IEEE Robotics and Automation Letters (RAL) paper presented at
2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

LiDAR-Link: Observability-Aware Probabilistic Plane-Based Extrinsic Calibration for Non-Overlapping Solid-State LiDARs

Jie(Rover) Xu, Song Huang, Shuxin Qiu, Lijun Zhao,
Wenlu Yu, Mingxing Fang, Minhang Wang, Ruifeng Li

***Abstract*—As solid-state LiDAR technology advances, mobile robotics and autonomous driving increasingly rely on multiple solid-state LiDARs for perception. However, limited or non-overlapping fields of view (FoV) among these sensors pose significant challenges for extrinsic calibration. Moreover, there are no quantitative indicators to evaluate calibration quality currently. To tackle these challenges, we introduce LiDAR-Link, a novel approach for calibration and evaluation, consisting of LiDAR-Bridge and LiDAR-Align. LiDAR-Bridge uses a wide-angle LiDAR to act as an intermediary, aligning point clouds from small-angle solid-state LiDARs. These small LiDARs have limited or non-overlapping FoVs, allowing us to indirectly determine the extrinsics between them. LiDAR-Align adaptively constructs a Voxel Probabilistic Plane Map to efficiently match point clouds and differentiate the contributions of various high-quality planar features. To align these point clouds, Iterated Extended Kalman Filter (IEKF) is utilized, which is based on point-to-plane residuals. The resulting covariance verifies whether point cloud alignment constraints are met and assesses calibration reliability. Furthermore, LiDAR-Align supports multi-scene joint calibration to overcome the limitations of fewer constraints caused by smaller LiDAR FoVs. We validate the accuracy of our method and the significance of observability analysis through a comprehensive set of experiments.**

***Index Terms*—Observability-aware, solid-state LiDAR, non-overlapping, extrinsic calibration, probabilistic plane, IEKF.**

I. Introduction

Manuscript received: July, 24, 2023; Revised November, 14, 2023; Accepted December, 28, 2023.

This paper was recommended for publication by Editor Lucia Pallottino upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported in part by the National Natural Science Foundation of China under Grant 62073101 and in part by the Self-planned Task of State Key Laboratory of Robotics and System (HIT) under Grant SKLRS202007B.

Jie(Rover) Xu and Song Huang contribute equally to the article. Lijun Zhao and Ruifeng Li are the common corresponding authors.

Jie(Rover) Xu, Wenlu Yu, Lijun Zhao, and Ruifeng Li are with State Key Laboratory of Robotics and Systems, Harbin Institute of Technology, Harbin 150001, China (e-mail: jeff_xu_0503@foxmail.com; yuwenlu0720@qq.com; zhaolj@hit.edu.cn; lrf100@hit.edu.cn).

Jie (Rover) Xu, Song Huang, Shuxin Qiu, and Lijun Zhao are with Yangtze River Delta HIT Robot Industry Technology Research Institute, Wuhu 241000, China (email:mmerit31@163.com; qsx97@foxmail.com).

Song Huang and Mingxing Fang are with Anhui Provincial Engineering Laboratory on Information Fusion and Control of Intelligent Robot, Anhui Normal University, Wuhu 241000, China (e-mail: mmerit31@163.com; mxfang@ahnu.edu.cn).

Song Huang and Minhang Wang are with HAOMO.AI Technology Co., Ltd, Beijing 100192, China(wangminhang@haomo.ai)

Digital Object Identifier (DOI): see top of this page.

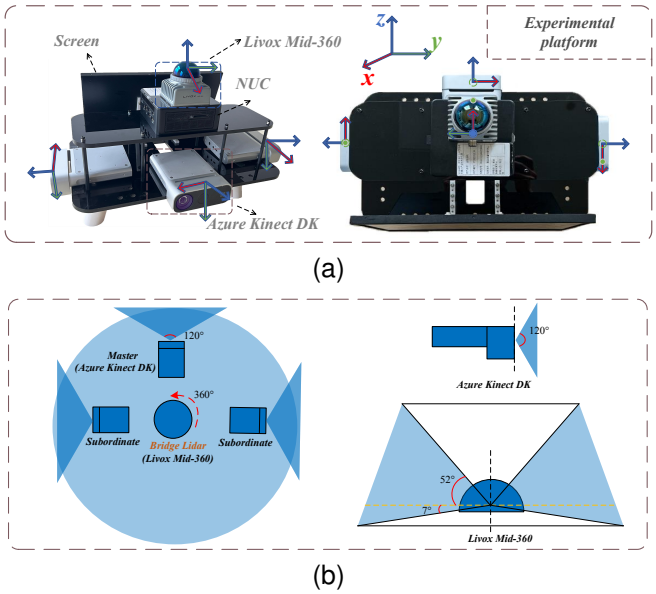

Fig. 1. Our experimental platform "Three-Eyed Raven". (a) The experimental setup consists of 3 Azure Kinect DK ToF cameras (LiDAR) and 1 Livox Mid-360. Our goal is to calibrate the extrinsics between the three Azure Kinect DK LiDARs. (b) The FoVs of the Azure Kinect DK and Livox Mid-360. The Livox Mid-360 serves as the bridge LiDAR and ultimately aims to align the point clouds from the two subordinate LiDARs to the coordinate system of the master LiDAR, utilizing the extrinsics obtained through calibration.

THE 3D Light Detection and Ranging (LiDAR) sensor has found a wide range of applications use in various domains, including mobile robotics [1], drones [2], and autonomous driving [3]. Additionally, solid-state LiDARs are more cost-effective, provide higher resolution, and have smaller dimensions and weight when compared to traditional mechanical LiDARs. However, solid-state LiDARs have a narrower field of view (FoV), which requires the use of multiple sensors to achieve complete omnidirectional coverage. Consequently, accurate extrinsic calibration becomes a prerequisite. Extrinsic calibration allows for the determination of transformations between LiDAR frames, facilitating the conversion of information from multiple LiDARs into a unified coordinate system.

Time-of-Flight (ToF) technology measures the time it takes for light or electromagnetic waves to travel from the transmitter to the receiver to determine the range. Both ToF depth cameras and solid-state LiDARs utilize this technology for measuring distances. Consequently, we will treat ToF depth

cameras as being comparable to solid-state LiDARs in the subsequent sections.

A. Motivations

(a) Several challenges arise in the extrinsic calibration involving small FoV LiDARs. The main challenges include limited or no overlap between FoVs of sensors and high-precision requirements. Hand-eye calibration [4] for extrinsic calibration can cause considerable errors in translation. Calibration using feature-based methods [5]–[8] can achieve high accuracy, but it requires all sensors to have a shared FoV, which may not be feasible in scenarios like autonomous driving, where multiple small FoV solid-state LiDARs are used with minimal or no overlap. However, the method also can rotate the LiDARs to observe the same features at different times, creating a historical common FoV and satisfying the usage requirements for feature-based methods. Nevertheless, this method is susceptible to cumulative errors and involves complex operations.

