BEV-LSLAM: A Novel and Compact BEV LiDAR SLAM for Outdoor Environment

Fengkui Cao, *Member, IEEE*, Shaocong Wang, Xieyuanli Chen, *Member, IEEE*, Ting Wang, and Lianqing Liu, *Senior Member, IEEE*

***Abstract*—LiDAR-based SLAM is an essential technology for autonomous robots, benefited from its high accuracy and scale invariance. Interestingly, researchers have been increasingly focusing on establishing simple, efficient, but effective LiDAR SLAM systems recently. In this paper, we propose a novel and compact LiDAR-only SLAM system BEV-LSLAM, leveraging visual features in BEV view for all the steps of pipeline including pose estimation, mapping, loop closing and back-end graph optimization. The proposed BEV features are more stable than traditional geometrical features, which can be adapted to various LiDARs sensors without changing the hyperparameters. In addition, benefited from filtering vulnerable features based on tracking process in consecutive frames, only high-quality feature points are used for lightweight point-cloud map construction. Extensive experiments on UrbanLoco, KITTI, and our 16-channel LiDAR datasets prove the superiority of our approach, compared with state-of-the-art LiDAR SLAM methods.**

***Index Terms*—SLAM, localization, mapping.**

I. Introduction

SIMULTANEOUSLY localization and mapping (SLAM) is an essential technique for autonomous robots, which enables them to explore unknown environments and localize themselves without GPS's aid. Since LiDAR sensors is not sensitive to lighting conditions and own high accuracy of ranging, LiDAR SLAM has extensively attracted attention of researchers in robotics community [1]. Usually, feature extraction and association is the key step of SLAM tasks. Most of them [2], [3], [4], [5] directly extract geometrical features (e.g. key points, lines or surfaces) from raw LiDAR pointcloud, whose performances depend on the quality of presenting the scene structures. However, these methods has a high dependence on the density of point cloud, and a large number of hyperparameters need to be modified with different beams of LiDAR. This is also the reason why researchers integrate the IMU or visual sensors into LiDAR SLAM system to compensate for the instability of sparse point cloud feature extraction in localization and mapping task [6], [7], [8], [9], [10]. Some methods [11], [12], [13] detect structural features based on LiDAR scanning images (e.g. range and intensity image). However, their performances are limited to sparse vertical resolution of sparse-channel LiDARs, such as Velodye HDL-32 and VLP-16. To avoid the feature extraction process, ICP-based LiDAR SLAM systems [14], [15], [16] use direct registration of LiDAR scan for motion estimation and mapping. Usually, these methods often need to employ severe downsampling to ensure the real-time performance of the system, which may lead to a decrease in map quality to a certain extent.

Different from the abovementioned methods, we introduce BEV projection of LiDAR scans as an intermediary for key point feature extraction in outdoor environment. With the layouts of scene structures in bird-view, the proposed BEV features has better adaptability to different kinds of LiDAR sensors with few hyperparameter adjustment. In mapping process,we aggregate all the points in the same projecting pillars with tracked features to construct denser point cloud for scan-to-map mapping. As shown in Fig. 1, the proposed BEV features simultaneously serve for pose tracking, mapping, loop closing and back-end graph optimization, which makes the pipelines of LiDAR SLAM compact.

The main contributions of our approach are quadruple:

- A novel and compact LiDAR-only SLAM for outdoor environment, extracting visual features from BEV projection for pose estimation, mapping, loop closing and graph optimization, which is suitable for various LiDARs without additional hyperparameter adjustment.
- A novel enhanced BEV representation of LiDAR scan is designed combining height, intensity and geometrical angles among the points to highlight the structures (e.g. pillar or edge), suitable for feature detection.

Received 5 September 2024; accepted 7 January 2025. Date of publication 20 January 2025; date of current version 30 January 2025. This article was recommended for publication by Associate Editor S. Scherer and Editor S. Behnke upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grant 62203091 and Grant U20A20201, in part by the China Postdoctoral Science Foundation under Grant GZB20230804, in part by the Natural Science Foundation of Liaoning Province under Grant 2024-BSBA-49, and in part by the Autonomous Project of State Key Laboratory of Robotics under Grant 2024-Z09. *(Fengkui Cao and Shaocong Wang contributed equally to this work.) (Corresponding authors: Ting Wang; Lianqing Liu.)*

Fengkui Cao, Ting Wang, and Lianqing Liu are with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China, and also with the Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China (e-mail: caofengkui@sia.cn; wangting@sia.cn; lqliu@sia.cn).

Shaocong Wang is with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China, and with the Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China, and also with the University of Chinese Academy of Sciences, Beijing 101408, China (e-mail: wangshaocong@sia.cn).

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

The code will be available at: https://github.com/ROBOT-WSC/BEV-LSLAM.git.

Digital Object Identifier 10.1109/LRA.2025.3531727

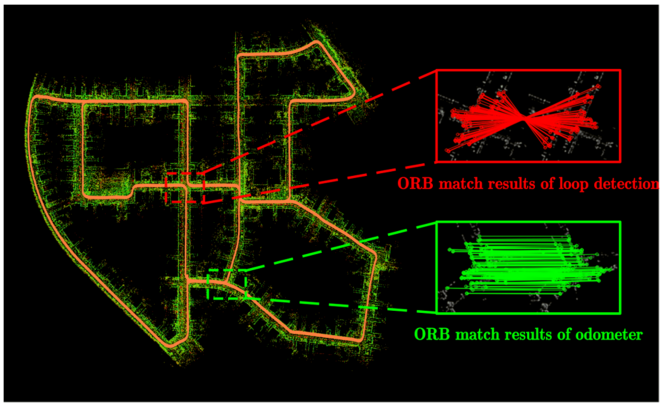

Fig. 1. The proposed approach leverages visual features as intermediaries for key point feature extraction, serving for pose tracking and loop closing simultaneously. It can be seen from the above picture, our approach achieves good feature matching performance in tracking and detecting loop closures. In addition, the resulting global pointcloud map is lightweight without sacrifice in presenting scenes fundamental structures.

