is used to train the models and sequences 2012-02-05, 2012-06-15, 2013-02-23, and 2013-04-05 are used to evaluate the recognition performance. Besides NCLT dataset, we also recorded our own dataset with multiple reverse driving conditions repeated in the same environment, which are frequent cases in autonomous driving but challenging to place recognition methods. We use sequence-1 as the database for training and two other runs (sequence-2 and sequence-3) as query sequences for evaluation. They all share the same route but driving from different directions. In contrast, KITTI odometry sequences are used for evaluating loop closure detection, where we use sequences 03–10 for training, sequence 02 for validation, and sequence 00 for evaluation.

We use average top 1 recall (AR@1), top 5 recall (AR@5), and top 20 recall (AR@20) as the metrics of the evaluation conducted on NCLT and the self-recorded datasets. We also use AUC and F1 max scores as the evaluation metrics for the

TABLE I
COMPARISON OF RECOGNITION PERFORMANCE ON THE NCLT DATASET

Approach	2012-02-05			2012-06-15			2013-02-23			2013-04-05		
	AR@1	AR@5	AR@20	AR@1	AR@5	AR@20	AR@1	AR@5	AR@20	AR@1	AR@5	AR@20
Scan context [7]	0.767	0.836	0.909	0.581	0.637	0.724	0.481	0.564	0.726	0.418	0.496	0.649
PointNetVLAD [21]	0.746	0.823	0.875	0.612	0.720	0.782	0.469	0.604	0.719	0.449	0.576	0.683
MinkLoc3D [10]	0.802	0.864	0.926	0.630	0.685	0.774	0.507	0.616	0.751	0.482	0.587	0.685
OverlapTransformer [6]	0.861	0.899	0.930	0.639	0.697	0.780	0.536	0.645	0.764	0.496	0.603	0.715
FusionVLAD [16]	0.786	0.870	0.922	0.462	0.579	0.702	0.510	0.643	0.754	0.429	0.553	0.667
CVTNet (ours)	**0.932**	**0.946**	**0.957**	**0.793**	**0.835**	**0.871**	**0.817**	**0.866**	**0.899**	**0.780**	**0.826**	**0.869**

TABLE II
COMPARISON OF RECOGNITION PERFORMANCE ON THE KITTI AND SELF-RECORDED DATASETS

Approach	Sequence-2			Sequence-3 (reverse)			KITTI	
	AR@1	AR@5	AR@20	Recall@1	AR@5	AR@20	AUC	F1max
Scan Context [7]	0.877	0.958	0.986	0.564	0.710	0.810	0.836	0.835
PointNetVLAD [21]	0.913	0.936	0.957	0.400	0.521	0.623	0.856	0.846
MinkLoc3D [10]	0.948	0.971	0.993	0.450	0.653	0.808	0.894	0.869
OverlapTransformer [6]	0.974	0.987	0.995	0.776	0.878	0.931	0.907	0.877
FusionVLAD [16]	0.971	0.990	0.997	0.865	0.923	0.960	0.882	0.860
CVTNet (ours)	**0.992**	**0.996**	**0.998**	**0.907**	**0.951**	**0.993**	**0.911**	**0.880**

KITTI dataset as introduced in [6]. The evaluation results are given in Tables I and II where bold values correspond to the best performance. As can be seen, our proposed CVTNet has better performance than all the state-of-the-art baseline methods on the place recognition metrics. Compared with OverlapTransformer, CVTNet utilizes the combination of RIVs and BEVs as input rather than only range images, which increases the performance of recognizing places with large time gaps by 7.1%–28.4% on AR@1 of the NCLT dataset. In contrast to FusionVLAD, which also considers different views, our proposed CVTNet outperforms it by 14.6%–35.1% on AR@1 of the NCLT dataset, and 2.1%–4.2% on AR@1 of the self-recorded dataset. For loop closure detection on the KITTI dataset, our method also significantly outperforms all baseline methods.

C. Ablation Study on Transformer Modules

This ablation study investigates the effect of intra- and inter-transformer modules. We compare our CVTNet with the variants of not using transformer modules. We evaluate all variants using the average recall AR@N on sequence 2012-02-05 and 2013-02-23 of the NCLT dataset. The results are shown in Fig. 6. As can be seen, our CVTNet with both intra- and intertransformer modules consistently outperforms all variants. Besides, CVTNet with only an intertransformer outperforms the one using only an intratransformer, which indicates that the intertransformer contributes more to enhancing the discrimination of the final global descriptor than the intratransformer. The benefit of using two types of transformers is more prominent in place recognition with a larger time gap, which can be drawn by comparing the results between two sequences recorded from different times.