(b) Feature-based calibration is commonly achieved through the use of the Iterative Closest Point (ICP) algorithm [9] and its variations. By minimizing the Euclidean distance, this algorithm achieves optimal matching between two point clouds or between a point cloud and a locally fitted plane on a target point. The iterative process identifies optimal matches and continuously improves the alignment result until convergence is achieved for optimal alignment. However, ICP is prone to instability when dealing with smooth surfaces, resulting in alignment drift [10], [11] if the optimization lacks adequate constraints from geometric features. This, in turn, can lead to misaligned outcomes. Therefore, it is crucial to ensure sufficient constraints on the six degrees of freedom of the extrinsics when aligning point clouds, as these parameters significantly impact calibration accuracy.

To address the aforementioned issues, this paper proposes a calibration strategy called LiDAR-Link, comprising two parts: LiDAR-Bridge and LiDAR-Align. To address problem **(a)**, LiDAR-Bridge employs a large FoV and high-resolution LiDAR as a bridge to create a "common view" for other LiDARs indirectly. By aligning the point clouds from the bridge LiDAR and other LiDARs to a common view, the coordinate transformation between them can be obtained, and the extrinsics between LiDARs without a common view can be calculated. To address problem **(b)**, LiDAR-Align first performs coarse initial alignment on each point cloud using Iterated Extended Kalman filter (IEKF). Then, it identifies pairs of point clouds with sufficient constraints via observability analysis. Finally, it jointly calculates the extrinsics using the Generalized-ICP [12] algorithm for the point cloud pairs from multiple scenes.

B. Contributions

Our work is motivated by the extensive research on feature-based calibration for the high accuracy. The contributions of our study are as follows:

- We propose LiDAR-Bridge, an innovative calibration strategy specifically devised for LiDAR sensors with limited or non-overlapping FoV. This strategy employs a wide FoV LiDAR as a bridge, thus facilitating the indirect calibration of LiDARs bereft of a common FoV.
- We introduce a novel calibration method, named LiDAR-Align, for LiDARs with overlapping FoV. This method is coarse-to-fine and observability-aware, utilizing IEKF to align point clouds by constructing a voxel probabilistic plane map. Additionally, we use the eigenvalues of the covariance as an evaluation metric for calibration quality and perform joint fine calibration using Generalized-ICP for all point cloud pairs in parallel spaces.
- Our method has been rigorously validated through a series of experiments, which have collectively affirmed the effectiveness of using covariance as a key indicator for evaluating the appropriateness of calibration constraints. These experimental findings also confirm the high precision of our calibration method, LiDAR-Link, further cementing its accuracy.

II. Related Works

Extrinsic calibration is a fundamental process for accurately integrating data from multiple sensors and aligning them in a common coordinate system. This involves determining the rigid transformations, encompassing 3D rotation and translation, between the coordinate frames of the sensors. Two types of extrinsic calibration methods are commonly used for LiDAR sensors: (1) hand-eye calibration, which estimates the sensors' relative pose by tracking the trajectory of independent or fused sensor information; (2) feature-based methods, which require specific targets with clear geometric features.

Hand-eye approach. Extrinsic calibration using the hand-eye method involves establishing the relative relationship between the trajectories of sensors. This is accomplished by subjecting each sensor to the same rigid body motion [11], [13]–[15] within a specific time interval and transforming the problem into a hand-eye calibration problem [4]. Jiao et al. [16] proposed a method that utilizes inter-frame matching with edge and plane features to obtain the odometers of each LiDAR, followed by constructing a hand-eye calibration problem to determine the initial extrinsics between the LiDARs. They ensured that identical rigid motion occurred between sensors within the same time interval. Taylor et al. [17] demonstrated the use of hand-eye calibration to determine the extrinsics and temporal offsets between sensors by exploiting the motion of the system. Qin et al. [18] designed an online method for calibrating the extrinsics between the camera and the IMU. Das et al. [19] presented a method that utilizes GNSS reference poses to achieve real-time calibration of multiple LiDAR sensors based on motion. However, the hand-eye calibration method requires adequate motion in all degrees of freedom, which is challenging to achieve for LiDARs mounted on autonomous driving cars or mobile robots in their vertical direction. Additionally, the calibration results have low translational accuracy.

Feature-based approach. The feature-based approach for calibrating LiDARs can utilize either natural environmental features, such as edges and planes, or artificial markers that

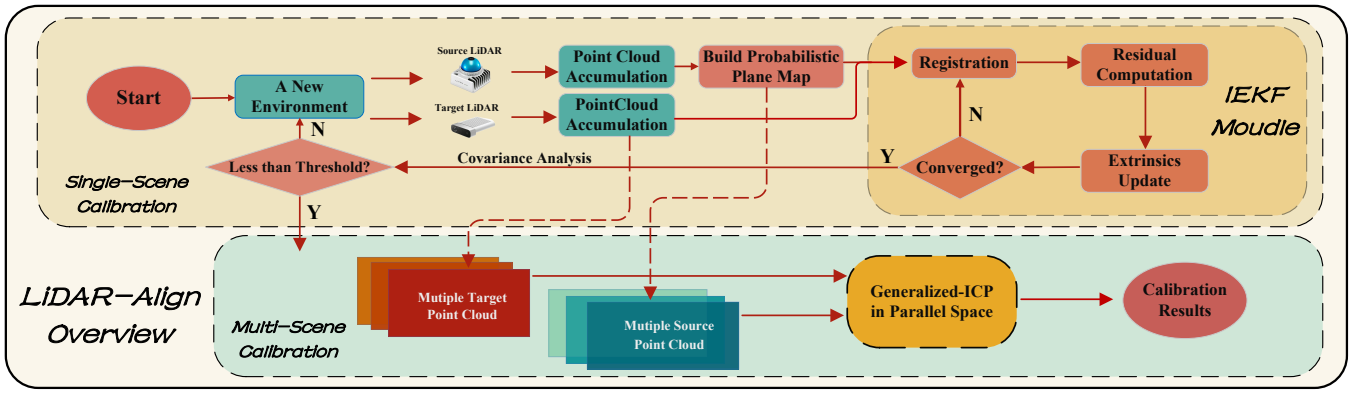

Fig. 2. System overview of LiDAR-Align.