- A local BA process of tracking stable BEV features in consecutive frames and only points related to stable BEV features are used for pose tracking and mapping, not only improving the system efficiency, but also resulting in a lightweight global pointcloud map.
- Extensive experiments on 64-, 32- and 16-channel LiDAR data demonstrate the superiority of our approach compared with baseline methods.

II. Related Works

A. LiDAR SLAM Based on Features

In the past decades, LiDAR SLAM has been extensively studied in robotics community and there are many excellent works leaving deep impressions. LOAM [2] can be regard as a pioneer work of LiDAR SLAM, which created much excitement in ten years ago. Then, there are many following works proposed, such as LeGO-LOAM [3] and F-LOAM [4], to improve the LiDAR-only SLAM performance. LeGO-LOAM is a lightweight and ground-optimized LiDAR odometry and mapping method, leveraging ground information for optimization and loop closing to enhance the localization system. Considering the high time cost of iterative calculation in SLAM system, F-LOAM adopts a non-iterative two-stage distortion compensation to provide a computationally efficient and accurate framework for LiDAR based SLAM. However, these feature detecting processes are similar with LOAM, whose performance is reduced with sparse-channel LiDARs. To compensate the divorce of geometrical features, intensity information was encoded in the feature detecting process in many works. Intensity-SLAM [17] introduces intensity features into SLAM system, in which intensity features also serve for loop closing based on scan context module [18]. SuMa++ [19] encoded geometric, intensity and semantic features in SLAM system for adapting dynamic environments.Since Normal Distributions Transform (NDT) is usually utilized in the point clouds registration, NDT-LOAM [20] proposed a weighted NDT combined with a Local Feature Adjustment (LFA) to process the point clouds and improve the registration accuracy. Although these methods get a good balance between efficiency and precision, these methods has a high dependence on the density of point cloud, and a large number of hyperparameters need to be modified with different beams of LiDAR.

B. LiDAR SLAM Based on ICP

Owing to the improvement of ICP-variants and the development of parallel computation, direct ego-motion estimation based on ICP registration without feature extraction becomes popular. Direct LiDAR Odometry (DLO) [14] quickly estimates the ego-motion of LiDAR through parallel computation and special keyframe management. CT-ICP [15] introduced combined continuity in the scan matching and discontinuity between scans allowing both the elastic distortion of the scan during the registration and the robustness to high frequency motions from the discontinuity. Instead of adding more complexity to the ego-motion estimation process, KISS-ICP [16] removed a majority of parts and focused on the core elements obtaining a surprisingly effective system. MULLS [21] extracted roughly classified feature points (ground, facade, pillar, beam, etc.) using dual-threshold ground filtering and principal components analysis, and then a multi-metric linear least square iterative closest point algorithm is proposed for scan-to-submap refinement.In fact, direct LiDAR SLAM methods based on ICP-invariants often need to employ severe downsampling to ensure the real-time performance of the system, which will equally decrease the role of essential structures.

C. LiDAR SLAM Based on Projection Images

Considering the sparse and scattered distribution of LiDAR point clouds in 3D space, leveraging images to represent LiDAR scans more compact is a relevant intermediary bank for feature engineering. SuMa [11] constructed a surfel-based map based on range image and estimated the changes in the robot's pose by exploiting the projective data association. RI-LIO [12] is a reflectivity image assisted tightly-coupled LiDAR-inertial odometry (LIO) framework that introducing additional reflectivity texture information to efficiently reduce the drift of geometric-only methods. Du et al. [13] extracted feature points from the LiDAR intensity images for pose tracking and loop closing in indoor environments. However, this work depends on dense-channel imaging LiDARs like Ouster-128, which is not suitable for low-cost robots.

In this paper, we propose BEV-LSLAM, an novel, universal and compact LiDAR-only SLAM system based on BEV projection, allowing few additional hyperparameter adjustment with various LiDAR sensors. The proposed features are detected using visual features as intermediaries and their associations are compactly suitable for all the pipeline including pose estimation, mapping, loop closing and optimization. In addition, benefiting from the feature tracking, only high-quality refined points related to stable BEV features are used for accurate pose estimation and lightweight mapping without additional downsampling operations.

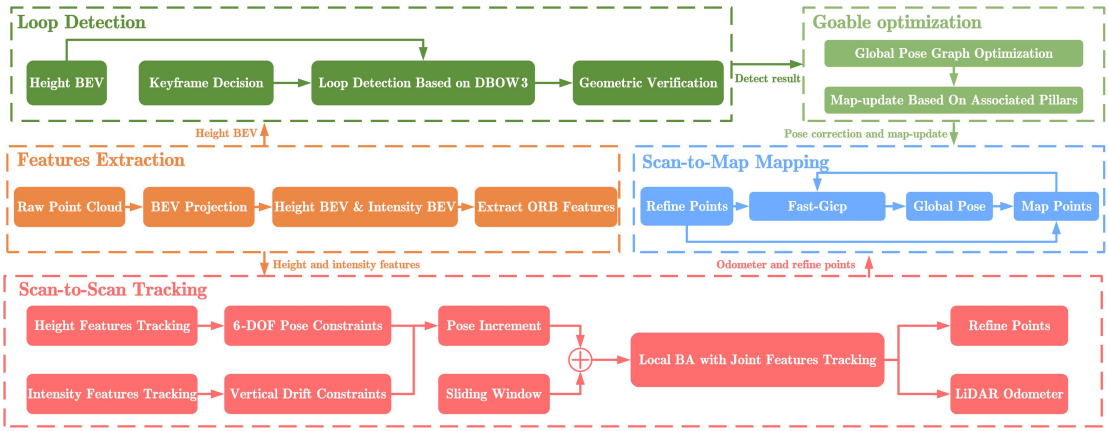

Fig. 2. System overview of BEV-LSLAM, including features extraction, scan-to-scan tracking, scan-to-map refinement and loop closing modules. Note that BEV projection keeps the scale-invariance of LiDAR scan well and is suitable for various LiDARs with different channel; ORB features locate on the corners and edges of objects in BEV images, where are corresponding to the fundamental structures of LiDAR scans benefiting for tracking; Only the points in pillars matched features belonging to are fed into the scan-to-map refinement, which combines Fast-GICP technique improving the efficiency effectively; Note visual features can serve all the pipeline making whole system compact.

