Unified Mapping Framework for Multi-modal LiDARs in Complex and Dynamic Environments

Gilhwan Kang[1], Hogyun Kim[1], Byunghee Choi[1], Seokhwan Jeong[1], Young-Sik Shin[2] and Younggun Cho[1]

***Abstract*— In construction sites, which are large-scale and complex urban environments, it is essential to merge multiple maps obtained from various platforms. However, sensor-modality and dynamic environments remain challenging problems for unified mapping. To address this issue, we present *Uni-Mapper*, a dynamic-aware 3D point cloud map merging framework for multi-modal LiDAR systems. Our scene descriptor rejects dynamic objects in real-time and is robust to LiDAR modality based on local triangle features. To ensure consistent mapping performance, we adopt centralized pose graph optimization with a two-step registration process. We thoroughly evaluate the superiority of the proposed framework using two datasets: HeLiPR (multi-modal) and INHA (multi-modal, multi-robot).**

I. Introduction and Related Works

Light Detection and Ranging (LiDAR) is essential for robot navigation, offering accuracy and wide coverage. As sensor technology advances, various types of LiDAR with unique Field Of View (FOV) and scan patterns have developed, necessitating careful selection for effective mapping. For instance, autonomous heavy equipment prefers omnidirectional LiDAR for safety, while lightweight LiDAR suits agile quadrupedal robots. Although large-scale mapping of diverse terrains has made the need for multi-robot mapping, the distinct configuration of each LiDAR becomes more challenging for multiple map alignment. Furthermore, dynamic objects frequently present in construction sites cause an undesired traces on maps as known as the ghost trail effect problem. Eliminating these dynamic objects is crucial for effective mapping in varied environments.

To address these challenges, ongoing researches focus on place recognition (PR) [1–4], Dynamic Object Removal (DOR) [5–7] and Map Merging (MM) [8]. However, recent approaches struggle to handle variations across LiDAR types, and the majority of DOR is used for offline post-processing. Furthermore, there are few publicly open map merging frameworks that account for both multi-modal LiDAR and dynamic environments.

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (RS-2023-00302589) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00448).

[1]Gilhwan Kang, [1]Hogyun Kim, [1]Byunghee Choi, [1]Seokhwan Jeong and [1]*Younggun Cho is with the Electrical and Computer Engineering, Inha University, S. Korea [rlfghks527, hg.kim, bhbhchoi, eric5709]]@inha.edu, yg.cho@inha.ac.kr

[2]Young-sik Shin is with Depart. of AI Machinery, Korea Institute of Machinery and Materials, Daejeon, South Korea yshin86@kimm.re.kr

Corresponding Authors: Young-Sik Shin and Younggun Cho

In this paper, we introduce *Uni-Mapper*, a LiDAR-modality-agnostic, dynamic-aware map merging framework. This framework facilitates robust MM in dynamic environments through dynamic-aware triangle descriptor and multiple pose graph optimization. The DOR module employs voxel representation using a coarse-to-fine strategy for achieving static maps. Additionally, it integrates with the stable triangle descriptor (STD) [1], a state-of-the-art (SOTA) LiDAR descriptor for 3D PR. Our proposed *DynaSTD* descriptor is capable of detecting loops across diverse LiDAR types and robot platforms by leveraging local geometric features. From a MM perspective, we successively optimize intra-session and multi-map pose graphs and employ anchor factors to align multiple maps to a unified coordinate system. To summarize, our contributions are as follows:

- We propose a novel dynamic-aware and LiDAR-modality-agnostic MM framework for multi-session and multi-robot scenarios. Our system simultaneously estimates dynamic points and unifies multiple maps in a single process.
- *DynaSTD*, a global descriptor which integrated with DOR module is proposed. As DynaSTD is built from a combination of local descriptors, it demonstrates robust performance to LiDAR modality and dynamic environments.
- The proposed framework is evaluated through different types of LiDAR and platforms on public and custom datasets including a number of dynamics. We demonstrate that our unified framework shows robust performance in dynamic environments and cross-modal LiDAR sensors.

II. Method

The system is built upon two main modules: dynamic-aware scene description and multi-map merging, as shown in Fig. 1.

