

Cross Fusion of Point Cloud and Learned Image for Loop Closure Detection

Haosong Yue[✉], Danyang Cao[✉], Zhong Liu[✉], Tian Wang[✉], *Senior Member, IEEE*,
and Weihai Chen[✉], *Member, IEEE*

Abstract—Loop closure detection (LCD) plays a crucial role in simultaneous localization and mapping (SLAM) systems to eliminate accumulated odometry drifts as the map is built, and using multi-modal information can improve the accuracy and robustness of this system compared to single sensor. However, traditional fusion methods often require sophisticated space alignment of the sensors, which places high requirements on hardware equipment. Moreover, these methods fuse the point cloud and image in a weak manner, such as concentrating two kinds of features using calibration information, which makes they cannot take full use of the multi-modal information. In this letter, we propose a method for fusing asymmetric point clouds and images to detect loops. Bird's Eye View (BEV) of point cloud and image achieve information interaction and associations through learnable modules, rather than through hard intrinsic and extrinsic matrices to perform cross-fusion. Multi-information is fused in BEV grids, then the features of BEV key points are enhanced. Through experiments, it is found that under the training of this fusion strategy, the image information can be encoded into the BEV processing modules, and ultimately the performance of the method can be improved without images. The proposed method is evaluated on KITTI and KITTI-360, and the results demonstrate the state-of-the-art performance and remarkable efficiency.

Index Terms—Bird's eye view, cross fusion, information interaction, lightweight, loop closure detection.

I. INTRODUCTION

ACCURATE perception of complex environments has always been a critical issue for many applications, such as unmanned vehicles and autonomous robots. Simultaneous localization and mapping (SLAM) is the pivotal solution for handling this problem. Loop closure detection (LCD) can provide additional spatial constraints to SLAM systems and improve the consistency of the maps [1]. Detailed environment

perception requires information collected by multiple sensors and most of them are cameras and LiDARs. LiDAR can provide accurate 3D geometric structure and has advantages in panoramic environment perception. But point cloud is sparse, and it is difficult to perceive less structured or transparent objects. Cameras obtain 2D image by capturing the projection of light on the image plane, which can provide rich texture and color information [2], and has more advantages in detail description. But the image data is easily affected by factors such as viewpoint and illumination changes. Recently, there has been a surge in popularity for approaches that leverage the complementary nature of these two modalities [3], [4], leading to improved performance.

However, how to effectively and reasonably combine the two kinds of information is still a very difficult problem. At present, many methods use independent backbone networks to extract features when using two kinds of information. Then fusion modules are used to combine the two features into one [5]. Finally the fusion features are input into subsequent task processing modules, such as semantic segmentation and object detection [4]. These methods require good symmetry between input images and point clouds, placing higher demands on sensor equipment. Besides, they only establish a connection between the fused information and the inputs, without leveraging the deep relationships between them.

In this letter, we propose a novel feature fusion method suitable for LCD based on asymmetric point clouds and images. The converter and generator modules are designed to mine the deep relationship between point cloud features and image features, so that they can be closely combined and realize bi-directional flow of information. Since the proposed approach primarily relies on Bird's Eye View (BEV) information, with image information as a supplementary, the network during training will also tend to favor BEV. In the end, it can operate without depending on images entirely but still learn some valuable image-related insights, thereby enhancing BEV features and improving LCD performance.

The main contributions of this work are summarized as follows:

- 1) A cross-fusion strategy is designed to fully interact between point clouds and images, which enables bi-directional flow of information during fusion.
- 2) Using relationships between features makes it possible to handle asymmetric point clouds and images, and can get rid of alignment when testing.

Manuscript received 9 October 2023; accepted 30 January 2024. Date of publication 6 February 2024; date of current version 16 February 2024. This letter was recommended for publication by Associate Editor H. Araujo and Editor P. Vasseur upon evaluation of the reviewers' comments. (*Corresponding author: Haosong Yue.*)

Haosong Yue, Danyang Cao, and Zhong Liu are with the School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China (e-mail: yuehaosong@buaa.edu.cn; caodanyang@buaa.edu.cn; liuzhong@buaa.edu.cn).

Tian Wang is with the Institute of Artificial Intelligence, Beihang University, Beijing 100191, China (e-mail: wangtian@buaa.edu.cn).

Weihai Chen is with the School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China, and also with the Hangzhou Innovation Institute, Beihang University, Beijing 100191, China (e-mail: whchen@buaa.edu.cn).

We release our codes on <https://github.com/caodanyang/FUSIONLCD>.
Digital Object Identifier 10.1109/LRA.2024.3362681

- 3) Attention mechanism is utilized to mine the relations of the features and combine information of the two sensors.
- 4) Experiments on two commonly used datasets show that our approach has better performance to state-of-the-art methods and can run in real time.

II. RELATED WORK

In this section, we provide an overview of the state-of-the-art methods for LCD and fusion methods of point cloud and image.

A. Loop Closure Detection

Feature extraction can be regarded as quantitative description of the scene and is the basis and core part of LCD. Local features have better accuracy and robustness than global features and raw data, so they are often applied in LCD. In the feature extraction of image, handcrafted methods usually achieve the representations with rotation and scale invariance, such as SIFT (Scale-Invariant Feature Transform) [6], SURF (Speeded Up Robust Features) [7], and ORB (Oriented FAST and Rotated BRIEF) [8], while the methods based on deep learning usually possess stronger representational capabilities, such as SuperPoint [9] and ALIKE [10]. The commonality among these methods is that they explicitly obtain key points and their local features. In the feature extraction of point cloud, due to its data being unordered and discrete, there are different types of methods for the processing. One is the method based on the original point cloud, such as normal distribution transform (NDT) [11], point feature histograms (PFH) [12], linear key points representation for 3D LiDAR (Link3D) [13]. Most of them portray the spatial relationships or the geometric properties of the local areas, while Link3D describes the structural distribution information of key points. Deep learning also promotes the feature extraction of point cloud, such as PointNetVLAD [14], LCDNet [15], and PADLoC [16]. Another set of methods involves the utilization of 2D pseudo images derived from point cloud. Within this category, conventional approaches include Scan context [17], LiDAR Iris [18], and M2DP [19]. Furthermore, contemporary deep learning-based methods consist of OverlapNet [20], OverlapTransformer [21], etc. Driven by large-scale open-source datasets, methods based on deep learning achieve even more remarkable results.