III. Methodology

A. System Overview

As shown in Fig. 2, BEV-LSLAM consists of feature extraction, scan-to-scan tracking, scan-to-map refinement, loop closing and graph optimization modules. Distinctively, key point features are detected using visual features (e.g. ORB features) in BEV view as intermediaries, which serve for pose estimation, mapping, loop closing and graph optimization, making the pipeline compactly. To improve the robustness and local consistency of pose tracking, we propose an particular local BA module among continuous scans in scan-to-scan tracking. Scan-to-map refinement can enhance the robustness of pose estimation and further refine the mapping task, which is usually time-consuming. To get a balance between efficiency and map quality in mapping process, the high-quality feature points filtered by pose tracking and local BA module are fed into the Fast-ICP [22] operation, not only improving efficiency, but also resulting in lightweight point cloud map. Benefited from the fast search capability of DBOW3 [23], loop closing can operate online in this work.

B. Feature Extraction

1) BEV Projection: Motivated by the pipelines of visual SLAM, we seek to detect "visual" features in LiDAR scans. The reason why we chose BEV image instead of LiDAR scanning image is to generate scale-invariant representations and make our approach suitable for various LiDARs without additional hyperparameter adjustment. Once a LiDAR scan L is obtained, the pointcloud is divided into a grid map G with 0.4 m steps, which is converted to a Height BEV grayscale image B^H and an Intensity BEV grayscale image B^I. For B^H, p_{ij}^H is the raw point with the max height in G_{ij}, corresponding to the i-th row and j-th column pixel in B^H. The pixel value B_{ij}^H is defined as a positively correlated variable of the p_{ij}^H's height. For B^I, p_{ij}^I is the raw point with the max intensity in G_{ij}, corresponding to the i-th row and j-th column pixel in B^I. The pixel value B_{ij}^I is defined as a positively correlated variable of the p_{ij}^I's intensity. The B^H and B^I are defined as follows:

$$\begin{cases} B_{ij}^H = (H_{ij} - H_{\min}) / (H_{\max} - H_{\min}) \\ B_{ij}^I = (I_{ij} - I_{\min}) / (I_{\max} - I_{\min}) \end{cases} \tag{1}$$

where, H_{ij} is the Height value of p_{ij}^H; I_{ij} is the intensity value of p_{ij}^I; $H_{\max}$ and $H_{\min}$ are the max and min height value of raw point cloud respectively; $I_{\max}$ and $I_{\min}$ are the max and min intensity value of raw point cloud respectively.

2) BEV Enhancing: To further enhance the discriminations of structures in BEV image and filter the flat ground information, geometrical information and intensity curvature among neighboring points are leveraged to intensify the pixel values. As shown in Fig. 3, we select 16 representative neighboring pixel regions to enhance B_{ij}^H and B_{ij}^I. For Height BEV image, the 16 regions are divided into 8 point pairs $\{(p_1^A, p_1^B), (p_2^A, p_2^B), \ldots, (p_8^A, p_8^B)\}$ diagonally. The minimum geometrical angle between the centric point p_{ij}^H and each of the eight neighboring point pairs are used to enhance the discriminations of structures, which naturally ignores ground points while highlighting edge and sharp points. The process of refining the pixel value B_{ij}^H is defined as follows:

$$B_{ij}^H = \frac{B_{ij}^H}{2} \cdot \left(\max_{k \in [1,8]} \left(\frac{(p_k^A - p_{ij}^H) \odot (p_k^B - p_{ij}^H)}{|p_k^A - p_{ij}^H| \cdot |p_k^B - p_{ij}^H|} \right) + 1 \right) \tag{2}$$

where, the k-th neighboring diagonal point pair of p_{ij}^H is defined as (p_k^A, p_k^B). The weight of B_{ij}^H is positively related to the maximum cosine of the angle between vectors $p_{ij}^H \to p_k^A$ and $p_{ij}^H \to p_k^B$. When the weight is small, it indicates that the surrounding environment of p_{ij}^H is relatively flat, which may be the ground point. When the weight is large, it indicates that p_{ij}^H is more prominent relative to the surrounding points, which can better represent the geometric structure in the environment.

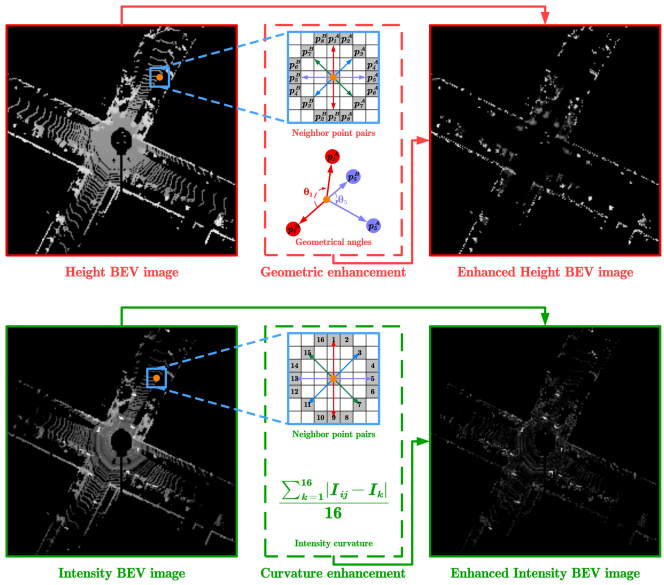

Fig. 3. Geometrical enhancement of Height BEV image of LiDAR scan using the geometrical angles between the centric point and neighboring diagonal point pairs.Curvature enhancement of Intensity BEV image of LiDAR scan using the intensity curvature between the centric point and neighboring diagonal point pairs.