A. Dynamic-aware scene description

1) Online dynamic object removal: As reported in Lim et al. [7], most dynamic objects are classified as non-ground points $\mathbf{P}^{ng}$ situated on the ground points $\mathbf{P}^{g}$. Our DOR approach is to segment free space based on the ground prior and identify dynamic points that exist in free space.

For ground segmentation, LiDAR scans are accumulated into a dense keyframe point cloud $\mathbf{P}$ to compensate for the sparsity issue. $\mathbf{P}$ is voxelized with coarse leaf size L_c (e.g. 2m) and voxel-wise planarity is checked by Principal Component Analysis (PCA). Ground voxels $\mathcal{G}$ are selected

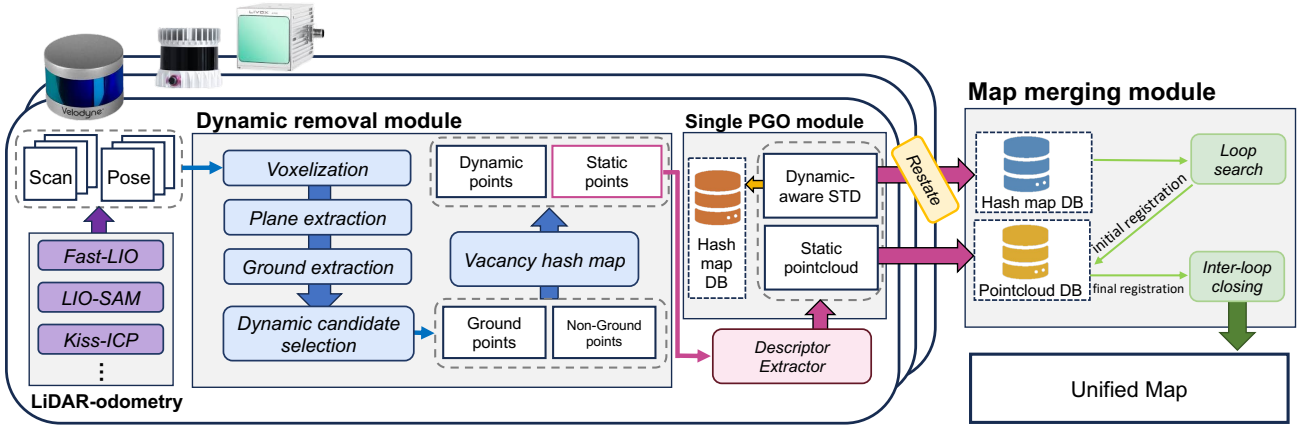

Fig. 1: Our proposed framework for dynamic-aware LiDAR-agnostic multiple map merging.

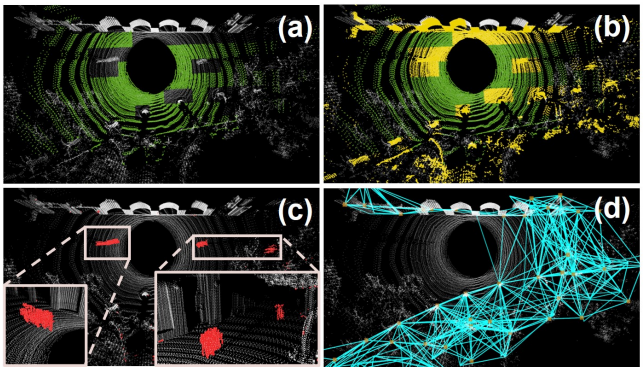

Fig. 2: Four steps of *DynaSTD*. (a) Ground estimation (green), (b) dynamic candidates selection (yellow), (c) dynamic points segmentation (red), and (d) dynamic-aware descriptor (blue)

from plane voxels by calculating relative angle with plane normal and robot uprightness similar with [7]. As shown in Fig. 2, we set dynamic candidate voxels $\mathcal{C}$ as neighboring voxels to $\mathcal{G}$, which dynamic objects can be present.

Since $\mathcal{C}$ consists of partial ground region, we additionally segment ground points in candidate voxel $\mathbf{P}_C^g$ using neighboring ground information as:

$$\mathbf{P}_C^g = \left\{ \mathbf{p} \in \mathbf{P}_C \mid \mathbf{n}_a^T (\mathbf{p} - \mathbf{p}_a) < m_{th} \right\}, \tag{1}$$

where $\mathbf{n}_a$, $\mathbf{p}_a$ and m_{th} as average plane normal, average central point of adjacent ground voxels, and margin threshold, respectively. For detailed estimation, we subdivide $\mathcal{C}$ into finer voxels, using a smaller leaf size L_f (e.g. 0.2m) and refer to the combined coarse-to-fine voxel map.