provide additional features for extrinsic calibration. Xue et al. [20] achieved automatic extrinsic calibration between LiDARs using two non-parallel poles affixed with retro-reflective tape stickers. However, the recognition of the poles relies on the intensity information of the LiDARs. Kim et al. [21] estimated the extrinsics between two LiDARs by assuming three different poses and using a single reflective plane with a reflective strip. Zhang et al. [22] employed the invariant property of the sphere center viewpoint for calibrating multiple LiDARs with long baseline and large viewpoint difference. Although using auxiliary markers as targets enhances accuracy and reduces computational complexity, it incurs high time and energy costs. In contrast, methods based on natural environmental features [23] are relatively simpler and do not require additional effort to set up the scene. For example, three linearly independent planar features can be used to form a corner shape as the calibration target, and dual LIDARs can be calibrated through plane matching. Methods [24]–[27] employ road scene features to calibrate the extrinsics of multiple LiDARs. Nonetheless, the feature-based method mentioned above requires all LiDARs to have overlapping FoV, and it may degrade or fail in situations where the common FoV is small or non-existent. To address this issue, motion [28]–[31] is introduced into the system to artificially create FoV overlap and generate co-visible features. Map alignment is then employed to calibrate the LiDAR extrinsics. Nevertheless, this extrinsic calibration method is dependent on the accuracy of each LiDAR's odometry. When there are few constrained features in the environment, large odometry errors may occur, resulting in incorrect map alignment and low or failed calibration accuracy.

In summary, we contend that using feature-based methods can achieve higher accuracy, but calibrating LiDARs without overlapping FoV can pose challenges. Liu et al. [29], [30] employed solid-state LiDAR sensors characterized by small FoV and limited range. While motion is utilized to generate a historical common FoV, this approach is susceptible to cumulative motion errors and complicated by the calibration process. To overcome these issues, we propose incorporating the LiDAR with a large FoV as a bridge to generate an "overlapping FoV". In comparison to the approaches proposed in [28]–[31], our method avoids the issue of accumulated errors caused by motion drift. Furthermore, we propose the utilization of multi-scene joint calibration to address the issue of missing constraints. This approach allows for the inclusion of multiple scenes in the calibration process, which can compensate for the lack of constraints in individual scenes. By adopting this approach, the calibration process's accuracy and robustness can be improved.

The subsequent sections detail our work. Section III expounds on our methodology, encompassing LiDAR-Bridge and LiDAR-Align. In Section IV, we present concrete experiments to validate the accuracy of LiDAR-Link. Lastly, Section V offers our conclusion.

III. Methodology

A. Overview

The primary goal of LiDAR-Link is to calibrate the extrinsics among multiple solid-state LiDARs with limited or no overlapping FoV, as illustrated in the Fig. 1. LiDAR-Link consists of two modules: LiDAR-Bridge and LiDAR-Align.

LiDAR-Bridge employs a large, high-resolution LiDAR as a bridge LiDAR to establish an overlapping FoV among other smaller LiDARs. This method enables calibration without a common FoV to be transformed into calibration between LiDARs with an overlapping FoV, meeting the usage requirements of LiDAR-Align. As depicted in Fig. 1, the bridge LiDAR coordinate system is denoted by B, the master LiDAR coordinate system by M, and the subordinate LiDAR coordinate system by S. By aligning point clouds with an overlapping FoV, the coordinate system transformations $\mathbf{T}_{BM}$ between the bridge LiDAR and the master, as well as $\mathbf{T}_{BS}$ between the bridge LiDAR and the subordinate, can be determined. Subsequently, the extrinsics $\mathbf{T}_{MS}$ between the master and the subordinate can be indirectly calculated:

$$\mathbf{T}_{MS} = \mathbf{T}_{BM}^{-1} \cdot \mathbf{T}_{BS}. \tag{1}$$

It is crucial to mention that the Livox Mid-360 LiDAR used in this method is a non-repetitive scanning LiDAR. In a stationary state, the point cloud resolution continually improves with increasing scanning time. Conversely, traditional mechanical LiDARs maintain constant resolution in a stationary state, and those with 16 or 32 lines exhibit low vertical resolution, resulting in missed details and diminished calibration accuracy. Although 64 or 128 line mechanical

LiDARs can theoretically serve as bridge LiDARs, their cost may be prohibitive.

LiDAR-Align is an observability-aware LiDAR extrinsic calibration method that supports multi-scene joint computation with overlapping FoVs. The pipeline is depicted in Fig. 2:

1) Multiple LiDARs remain stationary and collect sufficient point clouds.
2) LiDAR-Align adaptively extracts planes in voxels, while probabilistically modeling the planes.
3) It constructs point-plane residuals using nearest neighbor matching and aligns the point cloud through IEKF.
4) Based on the covariance of IEKF, LiDAR-Align determines whether the currently aligned environmental point cloud has sufficient constraints. If not, it continues to collect additional environmental point clouds and repeats step 1) - 4).
5) Ultimately, it combines the collected point clouds from all scenes and aligns them using the Generalized-ICP in parallel spaces to obtain the extrinsics.

The remainder of this section focuses on LiDAR-Align.

B. Probabilistic Plane Map Building

1) Probabilistic Plane Modelling:

Initially, the uncertainty of the points $\mathbf{p}$ in the source LiDAR point cloud is modeled. As suggested by [32], the noise present in a point $\mathbf{p}_i$ includes ranging noise ($\delta_{d_i} \sim \mathcal{N}(0, \mathbf{C}_{d_i})$) in the range measurement $d_i \in \mathbb{R}$ and bearing direction noise ($\boldsymbol{\delta\omega}_i \sim \mathcal{N}(\mathbf{0}_{2\times1}, \mathbf{C}_{\boldsymbol{\omega}_i})$) on the tangent plane of $\boldsymbol{\omega}_i \in \mathbb{S}^2$. The noise $\boldsymbol{\delta}_{\mathbf{p}_i}$ and covariance $\mathbf{C}_{\mathbf{p}_i}$ related to point $\mathbf{p}_i$ can be formulated as

$$\begin{aligned} \boldsymbol{\delta}_{\mathbf{p}_i} &= \underbrace{\left[\boldsymbol{\omega}_i - d_i \left\lfloor \boldsymbol{\omega}_i \right\rfloor_\wedge \mathbf{N}(\boldsymbol{\omega}_i)\right]}_{\mathbf{A}_i} \begin{bmatrix} \delta_{d_i} \\ \boldsymbol{\delta}_{\boldsymbol{\omega}_i} \end{bmatrix} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\mathbf{p}_i}), \\ \mathbf{C}_{\mathbf{p}_i} &= \mathbf{A}_i \begin{bmatrix} \mathbf{C}_{d_i} & \mathbf{0}_{1\times2} \\ \mathbf{0}_{2\times1} & \mathbf{C}_{\boldsymbol{\omega}_i} \end{bmatrix} \mathbf{A}_i^T, \end{aligned} \tag{2}$$

where $\mathbf{N}(\boldsymbol{\omega}_i) = [\mathbf{N}_1\ \mathbf{N}_2] \in \mathbb{R}^{3\times2}$ represents an orthonormal basis of the tangent plane at ω_i, $\lfloor\rfloor_\wedge$ denotes the skew-symmetric matrix used for the cross product.