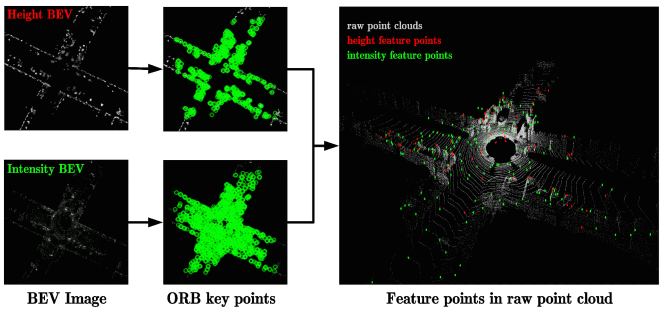

Fig. 4. An example presenting the extracted key point features from BEV images of LiDAR scan, which tend to be located on fundamental structures of scene.

For Intensity BEV image, the intensity curvature calculated between the centric point and each of the 16pixel regions is employed to enhance the edge intensity points.

3) Visual Feature Extraction: Owing to the high quality of BEV images highlighting the details (e.g. corners and edges) of scenes, ORB features are suitable for key point feature extraction. Usually, ORB features tend to be in concentrated distribution, we refer to the idea of ORB-SLAM2 [24] using uniform grid division to supervise ORB feature detection. As shown in Fig. 4, the ORB features detected from BEV images are in uniform distribution benefiting for scan pose tracking, in which each ORB feature is related to a key point in raw LiDAR pointcloud.

C. Scan-to-Scan Tracking

1) Feature Tracking: Since the convenience of visual matching technique, we match ORB features in continuous LiDAR scans to obtain feature point pairs for motion estimation. To remove the outliers, GMS [25] is leveraged to check the geometrical consistency constraints of matched feature point pairs. Denote the matched key points in current Height BEV image as $p^H = \{p_0^H, p_1^H, \ldots, p_n^H\}$, while the corresponding matched key points in last Height BEV image are defined as $\tilde{p}^H = \{\tilde{p}_0^H, \tilde{p}_1^H, \ldots, \tilde{p}_n^H\}$. Then denote the matched key points in current Intensity BEV image as $p^I = \{p_0^I, p_1^I, \ldots, p_n^I\}$, while the corresponding matched key points in last Intensity BEV image are defined as $\tilde{p}^I = \{\tilde{p}_0^I, \tilde{p}_1^I, \ldots, \tilde{p}_n^I\}$. For simplicity, p^H and $\tilde{p}^H$ construct the residuals r_k^H to estimate the 6-DOF state $\chi_k^{k-1} = [R_k^{k-1} \quad t_k^{k-1}]^T$ between the k-th frame and $k-1$-th frame. Here, R_k^{k-1} and t_k^{k-1} are the rotation matrix and translation vector, respectively. However, the feature points extracted on Height BEV images are concentrated in the local highest points, which easily aggravates the cumulative error of odometer drifts on the z axis. Therefore, we extend denser feature points aggregating all the points in the same projecting pillar with extracted key points to constrain the pose changes in the vertical direction. Therefore, we use strength feature points with more uniform distribution to constrain the pose changes in the vertical direction. p^I and $\tilde{p}^I$ construct the residuals r_k^I to constraint the vertical state of χ_k^{k-1}. The residuals optimized during the odometer process are defined as follows:

$$
\begin{aligned}
r_k^H\left(p^H, \tilde{p}^H, \boldsymbol{\chi_k^{k-1}}\right) &= \sum_{i=0}^{n} w_i^H \left(\boldsymbol{R_k^{k-1}} p_i^H + \boldsymbol{t_k^{k-1}} - \tilde{p}_i^H\right), \\
r_k^I\left(p^I, \tilde{p}^I, \boldsymbol{\chi_k^{k-1}}\right) &= \sum_{j=0}^{m} W_j^I \left(\boldsymbol{R_k^{k-1}} p_j^I + \boldsymbol{t_k^{k-1}} - \tilde{p}_j^I\right), \\
W_j^I &= \left[0, 0, w_j^I\right]
\end{aligned} \tag{3}
$$

where, w_i^H is the weight of i-th matched point pairs based on Height BEV image, while w_j^I is the weight of j-th matched point pairs based on Intensity BEV image, all depending on the score of feature matching. These residuals can be solved as a nonlinear least squares problem, and we use the Levenberg-Marquardt algorithm to solve it.

2) Local BA With Joint Features Tracking: Considering many features exist in several adjacent frames simultaneously, we form a short-term sliding-window including n LiDAR scans for joint features tracking, in which matched ORB features between current frame and last frame are associated with the previous frames autonomously in the way shown in Fig. 5. Then we can use the tracked joint features to achieve local BA operation and avoid the redundancy of feature matching between current frame with previous frames. After the joint features are determined, we can use them to construct up to n sets of residual errors similar to (3) to constrain the pose increment between the k-th frame and the historical frame that can be tracked to the common view point in the sliding window. Since the global pose of the history frame has been optimized before, in order to ensure better computational efficiency, we take the global pose of the history frame as a constant, and only optimize the global state $\hat{\chi}_k^w = [\hat{R}_k^w \quad \hat{t}_k^w]^T$ of the k-th frame on the basis of the odometer.

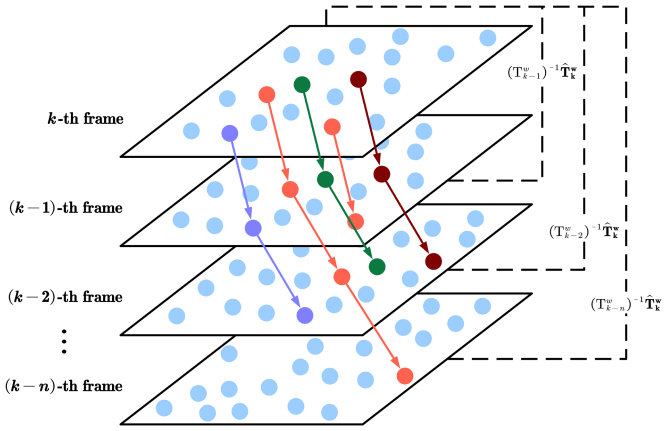

Fig. 5. Local BA with Joint Features Tracking in continuous LiDAR scans. The dots present the detected features, while the narrows present the tracking routes of Joint features matched among adjacent frames.