Instead of relying on ray-casting-based occupancy updates [5] which require heavy computation, we simplify the free space estimation problem to segment vacant space extending from ground height h^g to the lowest non-ground voxel z^m. In the case of $\mathcal{G}$, which has no objects above, we set z^m as *infinity*, and all spaces above ground are regarded as free space for efficiency.

We utilize sliding windows of free space hash maps to estimate consistent vacant regions. By adopting recursive binary bayes filter, the local free space submap denoted as $\mathcal{E}$ is determined where the majority of temporal voxels are vacant. Finally, $\mathbf{P}$ is classified into static and dynamic points as:

$$\mathbf{p} = \begin{cases} \mathbf{p}^{\text{dyn}}, & \text{if } \exists \mathbf{p} \in \mathcal{E} \\ \mathbf{p}^{\text{sta}}, & \text{else.} \end{cases} \tag{2}$$

2) Dynamic-aware scene description: Based on our online DOR module, we integrate this module with STD[1]. As global descriptor $\mathbf{d}$ is composed of local geometric triangle descriptors Δ, it facilitates loop detection which has only partial overlaps across varied LiDAR FOV and can estimate relative pose only using $\mathbf{d}$ which is efficient than conventional scan matching algorithms. However, dynamic objects identified as key points act as outliers which degrade PR performance. Removing dynamic elements enables the generation of robust global descriptors suitable for highly dynamic environments.

B. Multiple pose graph optimization

1) intra-session pose graph optimization: As estimated trajectory $\mathbf{X}$ from LiDAR odometry contains drift errors from sensor measurement and processing noise, loop closing is essential to correct undesired effects. Intra-session pose graph optimization of each maps are represented as:

$$\hat{\mathbb{X}} = \underset{\mathbb{X}}{\operatorname{argmin}} \sum_{m \in \mathbb{N}} \mathcal{F}(\mathbf{X}^m), \tag{3}$$

where $\hat{\mathbb{X}}$ and $\mathbb{N}$ represent the optimized trajectory set of multiple maps and map index set, respectively. Single pose graph $\mathcal{F}(\cdot)$ is defined as:

$$\begin{aligned} \mathcal{F}(\mathbf{X}) = & \sum_t \left\| f\left(\mathbf{x}_t, \mathbf{x}_{t+1}\right) - \hat{\mathbf{z}}_{t,t+1} \right\|^2_{\sum_t} \\ & + \sum_{\langle i,j \rangle \in \mathbb{L}^{IS}} \rho \left(\left\| f\left(\mathbf{x}_i, \mathbf{x}_j\right) - \mathcal{S}(\mathbf{d}_i, \mathbf{d}_j) \right\|^2_{\sum_{i,j}} \right), \end{aligned} \tag{4}$$

where function $f(\cdot)$ is relative pose calculator, $\hat{\mathbf{z}}$ is estimated odometry, and $\mathbb{L}^{IS}$ is intra-session loop pairs by PR. $\mathcal{S}(\cdot)$ denote relative pose estimator based on descriptor. As $\mathcal{F}(\cdot)$ consists of an intra-session factor graph of each map, pose graphs are optimized independently. To handle incorrect loop constraints, we use a Cauchy distribution-based robust estimator $\rho(\cdot)$.

2) Multi-modal LiDAR map merging: Our multi-map merging module employs a centralized approach to optimize multiple pose graphs. To resolve intra-session drift errors while aligning multiple maps, we adopt anchor-node based pose graph optimization [9] as:

$$\mathbb{X}^* = \underset{\hat{\mathbb{X}}}{\operatorname{argmin}} \Big\{ \sum_{m \in \mathbb{N}} \mathcal{F}(\hat{\mathbf{X}}^m) + \sum_{Q \in \mathbb{Q}} \mathcal{A}(\hat{\mathbf{X}}^C, \hat{\mathbf{X}}^Q) \Big\}, \quad (5)$$