LCD aims to achieve accurate and efficient detection. However, large-scale scenes in reality have numerous local features. Therefore, efficient loop retrieval cannot be achieved only by using local features. Due to the excellent performance of BoW [22] and NetVLAD [23], they have been the most commonly used methods to aggregate local features by describing the same scene into a single global descriptor. In order to improve the accuracy of LCD, many methods pick up the spatial structure of the scene lost during feature extraction and aggregation to construct additional globally consistent constraints [1] to verify loop candidates. For LCD methods based on images, graph model [24], [25] and local motion [26], [27] are the popular ways to get the consistency for key points. On the other hand, point cloud-based methods [28] mostly achieve quantitative comparison of global consistency by estimating the rigid transformation matrix.

B. Fusion of Point Cloud and Image

The fusion can capitalize on the strengths of both modalities, harnessing the detailed geometry captured by point cloud and the rich visual information inherent in images. Since the domains of point cloud and image are completely different, how to associate these two modalities becomes the key of fusion. Using homogeneous transformation to match image information and point cloud information is the basis of many fusion methods. PointPainting [29] concatenate 3D points with semantic segmentation scores from an image-based semantic segmentation network and feed the painted point cloud to the 3D detection network. PointAugmenting [30] uses LiDAR points appended by the fetched pointwise image features as network inputs to perform 3D detection. BEVFusion [4] associate the camera features with BEV grids, and then pool the features within each BEV grid with symmetric function. [31], [32] use projection to associate camera features and LiDAR features and then perform the follow-up tasks with the fusion features. Recently, attention [33] has been widely used, it can capture temporal and spatial dependencies, and has become an effective method for multi-modal fusion, such as [5], [34], [35]. Projecting point cloud to depth and use RGB-D [36] to extract the learning-based high-dimensional global feature and histogram-based local feature.

III. METHOD

Fig. 1 presents the overview of the proposed method, which includes three main modules: feature extraction for each of the two modalities and the modules for fusion, and finally the loop retrieval and verification.

A. Feature Extraction

Similar to our previous work BEVLCD [37], point clouds are projected to BEV for processing efficiency. Another benefit of projecting point clouds is that we can use the same CNN backbone architecture for feature extraction of point clouds and images. The CNN backbone is from ALIKE [10], which is lightweight and efficient. As shown in Fig. 2, there are four blocks primarily comprised of convolutional layers for extracting features at different levels. Max-pooling is utilized to enlarge the receptive field, enhancing feature noise resistance. Upsampling with 1×1 convolutional layer is performed to adjust the size of the feature map to be consistent with the input size, alongside channel adjustments to facilitate the concatenation of multi-level features. Finally, a 1×1 convolutional layer is employed to estimate feature map and score map, and the scores are then normalized to 0-1 with *sigmoid*. Based on the BEV score map, Non-Maximum Suppression (NMS) is applied to retain the most significant points. Subsequently, we select the *topk* grids with the highest scores as key points. These grids are then used to sample both the key points' coordinates and the corresponding local features.

In order to integrate the geometric information lost during feature extraction with the CNN backbone, we encode the geometric position distribution of key points. Recognizing the robust rotation and translation invariance of Euclidean distance, we