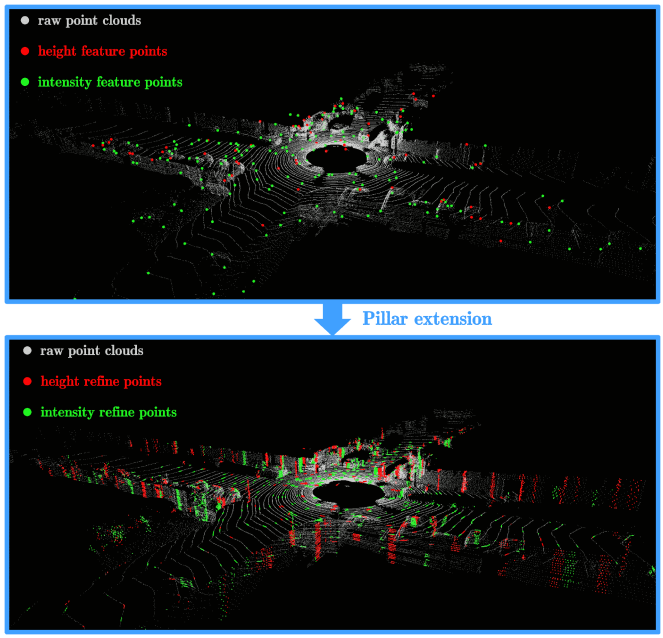

Fig. 6. Key points refinement using pillar extension for scan-to-map mapping, which enhance the registration process.

D. Scan-to-Map Mapping

In scan-to-scan tracking, only key points one-by-one corresponding to matched features are used to estimate the relative transformation between adjacent frames. To further refine the motion estimation and finish the mapping process, scan-to-map registration usually is operated in LiDAR SLAM system. Considering the matched feature points are sparse, we aggregate all the points projected into the same pillars with matched feature points, to construct a denser feature point cloud for mapping. An aggregated key feature point cloud is shown in Fig. 6, highlighting more structures of scenes. A refinement key feature point cloud is shown in Fig. 6.The red points are refined with features on Height BEV image, while the points are refined with features on Intensity BEV image. Then, the denser feature point cloud is leveraged for scan-to-map registration. To ensure the efficiency of mapping, we integrate the key feature point cloud of discrete key frames in short-term sliding window to form sub-map, which is fed into Fast-GICP operation combined with the key feature point cloud of current frame to finish mapping procedure. Note the point cloud map constructed by refine points can ensure the light weight of the map while reduce the loss of structure without additional downsampling.

E. Loop Closing

In this section, we introduce our previous proposed BEV-based loop closing method [26], which is proved viewpoint-invariant and suitable for sparse-channel LiDAR.

1) Global Searching: The loop closure detection in this paper mainly relies on the Height BEV extracted above, because Height BEV naturally removes the most of ground points, highlighting fundamental structures and scene layouts well. To reduce memory space, keyframe strategy is introduced in this work, which is simply determined with the distances of adjacent frames. Feature maps and bag-of-words description are only constructed on keyframes, to improve the efficiency of pose estimation and global searching. For current frame, we search the bag of words for the 10 candidate frames with the highest similarity score.

2) Geometric Verification: For the candidate frames, a simple and effective geometric verification will be employed to remove outlier. Firstly, we will match each candidate frame to the current frame with ORB feature points, and use GMS to remove the outer points without geometric consistency. After that, we will keep the candidate frames with the highest matching feature points and above a certain threshold value for the next check. Let the covariance of the feature points P^c matched by the current frame be $C^c = \sum_{i=0}^{N} (p_i^c - \bar{p}^c)(p_i^c - \bar{p}^c)^T$, the covariance of the feature points P^h matched by the historical frame be $C^h = \sum_{j=0}^{M} (p_j^h - \bar{p}^h)(p_j^h - \bar{p}^h)^T$, and the matrix obtained by their difference be $D = C^c - C^h$. The final loop closing discriminant F_{loop} is defined as follows:

$$F_{loop} = \begin{cases} 0, |D(0,0)| + |D(1,1)| \geqslant D_{threshold} \\ 1, |D(0,0)| + |D(1,1)| < D_{threshold} \end{cases} \tag{4}$$

where, $D(0,0)$ represents the element in the first row and first column of matrix D, $D(1,1)$ represents the element in the second row and second column of matrix D, $D_{threshold}$ is the discrimination threshold, and when F_{loop} is equal to 1, it means that the candidate frame can be accepted for global optimization.

F. Global Mapping

Once the loop closure candidate is confirmed, we use the refine key points corresponding to the current frame and the surrounding map of the detected frame for an ICP registration. In this way, the relative transformation between the current frame and the detected frame can be obtained, which switch on the optimization of global pose graph. Specially, we build association of refined key points using visual feature matching, and then a simple replacement strategy is used for map updating. After

TABLE I
QUANTITATIVE RESULTS USING APE [M] ON ULHK DATASET

Sequence		ULHK 01			ULHK 02			ULHK 03			ULHK 05		
Trajectory Length		740m			230m			660m			600m		
Stats(m)		MAX	MEAN	RMSE	MAX	MEAN	RMSE	MAX	MEAN	RMSE	MAX	MEAN	RMSE
Methods	A-LOAM	2.804	1.159	1.355	3.106	1.314	1.508	4.886	1.452	1.845	52.669	11.029	16.155
	LeGO-LOAM [3]	9.660	1.581	2.329	**2.043**	**1.049**	**1.133**	7.915	2.608	3.025	3.179	0.918	1.092
	DLO [14]	3.658	1.542	1.794	11.117	4.802	5.646	9.189	2.065	2.658	**2.636**	1.035	1.222
	KISS-ICP [16]	14.286	2.477	3.437	8.726	3.664	4.163	8.835	2.828	3.256	4.852	1.679	1.957
	NDT	5.386	2.011	2.332	3.831	1.676	1.903	11.162	3.013	3.443	6.078	2.276	2.580
	NDT-LOAM [20]	2.498	1.129	1.349	3.128	1.296	1.516	**2.858**	1.563	1.699	2.987	1.103	1.219
	Ours	**2.497**	**1.093**	**1.185**	3.344	1.518	1.678	4.042	**1.251**	**1.513**	2.889	**0.875**	**1.060**