Then, a plane $(\mathbf{n}, \mathbf{q})$ is represented by its normal vector $\mathbf{n}$ and a point $\mathbf{q}$ on the plane, and it is fitted by N points $\mathbf{p}_i$ $(i = 1, ..., N)$ through the function $\mathbf{f}$:

$$[\mathbf{n}, \mathbf{q}]^T = \mathbf{f}(\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N). \tag{3}$$

Therefore, the ground-truth $(\mathbf{n}^{gt}, \mathbf{q}^{gt})$ can be formulated as

$$\begin{aligned} \left[\mathbf{n}^{gt}, \mathbf{q}^{gt}\right]^T &= \mathbf{f}(\mathbf{p}_1 + \boldsymbol{\delta}_{\mathbf{p}_1}, \mathbf{p}_2 + \boldsymbol{\delta}_{\mathbf{p}_1}, \ldots, \mathbf{p}_N + \boldsymbol{\delta}_{\mathbf{p}_N}) \\ &\approx [\mathbf{n}, \mathbf{q}]^T + \sum_{i=1}^{N} \frac{\partial \mathbf{f}}{\partial \boldsymbol{p}_i} \boldsymbol{\delta}_{\mathbf{p}_i}. \end{aligned} \tag{4}$$

As a result, the covariance matrix $\mathbf{C}_{plane}$ is:

$$\mathbf{C}_{plane} = \sum_{i=1}^{N} \frac{\partial \mathbf{f}}{\partial \mathbf{p}_i} \mathbf{C}_{\mathbf{p}_i} \frac{\partial \mathbf{f}}{\partial \mathbf{p}_i}^T \tag{5}$$

The specific derivation is omitted here due to space limitations, more details can be found in [32], [33].

2) Voxel Plane Map Building:

To construct a voxel plane map, we utilize an adaptive voxel map organized by a Hash table and an octree for each Hash entry. Initially, the global world frame is divided into voxels, each with a size corresponding to the map resolution. Subsequently, the accumulated LiDAR points are distributed among the voxels. Voxels that contain points are indexed in the Hash table. For each populated voxel, if all the points within it lie on a plane (indicated by the minimum eigenvalue of the point covariance matrix being below a specified threshold), the plane points are stored. The plane parameters $(\mathbf{n}, \mathbf{q})$ as in (3) and their uncertainty $\mathbf{C}_{plane}$ as in (5) are then calculated. In cases where the points in the voxel do not lie on a plane, the voxel is subdivided into eight octants. The process of plane checking and voxel subdivision is repeated in each octant until the maximum number of layers is reached. It's worth noting that the size of voxels varies, each voxel contains one plane feature obtained by fitting the raw LiDAR points it contains.

C. Alignment and Observability Analysis for Single-Scene Calibration

The extrinsics between two LiDARs are represented as

$$\begin{aligned} &\mathcal{M} = SO(3) \times \mathbb{R}^3, \dim(\mathcal{M}) = 6, \\ &\mathbf{T} \doteq \begin{bmatrix} \mathbf{R}^T & \mathbf{t}^T \end{bmatrix}^T \in \mathcal{M}. \end{aligned} \tag{6}$$

Here, we have $\mathbf{T} = \mathbf{T}_{SourceTarget}$, which denotes the transformation of the point cloud from the target LiDAR coordinate system to the source LiDAR coordinate system. To simplify notation, subscripts are omitted. Here, $\mathbf{R}$ denotes rotation and $\mathbf{t}$ stands for translation.

The alignment of the point cloud is achieved using IEKF, similar to the VoxelMap [32]. Below is a brief overview of the general principle, omitting the specific derivation of formulas. We first have an initial value $\widehat{\mathbf{T}}_n$ and its corresponding covariance $\widehat{\mathbf{P}}_n$, which can come from a CAD model or the previous estimation result, and the subscript n represents the number of iterations for single-scene calibration. Subsequently, nearest neighbor matching is employed to match the points from the target LiDAR with the planes in the Voxel Plane Map, leading to the construction of the following point-to-plane residuals:

$$\mathbf{p}_i = \mathbf{R}\mathbf{p}_i^{'} + \mathbf{t}, \tag{7}$$

$$d_i = \mathbf{n}_i^T (\mathbf{p}_i - \mathbf{q}_i), \tag{8}$$

where $\mathbf{p}_i^{'}$ represents a LiDAR point collected by the target LiDAR in the traget LiDAR coordinate system. According to (5), we can derive the uncertainty:

$$d_i = \left(\mathbf{n}_i^{gt} \operatorname{Exp}(\boldsymbol{\delta}_{\mathbf{n}_i})\right)^T \left[\left(\mathbf{p}_i^{gt} + \boldsymbol{\delta}_{\mathbf{p}_i}\right) - \mathbf{q}_i^{gt} - \boldsymbol{\delta}_{\mathbf{q}_i}\right], \tag{9}$$

$$d_i \sim \mathcal{N}(0, \mathbf{C}_{d_i}), \tag{10}$$

where $\operatorname{Exp}(\cdot) = \mathbf{I} + \frac{\cdot}{\|\cdot\|}\sin(\|\cdot\|) + \frac{\cdot^2}{\|\cdot\|^2}(1 - \cos(\|\cdot\|))$ is the exponential map, $\mathbf{C}_{d_i}$ is the covariance of d_i.

Hence, using the initial value $\widehat{\mathbf{T}}_n$ as a prior, $\widehat{\mathbf{P}}_n$ as a prior covariance, point-plane residuals d_i as observations and $\mathbf{C}_{d_i}$ as the observation covariance, IEKF is applied for iterative estimation of extrinsics to obtain the posterior extrinsics $\overline{\mathbf{T}}_n$,

and posterior covariance $\overline{\mathbf{P}}_n$, which are equivalent to the following maximum a posteriori (MAP) estimation:

$$\overline{\mathbf{T}}_n = \min_{\mathbf{T}_n} \left(\left\| \mathbf{T}_n \boxminus \widehat{\mathbf{T}}_n \right\|^2_{\widehat{\mathbf{P}}_n} + \sum_{i=1}^{\mathbf{I}} \|d_i\|^2_{\mathbf{C}_{d_i}} \right), \tag{11}$$

where $\boxminus$ is a generalized subtraction, $\mathbf{I}$ stands for the number of point-to-plane constrains, and the distance d_i is related to extrinsics $\mathbf{T}_n$ according to (7).