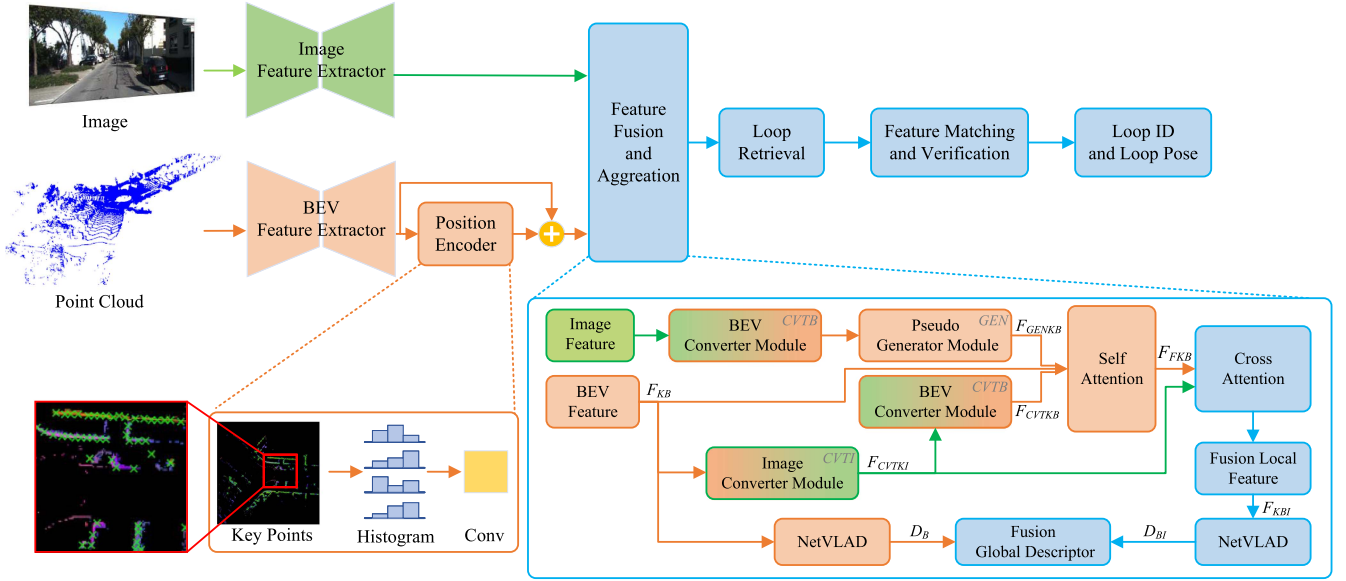


Fig. 1. Overview of the proposed method. Green, orange, and blue represent the information processing and transmission of image, point cloud, and fusion of them, respectively.

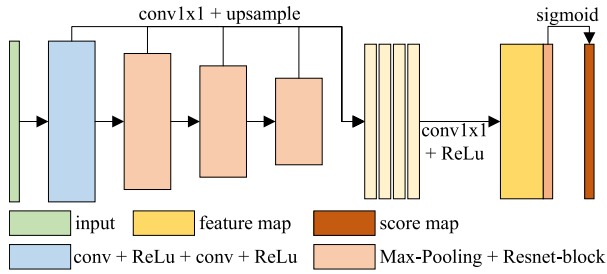


Fig. 2. CNN backbone used to extract features. The feature map is used for BEV and image while score map is only used for BEV.

conduct histogram-based statistical analysis on the distances between key points using a fixed criterion. For each histogram, 1×1 convolutional layer is employed to adopt the channels and add them with features originating from the CNN backbone. Given that distance calculations are centered on key points within BEV space, their relatively limited quantity ensures that such positional encoding has a negligible impact on computational efficiency.

B. Fusion Module

Most of the traditional fusion methods of point cloud and image focus on how to effectively combine these two modalities [29], [30]. They requires the view range of point cloud and image to be as consistent as possible, but not all applications can meet this condition [38], [39]. In this letter, we design a solution to the fusion problem of point cloud and image when the range of view is inconsistent.

Cropping and padding are performed on the image to meet downsampling requirements. Additionally, to ensure alignment between the cropped and padded images and point clouds, we also record the pixel offsets $\Delta w, \Delta h$ of cropping and padding. For a point (p_x, p_y, p_z) , with the intrinsic K and extrinsic R, t ,

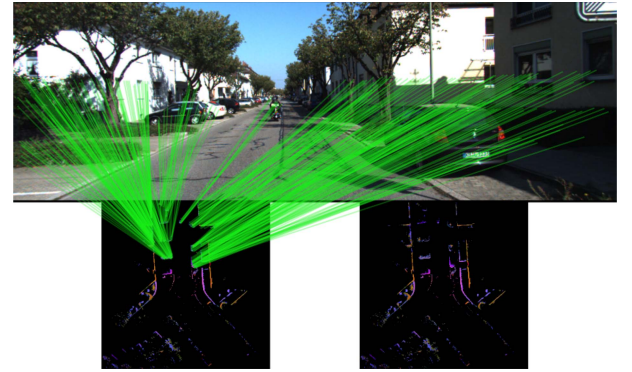


Fig. 3. Connections of BEV grids and image pixels. The bottom right figure is the same BEV as the bottom left for better readability.

it is possible to obtain its corresponding pixel in the image:

$$s \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \cdot [R, t] \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} + s \cdot \begin{bmatrix} \Delta w \\ \Delta h \\ 1 \end{bmatrix} \quad (1)$$

where u, v represent image pixels and s denotes the depth normalization factor.

As shown in Fig. 3, the results obtained by projecting 3D points onto the BEV grids and image allow us to establish associations between the BEV and image. Due to differences in range of view, only a partial range of the BEV can be connected with the image, and the connections that exist are also unbalanced. The image features corresponding to BEV grids are multiple in number and require to be aggregated. Firstly, we apply a convolutional layer to linearly combine these image features. Subsequently, we employ max-pooling to extract one single image feature for each BEV grid.

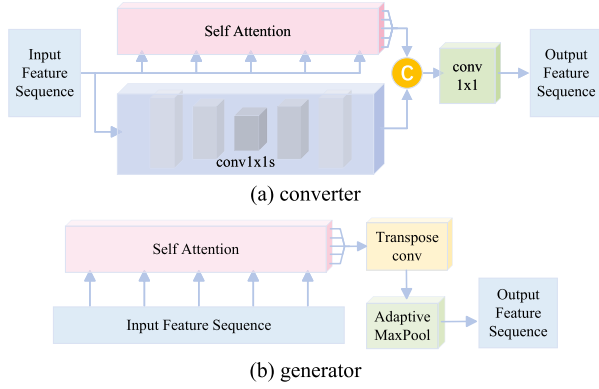


Fig. 4. Two basis modules of our fusion method. Attention focuses on mining the mutual relationships between features, convolutional layers concentrate on the internal connections within features, and transpose convolutions ensure that adaptive pooling has sufficient input information.

