2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Unified Calibration for Multi-camera Multi-LiDAR Systems using a Single Checkerboard

Wonmyung Lee*[1], Changhee Won*[1,2], and Jongwoo Lim†[1,2]

***Abstract*— In this paper, we propose a unified calibration method for multi-camera multi-LiDAR systems. Only using a single planar checkerboard, the captured checkerboard frames by each sensor are classified as either global frames if they are observed by at least two sensors, or a local frame if observed by a single camera. Both global and local frames of each camera are used to estimate its intrinsic parameters, whereas the global frames between sensors are for computing their relative poses. In contrast to the previous methods that simply combine the pairwise poses (e.g., camera-to-camera or camera-to-LiDAR) that are separately estimated, we further optimize the sensor poses in the system globally using all observations as the constraints in the optimization problem. We find that the point-to-plane distances are effective as camera-to-LiDAR constraints where the points are 3D positions of the checkerboard corners and the planes are estimated from the LiDAR point-cloud. Also, abundant corner observations in the local frames enable the joint optimization of intrinsic and extrinsic parameters in a unified framework. The proposed calibration method utilizes entire observations in a unified global optimization framework, and it significantly reduces the error caused by a simple composition of the relative sensor poses. We extensively evaluate the proposed algorithm qualitatively and quantitatively using real and synthetic datasets. We plan to make the implementation open to the public with the paper publication.**

I. INTRODUCTION

In many autonomous navigation systems, cameras and range sensors (e.g., LiDARs or radars) are commonly used to perceive surrounding environments and avoid obstacles. For instance, accurate 3D points from a LiDAR and corresponding RGB images from cameras are jointly utilized to detect objects' 3D bounding boxes [1]–[3], estimate vehicle's ego-motion [4], or reconstruct surrounding environments [5]. Meanwhile, to integrate individually observed data from heterogeneous sensors, calibration of the intrinsic and extrinsic parameters of the sensors must be preceded.

Calibration of heterogeneous sensors has been studied for decades. Various approaches [6]–[8] have been proposed for extrinsic calibration of a camera and a range sensor. Recently as more and more sensors are used in cars and robots, multi-sensor extrinsic calibration methods have also been proposed [9], [10]. The above multi-sensor calibration methods first estimate the relative poses of the pairs of sensors, e.g., camera-to-camera, LiDAR-to-LiDAR, or camera-to-LiDAR, and then find the sensor extrinsic by merging the estimated relative poses. However, when many sensors are involved, the errors in estimated relative poses propagate in pose composition and the resultant global poses may be inaccurate.

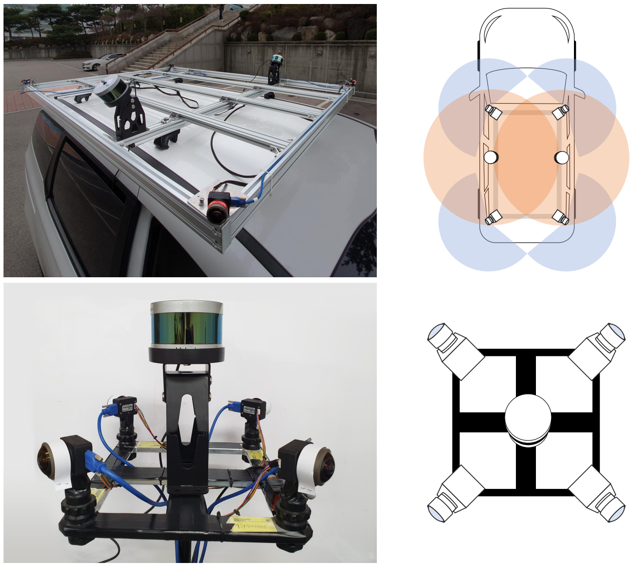


Fig. 1: Examples of our sensor configurations for omni-directional sensing, which consists of 4 cameras and 1 or 2 LiDARs. The proposed algorithm is capable of handling general sensor setups.

Furthermore, recent self-driving vehicles [11], [12] have multiple cameras and range sensors mounted with wide-baseline to observe surrounding objects in all directions. Wide field-of-view (FOV) lenses, e.g., fisheye or catadioptric lenses, are also commonly used to enlarge the sensing areas and minimize blind spots. However in such configuration with different FOVs and baselines, the overlapping region between the sensors may not be large enough for accurate relative pose estimation.