Observability analysis is conducted on the state covariance derived from state estimation. The study infers that a point-plane constraint can impose constraints on both rotational and translational degrees of freedom. The eigenvalues of the covariance matrix, $\overline{\mathbf{P}}_n$, are computed. Should the eigenvalues exceed the threshold vector $\epsilon_{\mathbf{P}}$ (determined experimentally), it is concluded that the constraints provided by the calibration scene are insufficient and additional constraints are required. In such a case, a new scene is selected, and point cloud data is gathered using the previous calibration results and covariance as initial values, instigating a new round of single-scene calibration. This process is repeated until the observability analysis indicates that the constraints are satisfied. Consequently, the magnitude of the covariance eigenvalues can serve as a quantitative indicator for evaluating calibration quality.

It is noteworthy that if certain degrees of freedom are underconstrained in one single-scene calibration and others are underconstrained in the subsequent scene, the information from these two scenes may complement each other. This complementarity provides complete and sufficient constraints for calibration. Thus, by employing the Kalman Filter to fuse the priori and covariance information from the previous iteration, it becomes possible to ascertain whether the constraints from all scenes are sufficient. Specifically, in the first single-scene calibration, sufficient translation constraints are discovered in the x and z directions, resulting in a larger covariance for the y-axis translation in the first calibration result. In the second calibration, sufficient translation constraints are found in the y and z directions. By using the first calibration result and covariance as the priori, it occurs that the covariance of the second calibration result is smaller than the threshold. This suggests that the constraints from both scenes can satisfy the requirements for extrinsic calibration.

D. Generalized-ICP for Multi-Scenes Calibration

As depicted in Fig. 2, upon collecting sufficient constraint point clouds, we conduct joint extrinsic estimation using Generalized-ICP [12] on all point clouds. Within a single scene, extrinsic estimation based on the Voxel Probabilistic Plane Map entails a plane fitting process, which might lead to some loss of detailed information and a reduction in accuracy. Moreover, in the iterative process of single-scene calibration, the calibration results of one scene are merged with those of another scene using the Kalman Filter, but without direct merging at the point-plane residual level. To address this concern, we leverage the raw point cloud and carry out extrinsic estimation using Generalized-ICP (point-to-plane) to align multiple target point clouds with multiple source point clouds. This can also be encapsulated as maximum a posteriori (MAP) estimation for optimal estimation of extrinsics $\mathbf{T}^*$:

$$\mathbf{T}^* = \min_{\mathbf{T}} \left(\sum_{j=1}^{\mathbf{J}} \sum_{i=1}^{\mathbf{I}} \|d_{ji}\|^2_{\mathbf{C}} \right), \tag{12}$$

where d_{ji} means the i_{th} point-to-plane residual in the j_{th} scene, which is also related to extrinsic $\mathbf{T}$. $\mathbf{C}$ is the corresponding covariance. Our LiDAR-Align method is summarized in Algorithm 1.

Algorithm 1 LiDAR-Align

Input: Prior extrinsics $\mathbf{T}_0$ and prior covariance $\mathbf{P}_0$.

$\widehat{\mathbf{T}}_{n=0} = \mathbf{T}_0$, $\widehat{\mathbf{P}}_{n=0} = \mathbf{P}_0$, $n = -1$;
repeat
  $n = n + 1$;
  Point cloud accumulation for stationary LiDARs;
  Constructing a Voxel Probabilistic Plane Map using point clouds from source LiDAR;
  $m = -1$;
  Start IEKF:
  $\widehat{\mathbf{T}}_n^{m=0} = \widehat{\mathbf{T}}_n$;
  repeat
    $m = m + 1$;
    Match the point cloud of the target LiDAR with the planes in the Voxel Probabilistic Plane Map;
    Compute residual and Jocobin $\mathbf{H}$;
    Compute the extrinsics update $\widehat{\mathbf{T}}_n^{m+1}$ and the Kalman gain $\mathbf{K}$;
  until $\left\| \widehat{\mathbf{T}}_n^{m+1} \boxminus \widehat{\mathbf{T}}_n^{m} \right\| < \epsilon_{\mathbf{T}}$;
  $\overline{\mathbf{T}}_n = \widehat{\mathbf{T}}_n^{m+1}$; $\overline{\mathbf{P}}_n = (\mathbf{I} - \mathbf{K}\mathbf{H})\widehat{\mathbf{P}}_n$;
  End IEKF;
  $\widehat{\mathbf{T}}_{n+1} = \overline{\mathbf{T}}_n$, $\widehat{\mathbf{P}}_{n+1} = \overline{\mathbf{P}}_n$;
until $\overline{\mathbf{P}}_n.eigenValue() < \epsilon_{\mathbf{P}}$;
Utilize multi-scene point clouds to perform Generalized-ICP in parallel spaces.

Output: Optimal extrinsics $\mathbf{T}^*$.

Remark 1: Multi-scene calibration does not entail amalgamating the point clouds in the same space for Generalized-ICP, as this approach would induce matching errors. Instead, it is executed in "parallel spaces" where the nearest neighbor matching is performed independently. Subsequently, point-plane residuals are constructed, finally, all the constraints from each parallel space are consolidated to estimate the extrinsics.

Remark 2: Our method of iterating through single-scene calibrations before executing multi-scene calibration boasts two main advantages over directly applying Generalized-ICP. Firstly, by cycling through multiple single-scene calibrations, we are able to achieve superior initial values that facilitate faster processing compared to using Generalized-ICP directly. Secondly, this strategy allows for a precise and quantitative evaluation of the constraints' adequacy, eliminating the necessity for manual judgement. Hence, it is well-suited for industrial automation calibration.

Remark 3: LiDAR-Align is capable of calibrating a diverse range of LiDAR types that have overlapping FoV. Such types

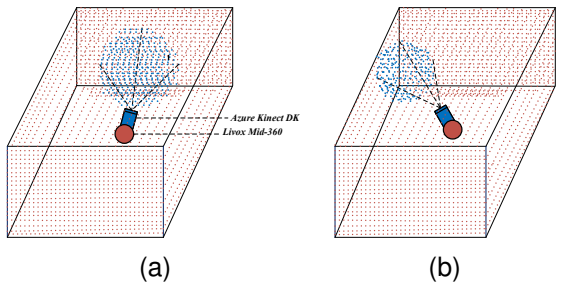

Fig. 3. In the room, the Mid 360 LiDAR and one of the Azure Kinect DK LiDAR devices collected point clouds. The red points correspond to the point cloud generated by Mid 360, while the blue points represent the point cloud from the Azure Kinect DK. (a) The alignment of the point clouds from both LiDARs lacks effective constraints in the left and right direction. (b) The constraints for aligning the two sets of point clouds are well-established.