The two modules shown in Fig. 4 capture the profound interrelation of point cloud and image, which is the basis for the enhanced features. The main component of each of these modules is attention [33]. It is particularly suitable for capture relations within sequences. The general form of the attention-mechanism can be expressed as the following formula:

$$Q = F_q M_q, K = F_k M_k, V = F_v M_v$$

$$A(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d}} \right) V \quad (2)$$

where Q , K , and V denote the query, key, and value obtained through linear transformations using matrices M_q , M_k , and M_v applied to the input sequences F_q , F_k , and F_v . \sqrt{d} is the dimension of Q and K . In these two modules, F_q , F_k , and F_v are the same feature sequences so that the relations between themselves can be captured. We apply a linear transformation to A to get the output. Other components such as convolution, transpose convolution, and adaptive pooling play supplementary and auxiliary roles.

The initial key points features of BEV and image features with projection from point cloud are denoted as F_{KB} and F_I , respectively. The two converters are represented as CVT_I and CVT_B , and the generator is indicated as GEN , then the features can be obtained by:

$$F_{CVTKI} = CVT_I(F_{KB})$$

$$F_{CVTKB} = CVT_B(CVT_I(F_{KB}))$$

$$F_{GENKB} = GEN(CVT_B(F_I)) \quad (3)$$

The fusion module is shown in Fig. 5. Three groups of BEV features F_{KB} , F_{CVTKB} and F_{GENKB} with different sources are expanded to a new dimension for stacking. Self-attention is employed to capture the 3×3 correlations which are used to weight and integrate them with each other. The enriched features, carrying comprehensive information, compete with each other during the max-pooling, thus retaining the most relevant representations F_{FKB} . Next, F_{FKB} and F_{CVTKI} are fused through cross attention, where F_{CVTKI} serves as keys and values and F_{FKB} is used as queries. The output of cross attention

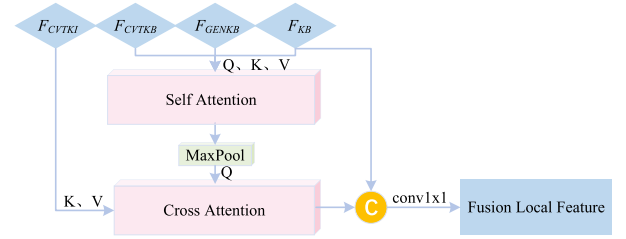


Fig. 5. Illustration of the fusion module. It first stacks the BEV features from different sources and employs self-attention across the stacked dimension to aggregate them. Then they are fused with image features using the cross-attention and a convolutional layer.

is combined with F_{KB} and subsequently fed into a convolutional layer for dimension adaptation. Finally, the fused features F_{KBI} of BEV and image are obtained.

C. Loop Retrieval and Verification

The global descriptor involves aggregating all local features into a single vector representation that encapsulates the entire scene. This approach facilitates efficient loop closure detection. To seamlessly integrate the aggregation module with the feature extraction network and train them in an end-to-end manner, NetVLAD stands out as the optimal choice. NetVLAD's essence lies in its learnable clustering and residual accumulation, making it highly versatile in applications like image retrieval and place recognition. Our fusion strategy is primarily built upon BEV features, complemented by image features. Notably, the information within the fused features is not solely derived from sensor observations but more rather from the network's learned correlation between BEV and images. To enhance the reliability and accuracy of the global descriptor, we employ NetVLAD to aggregate F_{KB} and F_{KBI} separately, yielding D_B and D_{BI} . The weighted sum of these descriptors serves as the global descriptor D for the scene.

$$D = \text{sigmoid}(\lambda) \cdot D_{BI} + (1 - \text{sigmoid}(\lambda)) \cdot D_B \quad (4)$$

where λ is a learnable parameter and sigmoid is the function which maps the weight to 0-1 based on λ .

Global descriptors, local fused features, and key points of the scenes are recorded to serve as a database for loop closure detection. During the retrieval phase, the Euclidean distances between the global descriptors of the query sample and the database are computed. The one with the smallest distance is selected as a loop candidate. In the verification phase, bilateral nearest neighbor matching of local features is employed to establish corresponding key points. Subsequently, an SVD-based transformation matrix estimation using RANSAC is applied to determine the scene transformation. If a sufficient number of local feature matches are found and the transformation matrix can be estimated, the candidate is selected.

D. Loss Function

Our fusion strategy primarily relies on BEV as the main source of features and integrates image features into BEV in the form of high-dimensional neural network information. When designing

the loss function, we not only consider optimizing the fused features but also preserve the optimization for the initial BEV features. Considering the sparsity of the BEV, we determine s^{label} to represent whether grids can be key points by examining the matching relationships of true loop pairs between the centers of BEV grids. If there are J grids that are not empty in a BEV and their scores are denoted by s , then L_{score} that guides the score map is:

$$L_{score} = \frac{1}{J} \sum_{j=1}^J (s_j - s_j^{label})^2 \quad (5)$$

The global descriptors D are trained by triplet loss, which is commonly used for metric learning. For a query sample, we randomly select a positive sample and a negative sample according to the poses, and their global descriptors are represented as D_q , D_p and D_n respectively. Then the triplet loss L_{trip} with a margin m is:

$$L_{trip} = \max(\|D_q - D_p\|_2 - \|D_q - D_n\|_2 + m, 0) \quad (6)$$

To optimize the selection of key points and their local features, we first use unbalance optimal transport algorithm to find the soft correspondence of F_{KB} and F_{KBI} , which are represented by T_B and T_{BI} , respectively. These two correspondences are also the soft matching of the key points P_q and P_p , so we employ weighted SVD to find the homogeneous transformation \hat{H}_B and \hat{H}_{BI} . Their ground truth H can be obtained from the true poses. Then the pose loss L_{pose} and match loss L_{match} are:

$$L_{pose} = \text{mean} \left(\text{abs} \left(\left(\hat{H}_B + \hat{H}_{BI} \right) \cdot P_q - 2H \cdot P_q \right) \right) \quad (7)$$

$$L_{match} = \text{mean} \left(\text{abs} \left(\left(T_B + T_{BI} \right) \cdot P_p - 2H \cdot P_q \right) \right) \quad (8)$$