In this paper, we propose a unified framework for calibrating intrinsic and extrinsic parameters of multiple cameras and multiple LiDARs using only a single checkerboard. Each scene with a different checkerboard pose is captured by either one or multiple sensors, generating local frames or global frames, respectively. Similar to the existing methods, the initial sensor parameters are set using the pairwise poses between sensors from global frames, and the intrinsic parameters from all frames.

The main difference of our framework is the global unified optimization of all sensor parameters using the observations from the global and local frames. The corners in the local frames constrain the camera intrinsic parameters, whereas

*Authors contributed equally.

†Corresponding author.

[1]Department of Computer Sicence, Hanyang University, Seoul, Korea. {wonmyung102, chwon, jlim}@hanyang.ac.kr

[2]MultiplEYE Co., Ltd., Seoul, Korea.

978-1-7281-6211-9/20/$31.00 ©2020 IEEE

those in the global frames act on the camera intrinsic and extrinsic as well as the LiDAR pose parameters. We also propose to use the point-to-plane distance in camera-to-LiDAR calibration for easy and reliable plane estimation and accurate distance computation. As LiDARs provide a sufficient number of accurate 3D points of the checkerboard, they can be easily segmented and the 3D plane equation can be computed robustly. The relative poses between cameras and LiDARs are optimized to minimize the point-to-plane distance where the 3D points are built by triangulation of the checkerboard corners in multiple frames. All camera and LiDAR observations are modeled into the cost functions in a unified non-linear optimization problem, and all calibration parameters of the cameras and LiDARs are jointly updated to minimize the cost.

While our algorithms can handle various general sensor layouts, we choose to use the rigs in Fig. 1 for omnidirectional sensing by using only four ultra-wide FOV fisheye cameras and one or two LiDARs with only 16 vertical channels. On top of qualitative evaluation using real datasets, we synthetically render the images and depth readings with ground-truth sensor poses for quantitative evaluation. The extensive experiments show that the proposed algorithm performs well on such extremely distorted fisheye images and low-resolution LiDAR readings.

Our main contributions are as follows:

(i) We propose a unified calibration method for multi-camera multi-LiDAR systems. Using a single checkerboard, the observed global and local frames are all used in initialization as well as global optimization. All observations are formulated into the cost functions in a unified non-linear optimization to determine the camera intrinsic parameters and the 6-DOF sensor poses.

(ii) We find that the point-to-plane distance is effective in camera-to-LiDAR calibration. The checkerboard plane is estimated by the LiDAR point-could, and the distance of the 3D corner points from the images to the plane is optimized.

(iii) We verify our algorithm through extensive experiments on real and synthetic datasets in both qualitative and quantitative ways. All datasets and the code will be made public when the paper is published.

II. RELATED WORK

Mainly there have been two approaches for extrinsic calibration: with or without calibration objects.

Methods Using Calibration Objects Most calibration algorithms use known calibration objects such as checkerboards, AprilTags, or textured images, for accurate and robust calibration of intrinsic or extrinsic parameters. Among those reference objects, the checkerboard is one of the most commonly used objects since it is easy to make and find, and it can be also used to calibrate the camera intrinsic parameters. Zhang *et al.* [13], and Herrera *et al.* [14] calibrate extrinsic between a camera and a range sensor using a checkerboard. Zhuo *et al.* [15] also propose a calibration method for a perspective camera and a LiDAR using line features and plane correspondences of a checkerboard.

AprilTags are also widely used for extrinsic calibration of multiple sensors. Tang *et al.* [9] propose an algorithm for calibrating extrinsic of multi-sensor systems which equip multiple cameras and multiple 2D range sensors. Further, Xie *et al.* [10] design an infrastructure-based extrinsic calibration system for multi-sensor setups which aims a mass production scenario. Meanwhile, textured patterns [16], [17] or polygons [8] are also used for optimizing the relative pose between a camera and a LiDAR.