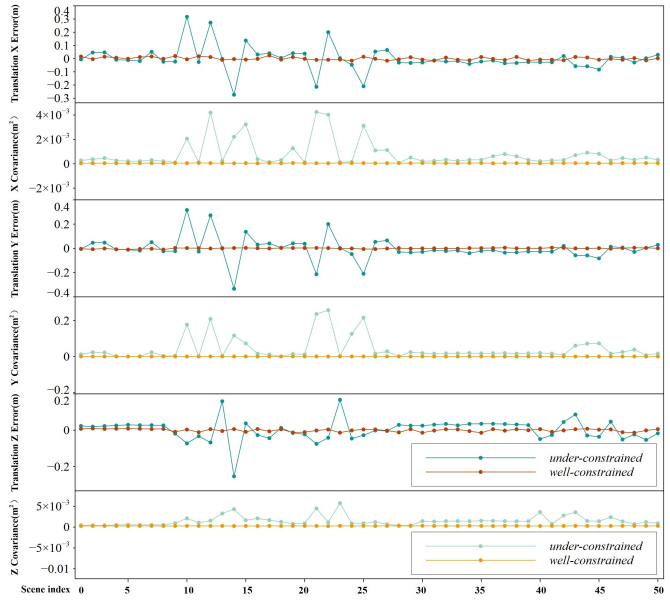

Fig. 4. The point clouds obtained from 100 different scenes are calibrated individually using LiDAR-Align for single-scene calibration, including 50 under-constrained scenes and 50 well-constrained scenes. The calibration errors of the three translational degrees of freedom (magnitude of numerical fluctuations) are found to be positively correlated with the size of the covariance. For the sake of visual convenience, we have applied a centering transformation to the calibration results. Due to space constraints, the three degrees of freedom of rotation are not shown, but the general trend is the same as translation.

include, but are not limited to, solid-state LiDAR, mechanical LiDAR, and Time-of-Flight (ToF) depth cameras.

IV. Experiments

A. Experiment Setup

For the experiment, we calibrate a handheld device consisting of a Livox Mid-360 and three Azure Kinect DK LiDARs, as illustrated in Fig. 1. The computations are executed on an NUC. The Livox Mid-360 serves as a bridge, offering a horizontal FoV of 360 degrees and a vertical FoV of 59 degrees. The Azure Kinect DK features both a horizontal and vertical FoV of 120 degrees. The three Azure Kinect DK LiDARs demonstrate minimal FoV overlap, whereas each smaller LiDAR overlaps substantially with the bridge LiDAR. We execute two series of experiments to evaluate the precision of LiDAR-Align and LiDAR-Bridge respectively.

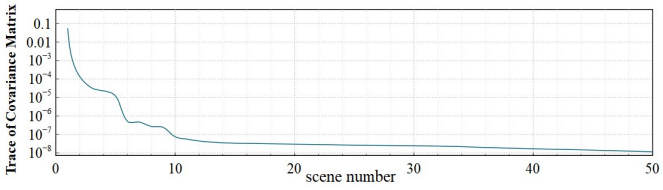

Fig. 5. As the number of calibration scenes increases, the trace of the translation covariance matrix in the LiDAR-Align calibration results gradually decreases and converges.

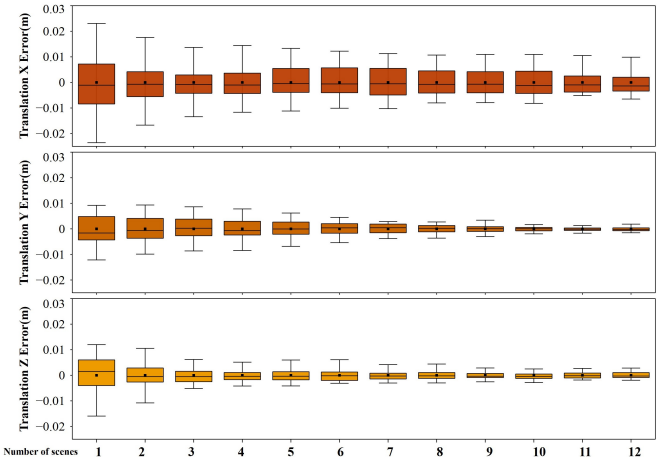

Fig. 6. 50 sets of repeated experiments are conducted using different numbers of scene point clouds with LiDAR-Align. The x-axis represents the number of scenes involved in the calibration and the y-axis represents the mean and standard deviation of the results in the three translational DoF.The results demonstrate a gradual reduction in the consistency error of the calibration results as the number of scenes increases. For the sake of visual convenience, we have applied a centering transformation to the calibration results.

B. Observability and Accuracy Evaluation (LiDAR-Align)

In this section, we collected 100 sets of point clouds from various scenes, 50 sets of which had experienced under-constrained alignment during calibration as shown in Fig. 3 (a), others were well-constrained in Fig. 3 (b). Fig. 4 displays the relationship between the index of the point cloud pairs collected in different scenarios and the estimated values of the three translational degrees of freedom (DoF) of extrinsics and their corresponding covariances. It can be observed that the size of the covariance (indicating the adequacy of constraints) is directly correlated with the fluctuation of the extrinsics. Generally, when the covariance is large, the error in the extrinsics tends to be high as well. Therefore, the covariance can serve as a criterion during calibration to evaluate if the current point cloud pair satisfies adequate constraints.

The LiDAR-Align in Algorithm 1 was executed (the convergence condition at line 17 was not being used), and a total of 50 experiments were conducted by gradually increasing the number of collected scenes from 1 to 50. In Fig. 5, the x-axis represents the number of scenes involved in the calibration, while the y-axis represents the trace of the covariance matrix corresponding to the translation results. It can be observed that as the number of scenes involved in the calibration increases, the covariance gradually decreases and eventually converges. This is consistent with the principle that the more observations we have, the higher the certainty of the state will be.