Features from BEV and images that have matches are used to train $CVTI$ and $CVTB$. Features of key points F_{KB} and F_{GENKB} are used to train GEN . Using F_I , F_B , F_{CVTI} , and F_{CVTB} to represent the original and converted image and BEV features respectively, there are three losses of these modules:

$$L_{CVTB} = \text{mean} \left(1 - \frac{F_B \cdot F_{CVTB}}{(\|F_B\|_2 \cdot \|F_{CVTB}\|_2)} \right) \quad (9)$$

$$L_{CVTI} = \text{mean} \left(1 - \frac{F_I \cdot F_{CVTI}}{(\|F_I\|_2 \cdot \|F_{CVTI}\|_2)} \right) \quad (10)$$

$$L_{GEN} = \text{mean} \left(1 - \frac{F_{KB} \cdot F_{GENKB}}{(\|F_{KB}\|_2 \cdot \|F_{GENKB}\|_2)} \right) \quad (11)$$

Finally, all the above mentioned loss functions are summed to compute the final loss. To balance L_{match} , we introduce a weighted term to it:

$$L_{total} = L_{score} + L_{trip} + L_{pose} + 0.05L_{match} + L_{CVTB} + L_{CVTI} + L_{GEN} \quad (12)$$

IV. EXPERIMENTAL EVALUATION

In this section, we conduct a series of experiments to assess the effectiveness of our proposed method. Initially, we outline the datasets employed in the evaluation, along with the specifics

of implementing our proposed method and configuring of the parameters. Subsequently, a comprehensive presentation of diverse experimental outcomes against existing methods are provided.

A. Datasets and Implementation Details

Our experimentation unfolds across two serialized datasets that are particularly conducive to the exploration of loop closure detection: KITTI [38] and KITTI-360 [39]. The point clouds of them readily align with images when subjected to projection, facilitating seamless correspondence. However, there is only one image perspective corresponding to the point cloud. Our experimental encompasses several sequences of these datasets, including sequences 00, 05, 06, 07, 08, and 09 of KITTI and sequences 00, 04, 05, 06, 09 of KITTI-360. All the deep-learning based methods except for OverlapTransformer are trained on sequences 00, 05, 06, 07, and 09 of KITTI, validated on sequence 08 of KITTI, and tested on sequences 00, 04, 05, 06, 09 of KITTI-360. Ground-truth are identified based on specific criteria within the related works [15], [16]. Specifically, if the Euclidean distance between the poses of two frames is less than 4 m, and their temporal separation exceeds 50 frames, they are classified as loops. Notably, while KITTI and KITTI-360 datasets exhibit a serialized structure, we refrain from imposing any order or sequence-based considerations during our evaluation process. The details of the datasets are shown in Table I.

We limit the valid range of the point cloud along the x and y axes to $(-32, 32)$ m and along the z-axis to $(-2.5, 1.5)$ m, and the resolution of BEV grids is 0.2 m. The local feature dimensions for BEV and image are both 128, and the cluster number of NetVLAD for BEV features and fused features is 16, so the dimension for global descriptors is 2048. We apply random rigid transformations to the point cloud for data augmentation. Specifically, they are ± 3 m of translation and $\pm 3^\circ$ of rotation on the x and y axes and ± 0.3 m of translation and $\pm 180^\circ$ of rotation on the z axis. In order to reduce the model's GPU memory usage, we first crop and pad the image to 1152×384 and then downsample it to 576×192 . No data augmentation is applied to the images. The model undergoes training on NVIDIA RTX TITAN GPU with 24 GB, running for 100 epochs with a batch size of 6. The learning rate of Adam optimizer is 0.001 and the weight decay is 5×10^{-6} . The learning rate experiences a decay of 0.99 for each epoch.

B. Loop Closure Detection

To evaluate the performance of the loop closure detection, we compare our proposed methods against M2DP [19], Scan context [17], LiDAR Iris [18], OverlapTransformer [21], LCD-Net [15], and BEVLCD [37]. In this letter, there are some differences between BEVLCD and the method in the original letter. Due to more efficient data augmentation than [37], we use traditional network layers. M2DP, Scan context and LiDAR Iris are handcrafted methods and others are deep learning based methods. We use BEVLCD+P and FUSION to represent methods of BEVLCD with Position encoder and our proposal. For convenience, when we test FUSION, the input of image is set to 0. The Euclidean distance is used between global descriptors

TABLE I
DETAILS OF DATASETS

Sequences	KITTI						KITTI-360				
	00 ¹	05 ¹	06 ¹	07 ¹	08 ²	09 ¹	00 ³	04 ³	05 ³	06 ³	09 ³
Num. of scans	4541	2761	1101	1101	4071	1591	10514	11052	6291	9186	13724
Num. of scans with loop	1682	1046	563	183	654	48	4798	4494	4285	4967	8489
Num. of scans only with reverse loop	23	2	0	0	613	0	3124	3961	3712	2996	2972
% of scans only with reverse loop	1.37%	0.19%	0%	0%	93.73%	0%	65.11%	88.14%	86.63%	60.32%	35.01%

If a pair of loop with the yaw difference is greater than 90°, it is treated as reverse loop. Sequences with ¹, ² and ³ represent splits for training, validation and testing respectively.