where $\mathbb{X}^*$ is a trajectory set aligned with the unified coordinate of the central map, C is central map and $\mathbb{Q}$ is query map set. Anchor node based factor graph $\mathcal{A}(\cdot)$ and anchor factor $\Phi(\cdot)$ are represented as:

$$\mathcal{A}(\hat{\mathbf{X}}^C, \hat{\mathbf{X}}^Q) = \sum_{\langle i,j \rangle \in \mathbb{L}^{IM}} \left\| \Phi\left(\hat{\mathbf{x}}_i^C, \hat{\mathbf{x}}_j^Q, \delta^C, \delta^Q\right) - \hat{\mathbf{z}}_{i,j} \right\|^2_{\sum_{i,j}} \quad (6)$$

$$\Phi\left(\mathbf{x}_i^C, \mathbf{x}_j^Q, \delta^C, \delta^Q\right) = \left(\left(\delta^C \oplus \mathbf{x}_i^C\right) \ominus \left(\delta^Q \oplus \mathbf{x}_j^Q\right) \right) \quad (7)$$

where $\mathbb{L}^{IM}$ is inter-map loop pairs and SE(3) pose operators are denoted as $\oplus$ and $\ominus$. Anchor node δ represents the relative transformation between each map's origin coordinate systems to the centralized target system. By allocating large covariance to query anchor nodes, all query anchor nodes are aligned to the central map to minimize measurement errors. For inter-map constraints $\hat{\mathbf{z}}_{ij}$, we utilize $\mathcal{S}(\cdot)$ as an initial map alignment and radius-search-based loop closing as a refined registration. G-ICP is adopted for refined registration to achieve a better precision in map unification.

III. Experimental Results

For the concrete validation, we tested the proposed method on multi-modal LiDAR datasets as follows: `TOWN1-2` sequences of HeLiPR [10], and `WHEEL`, `DOG`, and `HAND` sequences of our own dataset called INHA. A summary of each data sequence is provided in Table I. While HeLiPR provides ground truth trajectories, we manually constructed the ground truth using interactive SLAM [11] for INHA sequence. Also, we utilized FAST-LIO2 [12] to generate single-session odometry for both datasets. As both dataset lack of point-wise semantic information, we manually label dynamic points for `Town2` and `WHEEL` sequences to evaluate DOR performance.

TABLE I: Datasets Summary

Name	Complexity	Robot	LiDAR	FOV (H × V)
`TOWN1`	★ ★ ★	Vehicle	Ouster OS2-128	360° × 22.5°
`TOWN2`	★ ★ ★		Aeva Aeries	120° × 19.2°
`WHEEL`	★ ★	Mobile	Ouster OS1-64	360° × 45°
`DOG`	★ ★ ★	Quadruped	Velodyne VLP-16	360° × 30°
`HAND`	★	Hand-held	Livox Avia	70° × 77°

A. Evaluation on dynamic-aware scene description

To verify the necessity of a dynamic-aware and LiDAR-modal-agnostic description, we evaluate both DOR and PR performance. Table. II and Fig. 3 shows the quantitative and qualitative result of our DOR module compared to Removert [6]. In both datasets, Table. II shows competitiveness of our methods on static mapping (SA) and better performance on DOR (DA), which results in balanced performance (AA) while Removert depends on prior map data. In addition, our method demonstrates best efficiency in runtime.

TABLE II: Quantitative evaluation of dynamic removal

Seq	Methods	SA ↑	DA ↑	AA ↑	Runtime[ms]
`TOWN2`	Removert	**99.78**	24.38	49.22	58
	Ours	99.45	**66.36**	**81.24**	**23**
`WHEEL`	Removert	**98.85**	78.33	87.99	89
	Ours	91.87	**93.86**	**92.86**	**10**

By removing dynamic objects online in the scene description, our method generates a scene descriptor robust to dynamic environments. Fig. 4 represents the true positive (green) and false positive (red) matching pairs under maximum precision conditions. As shown in Fig. 4, our approach consistently exhibits superior loop detection performance compared to other methods. Global descriptors [2–4] which encode 3D spatial information into a low-dimensional space are susceptible to the different FOV of LiDAR. Conversely, local feature-based techniques like STD show robustness to cross-modality. However, STD tends to show lower performance in dynamic environments. In Fig. 4, the dashed blue boxes in ours represent the locations identified as a loop, unlike the dashed yellow boxes of STD.