TABLE II
QUANTITATIVE RESULTS USING APE [M] ON KITTI AND GROUNDROBOT DATASET

Sequence		KITTI 00			KITTI 05			GroundRobot 01			GroundRobot 02		
Trajectory Length		3724m			2205m			863m			566m		
Stats(m)		MAX	MEAN	RMSE	MAX	MEAN	RMSE	MAX	MEAN	RMSE	MAX	MEAN	RMSE
Methods	A-LOAM	15.652	6.351	7.551	10.974	2.803	3.359	8.328	4.365	4.708	3.059	1.056	1.367
	LeGO-LOAM [3]	37.481	12.912	15.016	19.098	3.863	4.921	13.917	5.815	6.621	1.917	0.817	0.979
	DLO [14]	13.369	5.236	6.137	11.097	2.727	3.184	4.714	2.467	2.589	2.064	0.620	0.799
	KISS-ICP [16]	20.004	7.433	8.545	9.764	2.832	3.317	7.220	3.937	4.274	3.145	1.158	1.481
	NDT	55.642	18.661	22.608	25.115	7.602	8.704	10.965	6.443	6.793	4.003	1.869	2.042
	NDT-LOAM [20]	20.152	6.584	7.843	11.104	2.695	3.228	6.682	3.703	4.010	2.175	0.838	1.039
	Ours	**4.788**	**2.156**	**2.403**	**2.559**	**1.188**	**1.307**	**2.109**	**1.281**	**1.306**	**1.791**	**0.569**	**0.694**

the optimization, the current frame can be aligned into global map and the accumulated error can be eliminated to a certain extent. Thanks to the "visual" pipeline, the final global map only consists of key points corresponding to scene's fundamental structures, which is relatively lightweight compared with full point map.

IV. EXPERIMENTS

In this section, we verify the performance of BEV-LSLAM by experiments on public datasets and real environments. The classical methods LeGO-LOAM [3] and A-LOAM, the recently proposed methods DLO [14] (*RAL 2022*), KISS-ICP [16] (*RAL 2023*) and NDT-LOAM [20] are compared to present the competitiveness of our approach. All the algorithms are deployed on a computer equipped with an Intel i7-10875H CPU for testing, and the test environment is Ubuntu 20.04.

A. Dataset

We test our approach on three datasets: KITTI [27], UrbanLoco [28] and a self-collected dataset with 64-channel, 32-channel and 16-channel LiDARs respectively for verifying the generalization performance.

1) KITTI Dataset: KITTI dataset is a challenging benchmarks for stereo, optical flow, visual odometry/SLAM and 3D object detection, which is captured by driving their autonomous driving car around a mid-size city, in rural areas and on highways. Considering the lengths of sequences and the loop closing task, LiDAR scans captured with Velodyne HDL-64E in 00 and 05 sequences are used in this paper.

2) UrbanLoco-HK Dataset: UrbanLoco is a mapping/localization dataset collected in highly-urbanized environments. The Hong Kong sequences collected by HDL-32E in has a large number of buildings and dynamic objects, which is employed in our experiments.

3) GroundRobot Dataset: GroundRobot dataset is collected by our self-developed mobile robot with 16-channel LiDAR in campus environments. And the groundtruth are made by the high-precision RTK combine point cloud offline registration.

B. Evaluation of State Estimation

To validity the performance of our approach, we compare it with several state-of-the-art LiDAR-only methods on three datasets. The Absolute Pose Error (APE) of motion estimation are presented in Tables I and II, from which we can see that our approach achieves competitive performances on all data sequences. Specially, the performances of our approach on KITTI sequences are far better than other methods. The reason may be that KITTI sequences are captured driving on highway resulting in wider distribution of pointcloud, which is more suitable for detecting features in BEV view. Our approach also achieves promising performances on our 16-channel sequences. The reason is that the sparse pointclouds prevent from extracting stable geometrical features directly from raw LiDAR scans.

However, the performance of our approach is not outstanding on UrbanLoco HK02 sequence. The reason is that there are many high walls in UrbanLoco HK02 sequence, which prevents the scanning of surrounding environments. We should admit that our approach is limited in over structured environments with limited information projected in BEV view and severe uneven terrain where the violent rolling influences the consistency of projecting planes. In future work, we will seek the external sensors like imu and wheel encoder to enhance our work in challenging environments like long corridor and field uneven terrain.

C. Ablation Experiments

In this section, we conduct ablation experiments to prove the validity of BA operation with joint feature tracking among

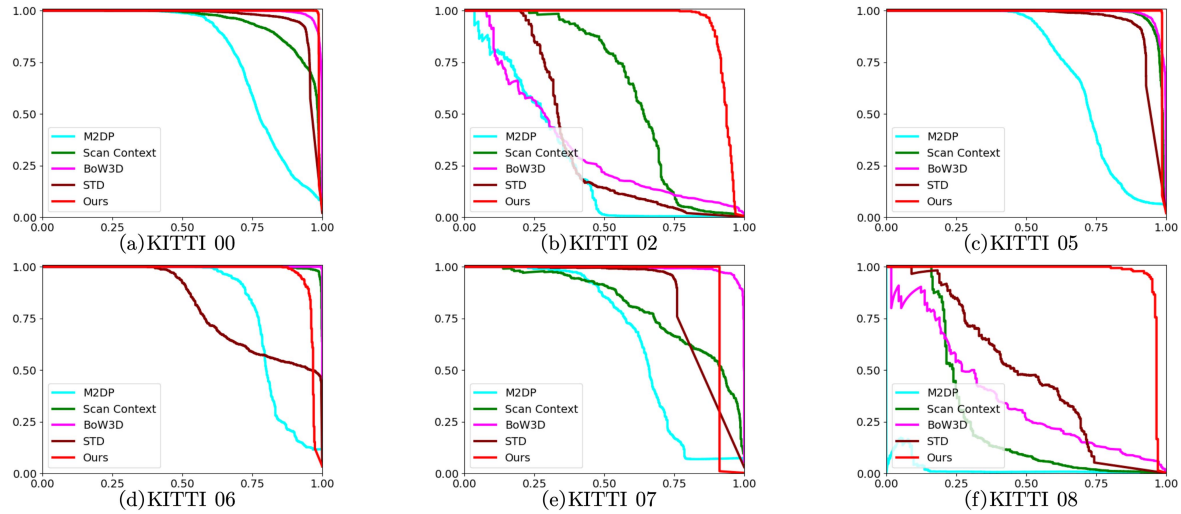

Fig. 7. Precision-Recall curves on KITTI dataset.