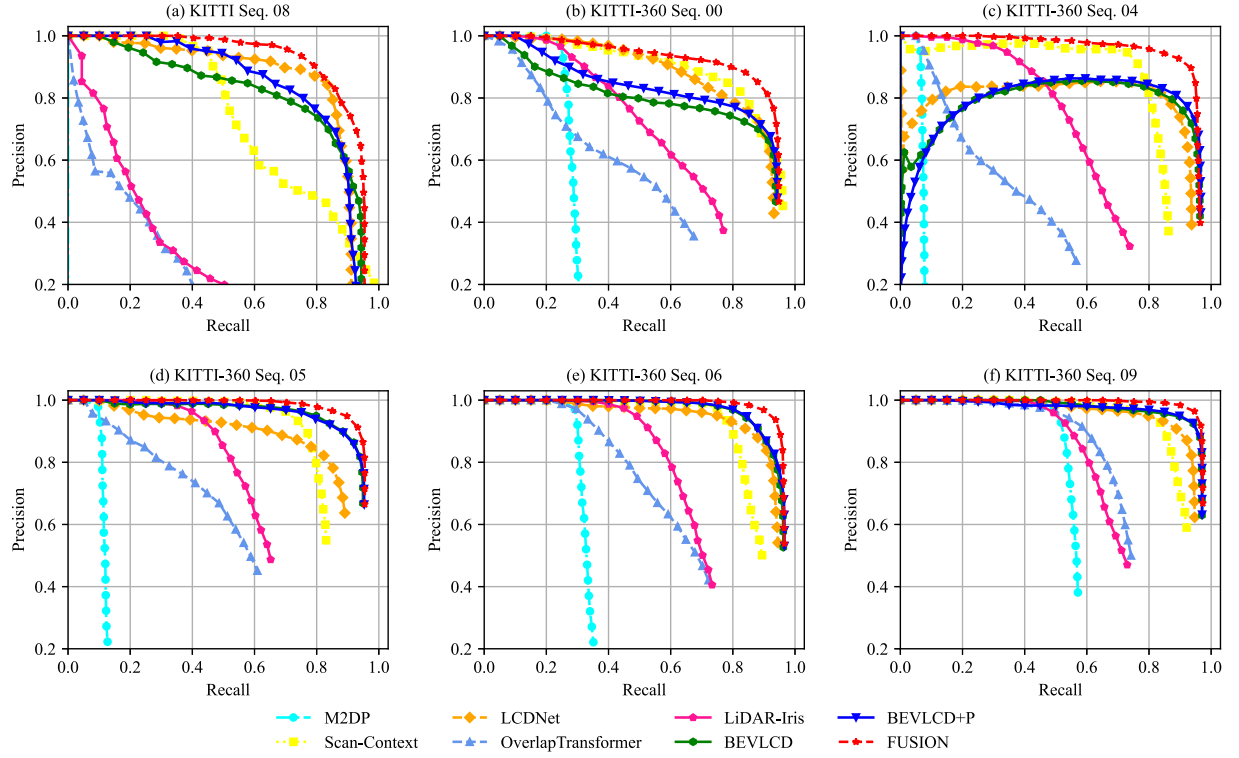


Fig. 6. Precision-recall curves of the proposed methods and others for loop closure detection.

TABLE II
COMPARISON OF LOOP CLOSURE DETECTION

		KITTI 08		KITTI-360 00		KITTI-360 04		KITTI-360 05		KITTI-360 06		KITTI-360 09	
		AP	R@1	AP	R@1	AP	R@1	AP	R@1	AP	R@1	AP	R@1
M2DP	HF&2D	0.001	0.018	0.285	0.310	0.074	0.088	0.118	0.133	0.328	0.354	0.555	0.572
Scan context	HF&2D	0.733	0.827	0.872	0.790	0.809	0.676	0.810	0.657	0.862	0.714	0.900	0.749
LiDAR Iris	HF&2D	0.295	0.766	0.620	0.777	0.613	0.739	0.597	0.657	0.666	0.735	0.671	0.730
OverlapTransformer	DL&2D	0.217	0.531	0.469	0.694	0.358	0.578	0.485	0.617	0.607	0.727	0.699	0.747
LCDNet	DL&3D	<u>0.841</u>	0.911	<u>0.848</u>	0.932	<u>0.766</u>	0.937	0.820	0.892	0.909	0.946	0.922	0.947
BEVLCD	DL&2D	0.802	<u>0.943</u>	0.769	0.939	0.759	<u>0.965</u>	0.921	0.949	0.941	0.962	<u>0.954</u>	0.970
BEVLCD+P	DL&2D	0.828	0.927	0.804	<u>0.944</u>	0.761	<u>0.967</u>	<u>0.924</u>	<u>0.953</u>	<u>0.944</u>	<u>0.966</u>	0.951	<u>0.971</u>
FUSION	DL&2D	0.908	0.954	0.895	0.948	0.940	0.965	0.945	0.955	0.959	0.967	0.968	0.973

“HF” and “DL” respectively represent methods based on handcrafted features and deep learning, while “3D” and “2D” represent extracting features from original 3D points and 2D projected pseudo images of point clouds respectively. AP of Scan context is obtained with 50 candidates and R@1 is with 1 candidate. **Bold** and underline indicate the best and second-best results, respectively.

to represent their similarity. If a frame in the database has the minimum Euclidean distance to the current frame, and the frame interval between them is greater than 50, as well as the Euclidean distance is less than or equal to a threshold thr , it is considered as a loop for the current frame. By adjusting the thr , results under different constraints on loop closure detection can be obtained,

leading to commonly used qualitative evaluation in the form of a PR (Precision-Recall) curve, as well as quantitative evaluation metrics such as Average Precision (AP) and Recall@1 (R@1).