We randomly selected n sets of data from these 50 groups

TABLE I
ACCURACY COMPARISON (LIDAR-ALIGN)

Method	Constrain		Rotation(deg)			Translation(m)		
			roll	yaw	pitch	x	y	z
ICP	well	mean	96.226	-0.042	90.079	-0.003	-0.182	-0.080
		std	0.093	0.026	0.138	0.010	0.005	0.006
	unwell	mean	97.408	-0.016	90.113	-0.009	0.033	-0.064
		std	2.114	7.430	0.351	0.103	0.227	0.036
GICP	well	mean	96.225	-0.042	90.081	-0.003	-0.182	-0.079
		std	0.094	0.025	0.138	0.010	0.005	0.006
	unwell	mean	97.253	-0.032	90.131	0.001	0.028	-0.066
		std	1.948	7.414	0.333	0.098	0.231	0.0337
CROON	well	mean	96.226	-0.042	90.086	-0.004	-0.182	-0.081
		std	0.093	0.026	0.140	0.011	0.005	0.006
	unwell	mean	96.691	0.114	90.266	-0.003	-0.165	-0.078
		std	1.923	6.954	0.383	0.088	0.202	0.029
Ours	well	mean	96.178	0.020	90.225	-0.003	-0.194	-0.089
		std	**0.050**	**0.024**	**0.050**	**0.006**	**0.002**	**0.002**
	unwell	mean	96.183	0.026	90.325	-0.003	-0.192	-0.087
		std	0.110	0.048	0.075	0.009	0.005	0.004

to apply the LiDAR-Align algorithm, where n ranges from 1 to 12, symbolizing the number of paired point clouds utilized from distinct scenes. For each n, we conducted 50 experiments using different pairs of point clouds collected from various scenes. The experimental results are presented in the box plot shown in Fig. 6. It can be observed that as the number of scenes involved in the calibration increases, the overall trend of the error in repeatability decreases, demonstrating the robustness and stability of our method. Furthermore, we found that the error reached a relatively small value when using 4 scenes. Therefore, we consider that in our calibration environment, it is necessary to have at least 4 or more different pairs of point clouds. It is important to note that the specific number of scenes selected depends on the calibration environment, specifically whether there are enough high-quality planar features. The determination of the number of different point clouds to be used should follow the convergence criteria employed in LiDAR-Align algorithm.

Simultaneously, we performed calibration using ICP [9], Generalized-ICP [12], CROON [24], and ours (with 6 scenes) in both well-constrained and under-constrained scenes. The results are presented in Table I. The bold numbers represent the best calibration results in well-constrained scenes, while the underscored numbers indicate the best results in under-constrained scenes. We performed 50 repetitions, obtaining the mean and standard deviation (std) for the six DoF. In this experiment, we employ the std as a metric to evaluate the calibration results, which can express the consistency error. It is evident that the calibration outcomes are generally better in well-constrained scenarios compared to under-constrained ones. Our LiDAR-Align algorithm, leveraging the fusion of data from multiple scenes, demonstrates good accuracy even in under-constrained calibration scenarios, as constraints complement each other.

In summary, this demonstrates that the proposed method can identify cases where ICP constraints are insufficient and effectively address the issue of reduced calibration accuracy resulting from ICP degradation, as discussed in Section I-A(b).

TABLE II
ACCURACY COMPARISON OF OURS WITH MLCC (LIDAR-BRIDGE)

Method	extrinsic		Rotation(deg)			Translation(m)		
			roll	yaw	pitch	x	y	z
MLCC	F-L	mean	84.329	-95.821	77.868	0.215	0.058	-0.169
		std	0.411	0.119	0.433	0.094	0.079	0.027
	F-R	mean	112.165	96.524	106.145	0.199	0.298	-0.207
		std	0.276	0.577	0.274	0.072	0.073	0.012
Ours	F-L	mean	86.933	-96.447	80.426	-0.189	0.0198	0.155
		std	**0.204**	**0.026**	**0.213**	**0.004**	**0.002**	**0.002**
	F-R	mean	111.921	96.326	-106.393	0.186	0.018	-0.164
		std	**0.186**	**0.029**	**0.189**	**0.004**	**0.001**	**0.002**

C. Non-Overlapping Calibration (LiDAR-Bridge)

Finally, we conducted calibration for configure in Fig. 1 and compared the results of our LiDAR-Bridge method with an open-source algorithm, MLCC [30], as it was the sole open-source project supporting non-overlapping calibration. In Table II, "F-L" and "F-R" represent the extrinsics for the front-left and front-right respectively. We used the initial rough extrinsics (derived from a CAD model) for both methods. For our approach, we used LiDAR-Align to calibrate the extrinsics using point clouds from ten scenes (captured using a Livox Mid-360 and an Azure Kinect DK), following the convergence criteria of LiDAR-Align. After obtaining the extrinsics, we further utilized LiDAR-Bridge to transform them into the extrinsics between the two Azure Kinect DK LiDARs. We performed 50 repetitions of this process. On the other hand, MLCC collected the required calibration data by rotating a full circle, while we conducted 50 repetitions of the consistency experiment using 50 different sets of data to get the results. By quantitatively comparing the results in Table II, our method achieved better accuracy in roll, yaw, pitch, x, y, and z. The alignment of point clouds from four LiDARs using LiDAR-Bridge's calibration results is shown in the Fig. 7. We attribute our higher accuracy to two main factors:

Firstly, LiDAR-Link employed a greater number of scenes to ensure sufficient constraints based on observability. In contrast, MLCC encountered degeneration during some calibration processes, leading to significant consistency errors.

Secondly, our indirect calibration process only involved a single intermediary, the bridge LiDAR, whereas MLCC's calibration relies on the fusion of multiple point clouds, resulting in larger cumulative errors.

As a result, it can be observed that LiDAR-Bridge can address the non-overlapping calibration issue described in Sec-

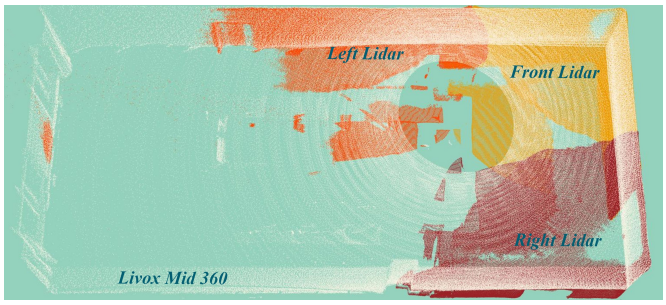

Fig. 7. The point clouds from three small FoV LiDARs are aligned to Livox Mid-360's coordinate system using the calibration result (bird's-eye view).

tion I-A(a). When combined with the Lidar-Align approach, it enables high-precision calibration results.

V. Conclusion

In this letter, we propose the calibration methods of LiDAR-Bridge and LiDAR-Align. LiDAR-Bridge enables the transformation of non-overlapping calibrations into overlapping calibrations through the utilization of a wide-angle and high-resolution LiDAR as a bridge. Meanwhile, LiDAR-Align is designed for LiDARs with overlapping FOVs, allowing the use of calibrated covariances to assess the adequacy of constraints and the reliability of calibration results. Moreover, it enables precise parallel spacetime calibration using collected point clouds from multiple scenes.