TABLE III
ERRORS OF SCAN-TO-SCAN TRACKING ON GROUNDROBOT

Sequence	GroundRobot 01		GroundRobot 02	
Stats(m)	MEAN	RMSE	MEAN	RMSE
A-LOAM	23.193	29.571	8.695	10.096
Ours(w/0 BA)	20.094	22.905	36.512	40.919
Ours(BA 5)	17.426	25.183	5.839	6.578
Ours(BA 10)	**14.918**	**22.179**	**5.741**	**6.414**

TABLE IV
MAXIMUM F1 SCORE (F1 MAX) [%] AND EXTENDED PRECISION (EP) [%] ON KITTI

Methods	00	02	05	06	07	08
M2DP [29]	74.7/69.7	37.0/51.8	69.3/68.5	80.3/77.4	65.4/61.3	11.0/50
SC [30]	85.7/69.6	64.3/61.6	94.5/85.8	98.3/93.8	72.0/56.9	32.8/58.1
BoW3D [31]	96.3/85.8	36.9/53.9	95.5/70.2	**99.8/99.8**	**96.1**/78.1	39.0/50.9
STD [32]	93.1/61.2	43.0/59.9	91.2/75.2	66.9/68.8	83.7/63.2	50.4/54.5
Ours	**98.9/98.4**	**91.8/88.4**	**99.2/99.2**	94.0/92.9	80.6/**80.9**	**94.9/90.1**

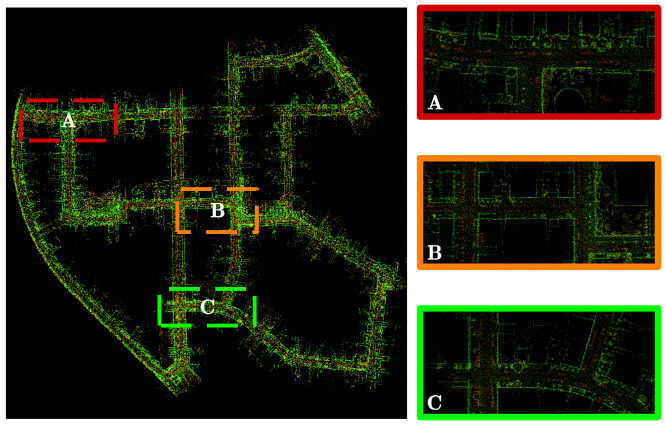

Fig. 8. Our lightweight global pointcloud map of KITTI 00 sequence and its local enlarged details.

TABLE V
ANALYSES OF RUNTIME, POINT DENSITY OF MAPPING AND LOCALIZATION ACCURACY ON KITTI 00 DATASET

Methods	Time costs(ms)	Mapping(points/m)	RMSE(m)
A-LOAM	261.34	1563.5	7.551
LeGO-LOAM	111.81	3103.7	15.016
DLO	45.08	1238.8	6.137
KISS-ICP	**30.92**	1595.0	8.545
NDT-LOAM	77.97	1364.2	7.843
Ours	60.59	**761.4**	**2.403**

adjacent frames. Since the LiDAR SLAM with sparse-channel sensors are more challenging, we conduct ablation experiments only on our 16-channel dataset. In the ablation experiment, we calculated the APE of A-LOAM front end, our tracking module without "BA" (*w/o BA*), our "BA" tracking module with 5 sliding window (*BA 5*), and our "BA" tracking module with 10 sliding window (*BA 10*). As shown in Table III, the "BA" operation with joint feature tracking improve the accuracy and robustness of motion estimation effectively, and the performance of "BA" tracking module increases with the width of the sliding window.

D. Loop Closing Results

To briefly present the performance of loop closing in our system, we compare our method with M2DP [29], ScanContext(SC) [18], BOW3D [30] and STD [31] on KITTI dataset. The Precision-Recall Curves are presented in Fig. 7, while Maximum F1 Score (F1 MAX) and Extended precision (EP) are presented in Table IV. The results prove our proposed loop closing module is competitive with baselines. In addition, the map detail of KITTI 00 showed in Fig. 8 proves that the lightweight feature points extracted by our method can generate a clear global feature map when combined with the reliable loop-close detection module.

E. Time Costs and Lightweight Mapping Analyses

The average time cost, point density of mapping and localization accuracy for each method on KITTI dataset is presented in Table V. Overall, our method achieves best performance on

lightweight mapping and localization accuracy while being able to run online over 10 Hz.

The average time cost of our method (60.59 ms) is made up of scan-to-scan tracking (36.27 ms) and scan-to-map mapping (24.32 ms). Though KISS-ICP and DLO are more efficient than our approach, their localization performances are poorer than ours, especially with sparse 16-channel LiDARs. Interestingly, owing to the feature-based scan-to-scan tracking and scan-to-map mapping, the resulting global pointcloud map is lightweight. It can be seen from Table V, the point density of our final global point cloud map only is 761.4 points per meter, which is far sparser than other methods. Although other methods can achieve similar densities after downsampling, this is often accompanied by a decrease in state estimation accuracy and map quality. The final global pointcloud map of KITTI 00 is shown in Fig. 8, in which the partial enlarged details present the fundamental structures well.

V. Conclusion

In this paper, a novel and compact LiDAR-only SLAM system BEV-LSLAM is proposed, which use visual features detected from BEV images of LiDAR scans as intermediaries for key point feature extraction. To exploit more geometrical structures of LiDAR scans, a BEV representation is proposed using geometrical angles among points to refine height values for pixel value computation. In scan-to-scan tracking, local BA operation with joint feature tracking is combined with feature matching to achieve robust motion estimation. To further improve the accuracy of odometry, scan-to-map registration is applied, in which only points projected into the same pillars with matched features are fed into the mapping module for high efficiency and lightweight mapping. Note visual features serve for state estimation, mappingm loop closing, and global map updating simultaneously, making the whole SLAM system compact. Finally, extensive experiments on KITTI, UrbanLoco and our 16-channel datasets proved the superiority of our approach compared with several state-of-the-art LiDAR-only SLAM methods.