Methods Not Using Calibration Objects Contrary to the above methods, there have been several methods which do not need a target object, mainly for on-line calibration. For autonomous vehicle systems, specific information from surrounding environments, e.g., planar road or surfaces, can be utilized for targetless extrinsic calibration. Jeong *et al.* [18] calibrate extrinsic between a stereo camera and a LiDAR by estimating the road from the 3D point-cloud by the stereo camera and aligning it with the 3D points from the LiDAR. Similar to [10], Jiao *et al.* [19] use a planar infrastructure without any marker to calibrate the extrinsic parameters of multiple LiDARs. For general environments, image features from cameras [7], [20] and estimated motion from a video [21] can be used to align cameras and range sensors. Bileschi [22] also proposes a fully automatic calibration method by applying a structure-from-motion technique on a video stream. In addition, online extrinsic calibration methods for unstable sensor systems have been proposed, e.g., [23].

In this paper, we choose to use a moving checkerboard as the calibration object, for an effective, accurate, and unified multi-sensor calibration framework. Instead of using edges or boundaries of the checkerboard which requires a large overlap between sensors and high-resolution data of range sensors, we compute the 3D plane parameters from LiDARs' point-cloud and the 3D corner points from cameras are matched to the plane. In principle, other planar markers with matchable corner points (e.g., AprilTags) can be used instead of the checkerboard.

III. METHOD

In this section, we introduce our unified calibration method for multi-camera multi-LiDAR systems. Our algorithm has the following three steps: intrinsic calibration, global pose initialization, and global optimization. In the following subsections, the notations used in this section and individual steps are described in detail.

A. Notation

$\boldsymbol{\kappa}$ denotes the intrinsic parameters of a camera including lens distortion parameters, and the projection function Π maps a 3D point $\mathbf{P}$ to a 2D image coordinate $\mathbf{p}$ as $\mathbf{p} = \Pi(\mathbf{P}; \boldsymbol{\kappa})$. The pose of each sensor is represented as a 6-DOF extrinsic parameter $(\mathbf{r}^\top, \mathbf{t}^\top)^\top$, where $\mathbf{r}$ is an axis-angle rotation vector, $\mathbf{t}$ is a translation vector ($\mathbf{r}, \mathbf{t} \in \mathbb{R}^3$). The 3×4 rigid transformation matrix for the extrinsic parameter is given as $\mathbf{T} = \begin{bmatrix} R(\mathbf{r}) & \mathbf{t} \end{bmatrix}$, where $R(\mathbf{r})$ is the 3×3 rotation matrix corresponding to $\mathbf{r}$. We use the vector and matrix notations interchangeably in the text. The extrinsic matrix transforms a 3D point $\mathbf{P}$ to $\mathbf{T} \star \mathbf{P} = \mathbf{T} \begin{bmatrix} \mathbf{P}^\top & 1 \end{bmatrix}^\top$, and $\star$ and ${}^{-1}$ denote the composition of transformations and the inverse transformation respectively.

We use three coordinate systems, world (w), sensor (s), and checkerboard (b). When needed, the coordinate system is marked as left super- and sub-scripts, like ${}^{\mathrm{s}}_{\mathrm{w}}\mathbf{T}$ meaning the transformation from the world to the sensor coordinate system (so that ${}^{\mathrm{w}}_{\mathrm{s}}\mathbf{T}$=${}^{\mathrm{s}}_{\mathrm{w}}\mathbf{T}^{-1}$), or ${}^{\mathrm{b}}\mathbf{P}$ a point in the checkerboard coordinate system. For example, a 2D point $\mathbf{p}$ on the checkerboard image of k-th frame by i-th camera is written as

$${}^{\mathrm{s}_i}\mathbf{p}_k = \Pi_i({}^{\mathrm{s}_i}_{\mathrm{w}}\mathbf{T} \star {}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k \star {}^{\mathrm{b}}\mathbf{P}),$$

where $\Pi_i(\cdot) = \Pi(\cdot; \boldsymbol{\kappa}_i)$.

B. Intrinsic Calibration

Using a checkerboard, we first capture checkerboard frames from each sensor simultaneously, with sufficient variations of the checkerboard's poses [24] and overlaps among sensors. Our calibration method only uses the images by the cameras and the point-clouds by the LiDARs without any additional efforts. All camera images are used in the intrinsic calibration step. We follow the conventional intrinsic calibration methods, for perspective cameras [25] and for omnidirectional cameras [26], [27] which both use checkerboard images. Through intrinsic calibration, we acquire the initial estimate of the intrinsic parameters $\{\boldsymbol{\kappa}_{\mathrm{c}}\}$ of c-th camera and the transformation from the checkerboard to the camera $\{{}^{\mathrm{s}_{\mathrm{c}}}_{\mathrm{b}}\mathbf{T}_k\}$ of k-th frame.