D. Ablation Study on Cross-Transformer Blocks

This ablation study aims to show the effectiveness of the number of cross-transformer blocks in the intertransformer module of CVTNet. Note that we only compare our method with 1–3 cross-transformer blocks to maintain low memory consumption. The experimental results on the NCLT dataset are given in Table III. As can be seen, the intertransformer with two cross-transformer blocks outperforms the ones with one and three cross-transformer blocks. The reason could be that more transformer blocks might need more training data and time to obtain good performance. We also show the average runtime to generate one global descriptor for sequence 2012-01-08 of the NCLT dataset in Table III. It can be seen that more cross-transformer blocks cost more time to extract features. Thus, we use two transformer blocks in our CVTNet to keep effectiveness and efficiency.

E. Study on Multilayer Input

In this experiment, we use selfrecorded datasets to validate the superiority of using multilayer input for CVTNet. We modify the input channel number of the first layer of OverlapNetLeg to receive the single-channel input. The experimental results are given in Table IV. As can be seen, CVTNet with multilayer input outperforms the baseline with only single-channel input. Our proposed multilayer input explicitly tackles the problem of occluded laser points in the background, providing more input point cloud information, thus enabling the network to reweight data from different subspaces to enhance the discrimination of the extracted features.

To further validate the effectiveness of the proposed multilayer form of inputs, we also apply it to another network, FusionVLAD as also given in Table IV. The multilayer input form also improves the performance of FusionVLAD, which indicates that our proposed multilayer input form generalizes well into different methods. Still, our method significantly outperforms the FusionVLAD with multilayer inputs due to the devised novel intra- and intertransformer network.

TABLE III
COMPARISON OF CVTNET WITH DIFFERENT NUMBERS OF CROSS-TRANSFORMER BLOCKS ON THE NCLT DATASET

Number	Runtime [ms]	2012-02-05			2012-06-15			2013-02-23			2013-04-05		
		AR@1	AR@5	AR@20	AR@1	AR@5	AR@20	AR@1	AR@5	AR@20	AR@1	AR@5	AR@20
1	5.46	0.924	0.943	0.955	0.781	0.826	0.860	0.798	0.843	0.893	0.749	0.797	0.840
2	6.49	**0.932**	**0.946**	**0.957**	**0.793**	**0.835**	**0.871**	**0.817**	**0.866**	**0.899**	**0.780**	**0.826**	**0.869**
3	7.32	0.923	0.940	0.955	0.783	0.824	0.861	0.787	0.840	0.889	0.757	0.807	0.856

TABLE IV
STUDY ON MULTILAYER INPUT

Approach	AR@1	AR@5	AR@20
FusionVLAD (single-channel)	0.865	0.923	0.960
FusionVLAD (multilayer)	0.868	0.930	0.970
CVTNet (single-channel)	0.896	0.948	0.987
CVTNet (multilayer)	**0.907**	**0.951**	**0.993**

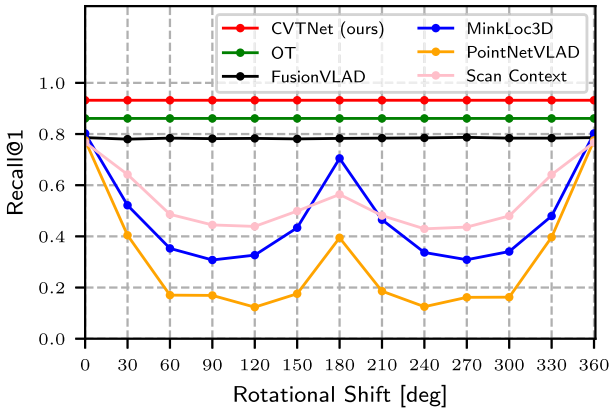

Fig. 7. Study on yaw-angle invariance.

F. Study on Yaw-Angle Invariance

We have shown the robustness of our proposed CVTNet to viewpoint changes using the self-recorded dataset in Section IV-B. In this experiment, we further design a more straightforward way to validate that our devised network architecture is strictly invariant to the yaw rotation of LiDAR scans obtained at the same place. We rotate each query scan of sequence 2012-02-05 along the yaw-axis in steps of 30° and search the same places in the database built from sequence 2012-01-08. We compare our method with the state-of-the-art baseline methods and use Recall@1 as the criterion to show the effect of yaw-angle changes on place recognition performance. The experimental results are shown in Fig. 7. As can be seen, CVTNet outperforms the other methods consistently in terms of Recall@1 in all the rotation conditions. Both CVTNet and OverlapTransformer are not affected by the rotation along the yaw-axis due to the devised yaw-rotation-invariant architectures. FusionVLAD is not significantly affected by yaw-angle rotation but is not strict yaw-rotation-invariant. In contrast, the performance of MinkLoc3D and PointNetVLAD decreases fast with the yaw angle increasing, which means their network architectures are not robust enough to viewpoint changes.