References

[1] X. Yue, Y. Zhang, M. He, J. Chen, X. Zhou, and M. He, "LiDAR-based slam for robotic mapping: State of the art and new frontiers," *Ind. Robot*, vol. 51, no. 2, pp. 196–205, Jan. 2024.

[2] J. Zhang and S. Singh, "Low-drift and real-time LiDAR odometry and mapping," *Auton. Robots*, vol. 41, no. 2, pp. 401–416, Feb. 2017.

[3] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized LiDAR odometry and mapping on variable terrain," in *Proc. 2018 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 4758–4765.

[4] H. Wang, C. Wang, C-L Chen, and L Xie, "F-LOAM: Fast LiDAR odometry and mapping," in *Proc. 2021 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 4390–4396.

[5] H. Guo, J. Zhu, and Y. Chen, "E-LOAM: LiDAR odometry and mapping with expanded local structural information," *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1911–1921, Feb. 2023.

[6] J. Zhang and S. Singh, "Visual-LiDAR odometry and mapping: Low-drift, robust, and fast," in *Proc. 2015 IEEE Int. Conf. Robot. Automat.*, 2015, pp. 2174–2181.

[7] T. Sha, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled LiDAR inertial odometry via smoothing and mapping," in *Proc. 2020 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5135–5142.

[8] J. Lin, C. Zheng, W. Xu, and F. Zhang, "R ^{2} LIVE: A robust, real-time, LiDAR-inertial-visual tightly-coupled state estimator and mapping," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 7469–7476, Oct. 2021.

[9] W. Xu and F. Zhang, "FAST-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021.

[10] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[11] J. Behley and C. Stachniss, "Efficient surfel-based SLAM using 3D laser range data in urban environments," in *Proc. Robot.: Sci. Syst. XIV*, 2018, doi: 10.15607/RSS.2018.XIV.016.

[12] Y. Zhang et al., "RI-LIO: Reflectivity image assisted tightly-coupled LiDAR-inertial odometry," *IEEE Robot. Automat. Lett.*, vol. 8, no. 3, pp. 1802–1809, Mar. 2023.

[13] W. Du and G. Beltrame, "Real-time simultaneous localization and mapping with LiDAR intensity," in *Proc. 2023 IEEE Int. Conf. Robot. Automat.*, 2023, pp. 4164–4170.

[14] K. Chen, B. T. Lopez, A. -A. Agha-mohammadi, and A. Mehta, "Direct LiDAR odometry: Fast localization with dense point clouds," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2000–2007, Apr. 2022.

[15] P. Dellenbach, J. -E. Deschaud, B. Jacquet, and F. Goulette, "CT-ICP: Real-time elastic LiDAR odometry with loop closure," in *Proc. 2022 IEEE Int. Conf. Robot. Automat.*, 2022, pp. 5580–5586.

[16] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "KISS-ICP: In defense of point-to-point ICP - simple, accurate, and robust registration if done the right way," *IEEE Robot. Automat. Lett.*, vol. 8, no. 2, pp. 1029–1036, Feb. 2023.

[17] H. Wang, C. Wang, and L. Xie, "Intensity-SLAM: Intensity assisted localization and mapping for large scale environment," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 1715–1721, Apr. 2021.

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

[19] X. Chen, A. Milioto, E. Palazzolo, P. Giguère, J. Behley, and C. Stachniss, "SuMA: Efficient LiDAR-based semantic SLAM," in *Proc. 2019 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4530–4537.

[20] S. Chen et al., "NDT-LOAM: A real-time LiDAR odometry and mapping with weighted NDT and LFA," *IEEE Sensors J.*, vol. 22, no. 4, pp. 3660–3671, Feb. 2022.

[21] Y. Pan, P. Xiao, Y. He, Z. Shao, and Z. Li, "Mulls: Versatile LiDAR SLAM via multi-metric linear least square," in *Proc. 2021 IEEE Int. Conf. Robot. Automat.*, 2021, pp. 11633–11640.

[22] K. Koide, M. Yokozuka, S. Oishi, and A. Banno, "Voxelized GICP for fast and accurate 3D point cloud registration," in *Proc. 2021 IEEE Int. Conf. Robot. Automat.*, 2021, pp. 11054–11059.

[23] D. Galvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Trans. Robot.*, vol. 28, no. 5, pp. 1188–1197, Oct. 2012.

[24] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[25] J. Bian, W. -Y. Lin, Y. Matsushita, S. -K. Yeung, T. -D. Nguyen, and M. -M. Cheng, "GMS: Grid-based motion statistics for fast, ultra-robust feature correspondence," in *Proc. 2017 IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2828–2837.

[26] F. Cao, H. Wu, and C. Wu, "Application of sparse-channel LiDAR sensors on viewpoint-invariant loop closing task," *IEEE Sensors J.*, vol. 22, no. 14, pp. 14592–14600, Jul. 2022.

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

[28] W. Wen et al., "UrbanLoco: A full sensor suite dataset for mapping and localization in urban scenes," in *Proc. 2020 IEEE Int. Conf. Robot. Automat.*, 2020, pp. 2310–2316.

[29] L. He, X. Wang, and H. Zhang, "M2DP: A novel 3D point cloud descriptor and its application in loop closure detection," in *Proc. 2016 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 231–237.

[30] Y. Cui, X. Chen, Y. Zhang, J. Dong, Q. Wu, and F. Zhu, "BoW3D: Bag of words for real-time loop closing in 3D LiDAR SLAM," *IEEE Robot. Automat. Lett.*, vol. 8, no. 5, pp. 2828–2835, May 2023.

[31] C. Yuan, J. Lin, Z. Zou, X. Hong, and F. Zhang, "STD: Stable triangle descriptor for 3D place recognition," in *Proc. 2023 IEEE Int. Conf. Robot. Automat.*, 2023, pp. 1897–1903.