The PR curves and evaluation metrics are shown in Fig. 6 and Table II. Because BEV-based methods have fewer model parameters compared to directly processing raw point cloud

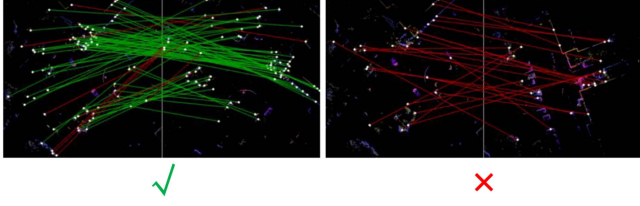


Fig. 7. Matching between key points (white points) of the two frames allows to estimate their relative pose, followed by obtaining inliers. Correct matches are indicated by green lines, while incorrect ones are represented by red lines. Sufficient correct matches signify verification passes (left), otherwise, it does not (right).

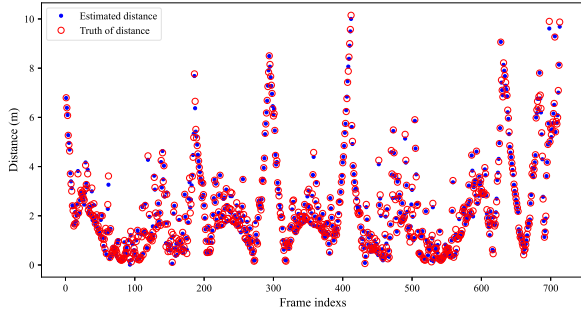


Fig. 8. Example of distance from the estimated poses and true poses on KITTI 08.

methods, their generalization performance is better. According to the results, the Position Encoder for key points and the fusion of image and BEV feature extraction proposed in this letter have both improved the performance of loop closure detection. FUSION, in particular, achieved the best results in most sequences. This is especially evident in sequences with a high proportion of reverse loops, such as 08 of KITTI and 04 of KITTI-360. In scenarios with low levels of structure, the consistency of a point cloud's description of the same object from different angles may be poor. This can result in methods relying solely on point cloud information performing normally on these two sequences. But when the key point features are enhanced with image information, the performances on these two sequences are significantly improved. Additionally, the images cannot directly provide consistent descriptions of scenes in reverse loops. However, our proposed fusion framework can explore the relationship between BEV and images, not just directly fuse them. As a result, it can effectively handle such misalignments of images and point clouds.

The precision of loop closure detection is crucial, so many methods use local information to establish global consistency constraints. As shown in Fig. 7, the transformation estimated in the verification phase can reflect the consistency of key points matching and exhibits good invariance to rotation and translation. Since the geometric information of the point cloud does not introduce scale changes, this verification method demonstrates considerable accuracy. However, using ground-truth based on inter-frame Euclidean distances for evaluating the precision of loops is not entirely reasonable, as the distance of loops on different scenarios cannot be same. An example of the detection on Fig. 8 shows that the distances of loops vary in different places. The best way to find loops for all samples is pairwise comparisons, which is very time-consuming. Therefore, we

TABLE III
COMPARISON ABOUT NUMBER AND PRECISION OF LOOP CLOSURE DETECTION

Seq	BEVLCD		BEVLCD+P		FUSION	
	Num	P	Num	P	Num	P
08	733	1.000	741	1.000	751	1.000
00	4751	1.000	4733	1.000	4829	1.000
04	4454	0.999	4455	1.000	4468	1.000
05	4377	1.000	4337	1.000	4432	1.000
06	5045	1.000	5042	1.000	5087	1.000
09	8643	1.000	8615	1.000	8679	1.000

Num, and P represent the number of samples where loop closures were detected, and the precision of loop closure detection based on pose, respectively.

TABLE IV
ABLATION STUDY ON PIPELINE CHOICE

F_{KB}	F_{GENKB}	F_{CVTKB}	F_{CVTKI}	AP	R@1
				0.828	0.927
✓				0.842	0.940
✓	✓			0.876	0.948
✓	✓	✓		0.903	0.950
✓	✓	✓	✓	0.908	0.954

Average precision (AP) and Recall@1 (R@1) of LCD evaluated on 08 of KITTI for different inputs of fusion module.

adopt an approach that doesn't rely on ground truth to assess the precision, that is to compare the estimated loop pose and the true loop pose. If the error between them is over 1 m, they are regarded as a false loop. Because we can only assess the correctness of the detection results and cannot determine any omissions that are not detected, we only report the number and precision of the results. As shown in Table III, it can be observed that the fusion of images into point clouds can make the method detect more loops while still ensuring precision, which demonstrates the benefits of fusion.

C. Ablation Study

In this section, we conduct ablation experiments to validate each part of the pipeline. We change the settings of the module in Fig. 5 to perform fusion of different information in our pipeline, and the results are shown in Table IV. None of the information is used means there is no fusion in the pipeline, and no F_{CVTKI} means cross-attention is removed. Other settings of the experiments are the same as before and those experiments that require images are only required during training. The results indicate the effectiveness of our designed approach, which leverages information interaction to explore the correlation of multi-modal features and enhancing them to improve the performance of LCD.

D. Runtime

The runtime of a LCD method is also worth noting. Due to the lightweight network of the proposed method and its focus on sparse features, it can run very efficiently. We have recorded the average runtime of the proposed method in three components: feature extraction, loop retrieval, and loop verification on the KITTI 08 sequence. They are 42.50 ms, 0.89 ms, and 3.05 ms, respectively, with a total of 46.44 ms, demonstrating real-time capability.