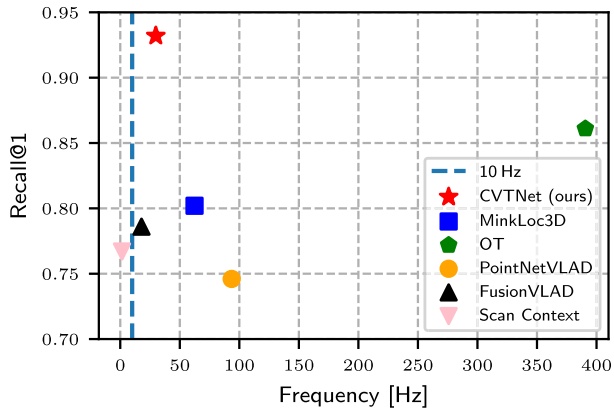

Fig. 8. Comparison of runtime with state-of-the-art methods.

G. Runtime and Memory Consumption

We further compare the real-time performance of our CVTNet with all the baseline methods, and evaluate its memory consumption. We conduct the experiments on a system with an Intel i7-11700 K CPU and an Nvidia RTX 3070 GPU. We use NCLT sequence 2012-01-08 as database and sequence 2012-02-05 as query to calculate the total runtime of global descriptor generation and top 20 candidates retrieval within 28 127 database laser scans for each query laser scan. FAISS library [32] is used here to accelerate the searching process except for Scan Context, which can only use an ad-hoc retrieval function for shift-based matching [7]. Fig. 8 shows the running frequency versus top 1 recall. The running frequency of CVTNet is 29.85 Hz (33.50 ms in total, 15.03 ms for descriptor generation, and 18.47 ms for retrieval), which is about three times faster than a typical LiDAR sensor frame rate at 10 Hz, while achieving the state-of-the-art place recognition performance. In addition, the number of parameters of CVTNet is 21.38 M, showing the small memory consumption of our approach.

V. CONCLUSION

In this article, a novel cross-view transformer network has been presented for LPR. Point clouds have been projected to multilayer RIVs and BEVs as the inputs of our method instead of the mundane single-channel form. The intratransformer has been utilized in the proposed network to capture the inner correlation within the individual view, while the intertransformer has been used to extract the cross-view correlation between different views. The experimental comparisons of our method with the state-of-the-art methods have validated its superior performance

in both place recognition and loop closure detection. Moreover, the experimental results have also shown that our proposed CVTNet is yaw-angle-invariant against viewpoint changes and operates online, which can be used for real autonomous driving applications.

REFERENCES

[1] X. Chen, T. Läbe, A. Milioto, T. Röhling, J. Behley, and C. Stachniss, "OverlapNet: A. Siamese network for computing LiDAR scan similarity with applications to loop closing and localization," *Auton. Robots*, vol. 46, pp. 61–81, 2021. [Online]. Available: http://www.ipb.uni-bonn.de/pdfs/chen2021auro.pdf

[2] J. Ma, S. Wang, K. Zhang, Z. He, J. Huang, and X. Mei, "Fast and robust loop-closure detection via convolutional auto-encoder and motion consensus," *IEEE Trans. Ind. Informat.*, vol. 18, no. 6, pp. 3681–3691, Jan. 2022.

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

[4] Z. Liu et al., "LPD-Net: 3D point cloud learning for large-scale place recognition and environment analysis," in *Proc. IEEE/CVF Intl. Conf. Comput. Vis.*, 2019, pp. 2831–2840.

[5] K. Vidanapathirana, M. Ramezani, P. Moghadam, S. Sridharan, and C. Fookes, "LOGG3D-Net: Locally guided global descriptor learning for 3D place recognition," in *Proc. IEEE Intl. Conf. Robot. Automat.*, 2022, pp. 2215–2221.

[6] 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.

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

[8] D. Cattaneo, M. Vaghi, and A. Valada, "LCDNet: Deep loop closure detection and point cloud registration for LiDAR SLAM," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2074–2093, Aug. 2022.

[9] Z. Fan, Z. Song, H. Liu, Z. Lu, J. He, and X. Du, "SVT-Net: Super light-weight sparse voxel transformer for large scale place recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 551–560.

[10] J. Komorowski, "MinLoc3D: Point cloud based large-scale place recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2021, pp. 1790–1799.

[11] M. Y. Chang, S. Yeon, S. Ryu, and D. Lee, "SpoxelNet: Spherical voxel-based deep place recognition for 3D point clouds of crowded indoor spaces," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 8564–8570.