C. Global Pose Initialization

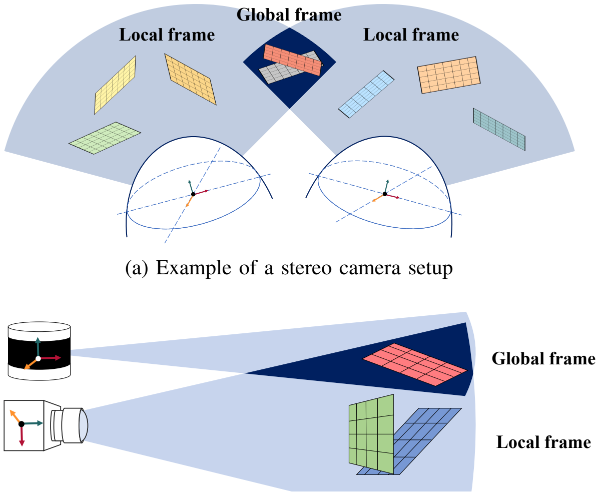
As in the conventional calibration methods for stereo camera or camera-to-LiDAR calibration, extrinsics between i-th and j-th sensors can be computed from the two relative poses, board to i-th camera and board to j-th camera, that are simultaneously taken at k-th frame:

$${}^{\mathrm{s}_j}_{\mathrm{s}_i}\mathbf{T}_k = {}^{\mathrm{s}_j}_{\mathrm{b}}\mathbf{T}_k \star {}^{\mathrm{s}_i}_{\mathrm{b}}\mathbf{T}_k^{-1},$$

where ${}^{\mathrm{s}_*}_{\mathrm{b}}\mathbf{T}_k$'s are acquired in the intrinsic calibration stage (Sec. III-B). To avoid choosing a bad relative pose, we use the relative pose ${}^{\mathrm{s}_j}_{\mathrm{s}_i}\mathbf{T}$ computed by taking element-wise median or average of the 6-DOF pose vectors $\{{}^{\mathrm{s}_j}_{\mathrm{s}_i}\mathbf{T}_k\}$. There might be no common frames between two sensors when there exist many sensors spread out in space. The extrinsic parameters between l-th and n-th cameras with no overlap are computed by composing the relative poses of an intermediate m-th camera as ${}^{\mathrm{s}_n}_{\mathrm{s}_l}\mathbf{T} = {}^{\mathrm{s}_n}_{\mathrm{s}_m}\mathbf{T} \star {}^{\mathrm{s}_m}_{\mathrm{s}_l}\mathbf{T}$.

By integrating all available relative poses, the poses of all sensors as well as the checkerboards in the global coordinate system can be initialized: the poses of c-th sensor and k-th checkerboard are given as

$$\begin{aligned} {}^{\mathrm{s}_c}_{\mathrm{w}}\mathbf{T} &= {}^{\mathrm{s}_c}_{\mathrm{s}_1}\mathbf{T} \star {}^{\mathrm{s}_1}_{\mathrm{w}}\mathbf{T}, \quad \text{and} \\ {}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k &= {}^{\mathrm{s}_c}_{\mathrm{w}}\mathbf{T}^{-1} \star {}^{\mathrm{s}_c}_{\mathrm{b}}\mathbf{T}_k, \end{aligned} \tag{1}$$



(a) Example of a stereo camera setup

(b) Example of a camera-LiDAR setup

Fig. 2: **Illustration of the global and local frames.** We classify the observed frames according to co-visibility of the sensors. Global frames are those observed by multiple sensors.

where ${}^{\mathrm{s}_1}_{\mathrm{w}}\mathbf{T}$ is the transformation from a reference sensor (1-st sensor) to the world coordinate system. However, the errors of each relative pose can be accumulated in the pose composition process, and choosing the right order of compositions can be tricky in complicated layouts. Therefore, we use the global poses by Eq. 1 only as the initialization, and we further optimize all calibration parameters, both intrinsics and extrinsics, in a unified non-linear optimization framework. The detailed formulation is described in the following subsection.