Extensive experiments have demonstrated the effectiveness and high accuracy of the proposed calibration methods. Our open-source code will contribute to the automation of LiDAR calibration in the fields of autonomous driving and robotics, catering to both overlapping and non-overlapping scenarios. In our future work, we aim to mount the Bridge LiDAR in the calibration room instead of on the robot or carrier.

References

[1] H. Kim, H. Kim, S. Lee, and H. Lee, "Autonomous exploration in a cluttered environment for a mobile robot with 2d-map segmentation and object detection," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6343–6350, 2022.

[2] J. Ji, T. Yang, C. Xu, and F. Gao, "Real-time trajectory planning for aerial perching," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 10 516–10 522.

[3] J. Chen, H. Wang, M. Hu, and P. N. Suganthan, "Versatile lidar-inertial odometry with se (2) constraints for ground vehicles," *IEEE Robotics and Automation Letters*, 2023.

[4] R. Horaud and F. Dornaika, "Hand-eye calibration," *The international journal of robotics research*, vol. 14, no. 3, pp. 195–210, 1995.

[5] L. Zhou, Z. Li, and M. Kaess, "Automatic extrinsic calibration of a camera and a 3d lidar using line and plane correspondences," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 5562–5569.

[6] Y. Xie, R. Shao, P. Guli, B. Li, and L. Wang, "Infrastructure based calibration of a multi-camera and multi-lidar system using apriltags," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 605–610.

[7] N. Heide, T. Emter, and J. Petereit, "Calibration of multiple 3d lidar sensors to a common vehicle frame," in *Isr 2018; 50th international symposium on robotics*. VDE, 2018, pp. 1–8.

[8] D.-H. Kim and G.-W. Kim, "Efficient calibration method of multiple lidars on autonomous vehicle platform," in *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*. IEEE, 2020, pp. 446–447.

[9] P. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[10] H. Kim and A. Hilton, "Influence of colour and feature geometry on multi-modal 3d point clouds data registration," in *2014 2nd international conference on 3D vision*, vol. 1. IEEE, 2014, pp. 202–209.

[11] L. Heng, "Automatic targetless extrinsic calibration of multiple 3d lidars and radars," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 669–10 675.

[12] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp." in *Robotics: science and systems*, vol. 2, no. 4. Seattle, WA, 2009, p. 435.

[13] J. Lin, X. Liu, and F. Zhang, "A decentralized framework for simultaneous calibration, localization and mapping with multiple lidars," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4870–4877.

[14] C. Chen, G. Xiong, Z. Zhang, J. Gong, J. Qi, and C. Wang, "3d lidar-gps/imu calibration based on hand-eye calibration model for unmanned vehicle," in *2020 3rd International Conference on Unmanned Systems (ICUS)*. IEEE, 2020, pp. 337–341.

[15] S. Das, N. Mahabadi, A. Djikic, C. Nassir, S. Chatterjee, and M. Fallon, "Extrinsic calibration and verification of multiple non-overlapping field of view lidar sensors," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 919–925.

[16] J. Jiao, H. Ye, Y. Zhu, and M. Liu, "Robust odometry and mapping for multi-lidar systems with online extrinsic calibration," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 351–371, 2021.

[17] Z. Taylor and J. Nieto, "Motion-based calibration of multimodal sensor extrinsics and timing offset estimation," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1215–1229, 2016.

[18] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[19] S. Das, L. af Klinteberg, M. Fallon, and S. Chatterjee, "Observability-aware online multi-lidar extrinsic calibration," *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2860–2867, 2023.

[20] B. Xue, J. Jiao, Y. Zhu, L. Zhen, D. Han, M. Liu, and R. Fan, "Automatic calibration of dual-lidars using two poles stickered with retro-reflective tape," in *2019 IEEE International Conference on Imaging Systems and Techniques (IST)*. IEEE, 2019, pp. 1–6.

[21] J. Kim, C. Kim, and H. J. Kim, "Robust extrinsic calibration for arbitrarily configured dual 3d lidars using a single planar board," in *2020 20th International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 2020, pp. 576–580.

[22] J. Zhang, Q. Lyu, G. Peng, Z. Wu, Q. Yan, and D. Wang, "Lb-l2l-calib: Accurate and robust extrinsic calibration for multiple 3d lidars with long baseline and large viewpoint difference," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 926–932.

[23] J. Jiao, Q. Liao, Y. Zhu, T. Liu, Y. Yu, R. Fan, L. Wang, and M. Liu, "A novel dual-lidar calibration algorithm using planar surfaces," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 1499–1504.

[24] P. Wei, G. Yan, Y. Li, K. Fang, X. Cai, J. Yang, and W. Liu, "Croon: Automatic multi-lidar calibration and refinement method in road scene," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 12 857–12 863.

[25] Z. Zhang, C. Fu, C. Dong, C. Mertz, and J. M. Dolan, "Self-calibration of multiple lidars for autonomous vehicles," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 2897–2902.

[26] J. Jiao, Y. Yu, Q. Liao, H. Ye, R. Fan, and M. Liu, "Automatic calibration of multiple 3d lidars in urban environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 15–20.

[27] F. Ma, S. Wang, and M. Liu, "An automatic multi-lidar extrinsic calibration algorithm using corner planes," in *2022 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2022, pp. 235–240.

[28] J. Xu, R. Li, L. Zhao, W. Yu, Z. Liu, B. Zhang, and Y. Li, "Cammap: Extrinsic calibration of non-overlapping cameras based on slam map alignment," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 879–11 885, 2022.

[29] X. Liu and F. Zhang, "Extrinsic calibration of multiple lidars of small fov in targetless environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2036–2043, 2021.

[30] X. Liu, C. Yuan, and F. Zhang, "Targetless extrinsic calibration of multiple small fov lidars and cameras using adaptive voxelization," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.

[31] M. Nie, W. Shi, W. Fan, and H. Xiang, "Automatic extrinsic calibration of dual lidars with adaptive surface normal estimation," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–11, 2022.

[32] C. Yuan, W. Xu, X. Hong, and F. Zhang, "Efficient and probabilistic adaptive voxel mapping for accurate online lidar odometry," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8518–8525, 2022.

[33] C. Yuan, X. Liu, X. Hong, and F. Zhang, "Pixel-level extrinsic self calibration of high resolution lidar and camera in targetless environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7517–7524, 2021.