V. CONCLUSION

In this letter, we introduce a loop closure detection method that fuse asymmetric images with point clouds. Independent backbone networks are used to extract features from images and BEVs. Feature conversion and generation modules are used to uncover the relationship between deep features of BEV and images. In the fusion stage, this relationship is managed and fused using attention mechanisms. Finally, global descriptors aggregated by NetVLAD and distribution of local features are used to perform retrieve and verify. The proposed method can embed image information into the BEV processing modules, thereby improving algorithm performance without the need of images. Experimental results on publicly available datasets demonstrate that the proposed method exhibits outstanding performance. The lightweight model design and sparse data processing enable our method to run efficiently.

REFERENCES

- [1] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "The revisiting problem in simultaneous localization and mapping: A survey on visual loop closure detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 19929–19953, Nov. 2022.
- [2] X. Li et al., "Homogeneous multi-modal feature fusion and interaction for 3D object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 691–707.
- [3] A. Piergiovanni, V. Casser, M. S. Ryoo, and A. Angelova, "4D-net for learned multi-modal alignment," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 15435–15445.
- [4] Z. Liu et al., "BEVFusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 2774–2781.
- [5] Y. Zeng et al., "LIFT: Learning 4D LiDAR image fusion transformer for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17172–17181.
- [6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, 2004.
- [7] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. 9th Eur. Conf. Comput. Vis. Comput. Vis.*, 2006, pp. 404–417.
- [8] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to sift or surf," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.
- [9] D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: Self-supervised interest point detection and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 224–236.
- [10] X. Zhao, X. Wu, J. Miao, W. Chen, P. C. Chen, and Z. Li, "ALiKE: Accurate and lightweight keypoint detection and descriptor extraction," *IEEE Trans. Multimedia*, vol. 25, pp. 3101–3112, 2023.
- [11] Z. Zhou et al., "NDT-transformer: Large-scale 3D point cloud localisation using the normal distribution transform representation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 5654–5660.
- [12] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 3384–3391.
- [13] Y. Cui, Y. Zhang, J. Dong, H. Sun, and F. Zhu, "LinK3D: Linear keypoints representation for 3D LiDAR point cloud," *IEEE Robot. Automat. Lett.*, 2024.
- [14] 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.
- [15] 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.
- [16] J. Arce, N. Vödisch, D. Cattaneo, W. Burgard, and A. Valada, "PADLoC: LiDAR-based deep loop closure detection and registration using panoptic attention," *IEEE Robot. Automat. Lett.*, vol. 8, no. 3, pp. 1319–1326, Mar. 2023.
- [17] 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.
- [18] 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 Int. Conf. Intell. Robots Syst.*, 2020, pp. 5769–5775.
- [19] L. He, X. Wang, and H. Zhang, "M2DP: A novel 3D point cloud descriptor and its application in loop closure detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 231–237.
- [20] X. Chen et al., "OverlapNet: Loop closing for LiDAR-based SLAM," in *Proc. Robot.: Sci. Syst.*, Corvallis, Oregon, USA, Jul. 2020.
- [21] 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.
- [22] Sivic and Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, 2003, pp. 1470–1477.
- [23] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5297–5307.
- [24] H. Yue, J. Miao, Y. Yu, W. Chen, and C. Wen, "Robust loop closure detection based on bag of superpoints and graph verification," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 3787–3793.
- [25] H. Yue, J. Miao, W. Chen, W. Wang, F. Guo, and Z. Li, "Automatic vocabulary and graph verification for accurate loop closure detection," *J. Field Robot.*, vol. 39, no. 7, pp. 1069–1084, 2022.
- [26] K. Zhang, J. Ma, and J. Jiang, "Loop closure detection with reweighting NetVLAD and local motion and structure consensus," *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 6, pp. 1087–1090, Jun. 2022.
- [27] J. Ma, K. Zhang, and J. Jiang, "Loop closure detection via locality preserving matching with global consensus," *IEEE/CAA J. Automatica Sinica*, vol. 10, no. 2, pp. 411–426, Feb. 2023.
- [28] Y. Zhu, Y. Ma, L. Chen, C. Liu, M. Ye, and L. Li, "GOSMatch: Graph-of-semantics matching for detecting loop closures in 3D LiDAR data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5151–5157.
- [29] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "PointPainting: Sequential fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4604–4612.
- [30] C. Wang, C. Ma, M. Zhu, and X. Yang, "PointAugmenting: Cross-modal augmentation for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11794–11803.
- [31] L. Zhao, H. Zhou, X. Zhu, X. Song, H. Li, and W. Tao, "LIF-Seg: LiDAR and camera image fusion for 3D LiDAR semantic segmentation," *IEEE Trans. Multimedia*, vol. 26, pp. 1158–1168, 2024.
- [32] S. Gu, J. Yang, and H. Kong, "A cascaded LiDAR-camera fusion network for road detection," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 13308–13314.
- [33] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.
- [34] X. Bai et al., "Transfusion: Robust LiDAR-camera fusion for 3D object detection with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1090–1099.
- [35] Z. Chen et al., "AutoAlign: Pixel-instance feature aggregation for multi-modal 3D object detection," in *Proc. Int. Joint Conf. Artif. Intell.*, 2022, pp. 1–7.
- [36] M. Yu, L. Zhang, W. Wang, and H. Huang, "Loop closure detection by using global and local features with photometric and viewpoint invariance," *IEEE Trans. Image Process.*, vol. 30, pp. 8873–8885, 2021.
- [37] D. Cao, H. Yue, Z. Liu, X. Wu, and W. Chen, "BEVLCD: Real-time and rotation-invariant loop closure detection based on BEV of point cloud," *IEEE Trans. Instrum. Meas.*, vol. 72, 2023, Art. no. 5026213.
- [38] 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.
- [39] Y. Liao, J. Xie, and A. Geiger, "KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2D and 3D," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3292–3310, Mar. 2023.