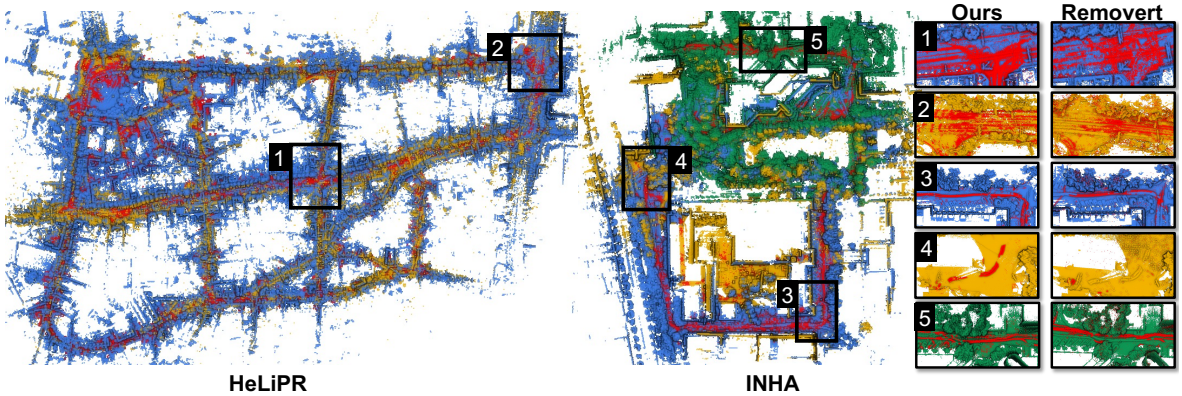

Fig. 3: Qualitative results of our framework. Colored pointcloud represent merged maps between multi-modal LiDARs. The red points indicate dynamic objects. The pair of colors and sequence is represented as follows. HeLiPR: OS2-128 (blue), Aeva (yellow) / INHA: OS1-64 (blue), AVIA (yellow) and VLP16 (green).

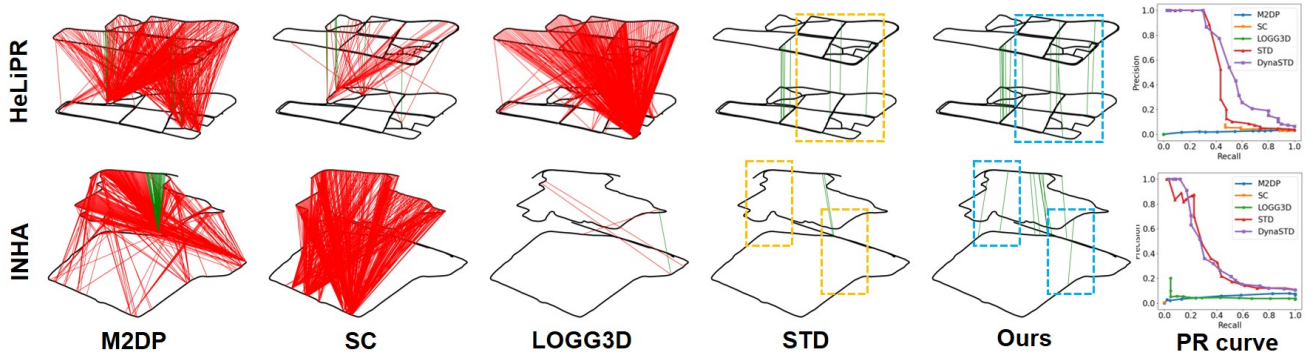

Fig. 4: The place recognition results for multi-modal datasets under maximum precision. In each pair, the lower and upper trajectories represent base and query maps, respectively. Each map pair is as follows. HeLiPR: `TOWN1`(base), `TOWN1`(query) / INHA: `WHEEL`(base), `HAND`(query).