[12] A. Zaganidis, A. Zerntev, T. Duckett, and G. Cielniak, "Semantically assisted loop closure in SLAM using NDT histograms," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4562–4568.

[13] Z. Zhou et al., "NDT-transformer: Large-scale 3D point cloud localisation using the normal distribution transform representation," in *Proc. IEEE Intl. Conf. Robot. Automat.*, 2021, pp. 5654–5660.

[14] H. Wang, C. Wang, and L. Xie, "Intensity scan context: Coding intensity and geometry relations for loop closure detection," in *Proc. IEEE Intl. Conf. Robot. Automat.*, 2020, pp. 2095–2101.

[15] L. Luo, S.-Y. Cao, B. Han, H.-L. Shen, and J. Li, "BVmatch: LPR using bird's-eye view images," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 6076–6083, Jul. 2021.

[16] P. Yin, L. Xu, J. Zhang, and H. Choset, "FusionVLAD: A multi-view deep fusion networks for viewpoint-free 3D place recognition," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2304–2310, Apr. 2021.

[17] X. Ye and J. Ma, "Neighborhood manifold preserving matching for visual place recognition," *IEEE Trans. Ind. Informat.*, vol. 19, no. 7, pp. 8127–8136, Jul. 2023.

[18] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5297–5307.

[19] L. Li, M. Yang, C. Wang, and B. Wang, "Hybrid filtering framework based robust localization for industrial vehicles," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 941–950, Mar. 2018.

[20] Y. Wang, Z. Sun, C-Z. Xu, S. E. Sarma, J. Yang, and H. Kong, "LiDAR Iris for loop-closure detection," in *Proc. IEEE/RSJ Intl. Conf. Intell. Robots Syst.*, 2020, pp. 5769–5775.

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

[22] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.

[23] R. Dubé et al., "SegMap: Segment-based mapping and localization using data-driven descriptors," *Int. J. Robot. Res.*, vol. 39, no. 2–3, pp. 339–355, 2020.

[24] P. Yin, F. Wang, A. Egorov, J. Hou, J. Hang, and H. Choset, "SeqSphereVLAD: Sequence matching enhanced orientation-invariant place recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5024–5029.

[25] J. Cui and X. Chen, "CCL: Continual contrastive learning for LiDAR place recognition," *IEEE Robot. Automat. Lett.*, vol. 8, no. 8, pp. 4433–4440, Aug. 2023.

[26] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[27] A. Zadeh, P. P. Liang, N. Mazumder, S. Poria, E. Cambria, and L.-P. Morency, "Memory fusion network for multi-view sequential learning," in *Proc. Conf. Advancements Artif. Intell.*, 2018, pp. 5634–5641.

[28] H. Pham, P.P. Liang, T. Manzini, L-P. Morency, and B. Póczos, "Found in translation: Learning robust joint representations by cyclic translations between modalities," in *Proc. Conf. Advancements Artif. Intell.*, 2019, pp. 6892–6899.

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

[30] 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. [Online]. Available: http://www.cvlibs.net/publications/Geiger2012CVPR.pdf

[31] 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.

[32] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Trans. Big Data*, vol. 7, no. 3, pp. 535–547, Jul. 2021.

Junyi Ma received the B.E. degree in vehicle engineering in 2020 from the Beijing Institute of Technology, Beijing, China, where he is currently working toward the master's degree in vehicle engineering.

He is trying to apply machine learning methods to robotics and use multiple sensor data for enhanced perception capability of robots and intelligent vehicles. His research interests include simultaneous localization and mapping, place recognition, and robotics.

Guangming Xiong received the Ph.D. degree in mechanical engineering from the Beijing Institute of Technology, Beijing, China, in 2005.

He is currently an Associate Professor with the School of Mechanical Engineering, Beijing Institute of Technology. His research interests include perception and SLAM for intelligent vehicles.

Jingyi Xu received the B.E. degree in vehicle engineering in 2020 from the Beijing Institute of Technology, Beijing, China, where she is currently working toward the master's degree in vehicle engineering.

Her research interests include energy management strategies for intelligent vehicles, semantic segmentation, and motion planning of robots.

Xieyuanli Chen received the bachelor's degree in electrical engineering and automation from Hunan University, Changsha, China, in 2015, the master's degree in robotics from the National University of Defense Technology, Changsha, in 2017, and the Ph.D. degree in robotics from Photogrammetry and Robotics Laboratory, University of Bonn, Bonn, Germany, in 2022.

He is currently an Associate Professor with the National University of Defense Technology.

Dr. Chen is currently an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS, ICRA, and IROS.