D. Global Optimization

The global frames G are the camera images and/or the LiDAR point-clouds that are observed by multiple sensors, and they are used as both intrinsic and extrinsic cost functions. The local frames L are the other camera images which are observed by single cameras only. Fig. 2 shows examples of the global and local frames. We use the intrinsic and extrinsic parameters by Eq. 1 as the initial parameters for the global optimization problem.

To use the corner points and/or checkerboard point-clouds in the global frames, a 3D point on the checkerboard ${}^{\mathrm{b}}\mathbf{P}$ in k-th global frame is transformed to i-th sensor coordinate system as follows:

$${}^{\mathrm{s}_i}\mathbf{P}_k = {}^{\mathrm{s}_i}_{\mathrm{w}}\mathbf{T} \star {}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k \star {}^{\mathrm{b}}\mathbf{P}. \tag{2}$$

The corner points in the global frames are used in the corner reprojection errors and in the point-to-plane errors if the frame includes LiDAR point-clouds.

Corner Reprojection Error In Sec. III-B and Sec. III-C, the camera intrinsic and extrinsic parameters are computed using the checkerboard's corner points. In our global optimization problem, the following reprojection cost function is used to further refine the camera parameters. Let ${}^{\mathrm{s}_i}\hat{\mathbf{p}}_k$ be

Algorithm 1: Unified calibration framework

```
Data: Labeled corner points $\{\hat{\mathbf{p}}_k\}$ and a plane
      parameter $\hat{\rho}_k$ for each $k$-th checkerboard
      frame.
Result: Intrinsic $\{\boldsymbol{\kappa}_i\}$, and extrinsic $\{{}^{\mathrm{s}_i}_{\mathrm{w}}\mathbf{T}\}$, $\{{}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k\}$
Intrinsic Calibration [25]–[27]
    return initial $\{\boldsymbol{\kappa}_i\}, \{{}^{\mathrm{s}_i}_{\mathrm{b}}\mathbf{T}_k\}$
Global Pose Initialization (Eq. 1)
    return initial $\{{}^{\mathrm{s}_i}_{\mathrm{w}}\mathbf{T}\}$, $\{{}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k\}$
Global Unified Optimization (Eq. 6)
    Objective function $E$
    for $k$-th checkerboard frame do
        if $k \in G$ then
            $E$ +=
              $\lambda_{\hat{\mathbf{p}}} \sum_{i \in C} \sum_{\hat{\mathbf{p}}} \varepsilon_{\hat{\mathbf{p}},k,i} + \lambda_{\hat{\boldsymbol{\rho}}} \sum_{j \notin C} \varepsilon_{\hat{\boldsymbol{\rho}},k,j}$
        else if $k \in L$ then
            $E \mathrel{+}= \lambda_L \sum_{i \in C} \sum_{\hat{\mathbf{p}}} \varepsilon_{\hat{\mathbf{p}},k,i}$
        end
    end
    return $\{\boldsymbol{\kappa}_i\}$, $\{{}^{\mathrm{s}}_{\mathrm{w}}\mathbf{T}\}$, $\{{}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k\}$ minimizing $E$
end
```

a 2D corner point of the k-th global frame captured by i-th camera.

Traditional calibration methods measure the 2D pixel distance error between the labeled point and the projected checkerboard point:

$$\|{}^{\mathrm{s}_i}\hat{\mathbf{p}}_k - \Pi_i({}^{\mathrm{s}_i}\mathbf{P}_k)\|^2, \tag{3}$$

where ${}^{\mathrm{s}_i}\mathbf{P}_k$ is computed by Eq. 2. However, in highly distorted raw images by >180° FOV lenses, the center and peripheral regions have different ray density [28]. To fairly reflect the reprojection error, we use the ray distance as the corner reprojection error $\varepsilon_{\hat{\mathbf{p}},k,i}$:

$$\varepsilon_{\hat{\mathbf{p}},k,i} = \|\Pi_i^{-1}({}^{\mathrm{s}_i}\hat{\mathbf{p}}_k) - \overline{{}^{\mathrm{s}_i}\mathbf{P}_k}\|^2, \tag{4}$$

where $\Pi^{-1}(\cdot)$ is the inverse camera projection function, and $\overline{\mathbf{P}}$ is a unit ray $\mathbf{P}/\|\mathbf{P}\|$. Note that $\varepsilon_{\hat{\mathbf{p}},k,i}$ involves the parameters ${}^{\mathrm{s}_i}_{\mathrm{w}}\mathbf{T}$, ${}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k$, and $\boldsymbol{\kappa}_i$.