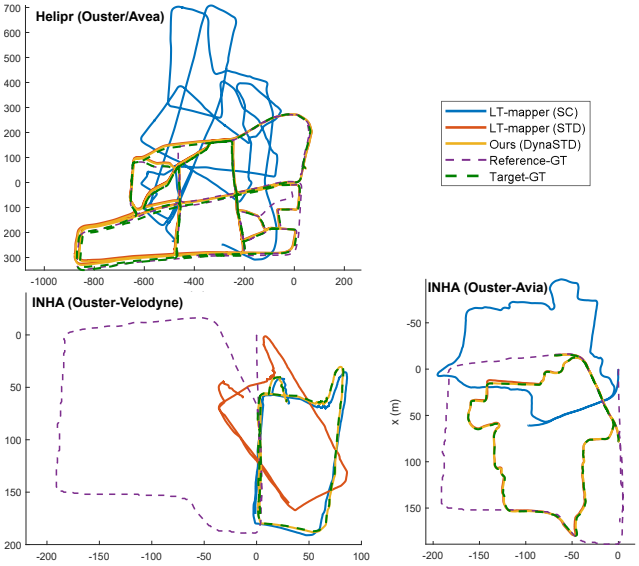

Fig. 5: Visualization of the trajectory after map alignment

TABLE III: Absolute Trajectory Error (m)

Sequence	LT-mapper (SC)	LT-mapper (STD)	Ours
HeLiPR (Ouster-Avea)	850.65	22.07	**17.98**
INHA (Ouster-Velodyne)	6.90	45.29	**0.81**
INHA (Ouster-Avia)	122.70	0.59	**0.35**

B. Map Merging and Alignment Evaluation

We evaluated the map merging performance comparing our method (*Uni-Mapper*) with the Scan Context (SC) based LT mapper (original) and the STD-based LT Mapper (Customized). Fig. 5 and table III represent the qualitative and quantitative performance of multiple map alignment. We utilize the base session from Ouster and the query session from Avea for HeLiPR and Velodyne/Avia for INHA. LT-mapper (SC) fails to align multiple maps from the different modalities of LiDARs, and LT-mapper (STD) suffers from dynamic objects that cause false matching and incorrect registration in INHA sequences (Ouster-Velodyne). On the other hand, our framework demonstrates robust map merging performance to both LiDAR modality and dynamic objects. In ATE criteria, our method shows an 87% and 32% improvement compared with the LT-mapper (SC) and the LT-mapper (STD), respectively.

IV. Conclusion

This paper presents *Uni-Mapper*, a multi-map merging framework that incorporates a dynamic-aware global descriptor for complex environments. This framework efficiently filters out dynamic objects online, while capturing structural information from various types of LiDAR using a local keypoint combination. Ultimately, *Uni-Mapper* successfully achieves an accurate unified map through a two-step registration process and shows superior map alignment performance robust to LiDAR modality and dynamic objects.

References

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

[2] 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 Intl. Conf. on Intell. Robots and Sys.* IEEE, 2016, pp. 231–237.

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

[4] 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. on Robot. and Automat.* IEEE, 2022, pp. 2215–2221.

[5] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *Autonomous robots*, vol. 34, pp. 189–206, 2013.

[6] G. Kim and A. Kim, "Remove, then revert: Static point cloud map construction using multiresolution range images," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.* IEEE, 2020, pp. 10 758–10 765.

[7] H. Lim, S. Hwang, and H. Myung, "Erasor: Egocentric ratio of pseudo occupancy-based dynamic object removal for static 3d point cloud map building," *IEEE Robot. and Automat. Lett.*, vol. 6, no. 2, pp. 2272–2279, 2021.

[8] G. Kim and A. Kim, "Lt-mapper: A modular framework for lidar-based lifelong mapping," in *Proc. IEEE Intl. Conf. on Robot. and Automat.* IEEE, 2022, pp. 7995–8002.

[9] B. Kim, M. Kaess, L. Fletcher, J. Leonard, A. Bachrach, N. Roy, and S. Teller, "Multiple relative pose graphs for robust cooperative mapping," in *Proc. IEEE Intl. Conf. on Robot. and Automat.* IEEE, 2010, pp. 3185–3192.

[10] M. Jung, W. Yang, D. Lee, H. Gil, G. Kim, and A. Kim, "Helipr: Heterogeneous lidar dataset for inter-lidar place recognition under spatial and temporal variations," https://sites.google.com/view/heliprdataset, year=2023.

[11] K. Koide, J. Miura, M. Yokozuka, S. Oishi, and A. Banno, "Interactive 3d graph slam for map correction," *IEEE Robot. and Automat. Lett.*, vol. 6, no. 1, pp. 40–47, 2020.

[12] 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, 2022.