Point-to-plane Distance Error We model the checkerboard plane as a 3D plane, whose equation is estimated from the LiDAR point-cloud using a PCA (principal component analysis) and RANSAC algorithm [29] to deal with the noisy points in and around the checkerboard. For k-th global frame captured by j-th LiDAR, we acquire the plane parameters ${}^{\mathrm{s}_j}\hat{\boldsymbol{\rho}}_k = \left[{}^{\mathrm{s}_j}\hat{\mathbf{n}}_k^\top, {}^{\mathrm{s}_j}\hat{d}_k\right]^\top$, where $\hat{\mathbf{n}}$ is the unit normal vector of the plane and $\hat{d}$ is the distance from the origin. We use the estimated plane equations for optimizing extrinsic parameters of the LiDARs and the checkerboards by computing all point-to-plane distance error in the checkerboard $\varepsilon_{\hat{\boldsymbol{\rho}},k,j}$ as

$$\varepsilon_{\hat{\boldsymbol{\rho}},k,j} = \sum_{\mathbf{P}_k} \left({}^{\mathrm{s}_j}\hat{\mathbf{n}}_k^\top \, {}^{\mathrm{s}_j}\mathbf{P}_k + {}^{\mathrm{s}_j}\hat{d}_k\right)^2, \tag{5}$$

where ${}^{\mathrm{s}_j}\mathbf{P}_k$ is computed by Eq. 2. Note that $\varepsilon_{\hat{\boldsymbol{\rho}},k,j}$ involves the parameters ${}^{\mathrm{s}_j}_{\mathrm{w}}\mathbf{T}$ and ${}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k$.

Local Reprojection Error The local frames are used to optimize the camera intrinsic parameters. In recent sensor systems for autonomous agents, the overlap between sensors are not large or spatially biased (Fig. 2), and it is too unstable to update camera intrinsics using only the global frames. We utilize the local frames to prevent the overfitting of the camera intrinsic parameters to the global frames by the following local reprojection error $\varepsilon_{\hat{\mathbf{p}},k,i}$ for a corner point $\hat{\mathbf{p}}$ of k-th local frame captured by i-th camera is same as the corner reprojection error defined in Eq. 4.

Unified Optimization Problem The unified optimization problem includes the global and local reprojection errors and the point-to-plane distance errors:

$$\min_{(\boldsymbol{\kappa}_i,\, {}^{\mathrm{s}}_{\mathrm{w}}\mathbf{T},\, {}^{\mathrm{w}}_{\mathrm{b}}\mathbf{T}_k)} \sum_{k \in G} \left(\lambda_{\hat{\mathbf{p}}} \sum_{i \in C} \sum_{\hat{\mathbf{p}}} \varepsilon_{\hat{\mathbf{p}},k,i} + \lambda_{\hat{\boldsymbol{\rho}}} \sum_{j \notin C} \varepsilon_{\hat{\boldsymbol{\rho}},k,j} \right) + \lambda_L \sum_{k \in L} \sum_{i \in C} \sum_{\hat{\mathbf{p}}} \varepsilon_{\hat{\mathbf{p}},k,i}, \tag{6}$$

where C is the set of cameras, and $\lambda_{\hat{\mathbf{p}}}$, $\lambda_{\hat{\boldsymbol{\rho}}}$ and λ_L are the weights for global reprojection errors, global point-to-plane errors, and local reprojection errors, respectively. The final calibration parameters are determined by minimizing the above non-linear optimization problem. The overall procedure of our method is described in Algorithm 1.

IV. EXPERIMENTAL RESULTS

A. Experimental Setup

We validate our proposed method on both real-world (Small and Big) and synthetic data. For the real data, we use four cameras with 220° FOV fisheye lenses mounted at the corners of the rectangular rig to perform an omnidirectional sensing [30]–[32] and two Velodyne VLP-16 LiDARs with 16 vertical channels. $4 \times (1600 \times 1532)$ gray images per frame are captured while being synchronized by the software trigger, and $2 \times (\sim$24K) LiDAR points are measured while rotating in 600 RPM.

The Small rig is a square-shaped rig (30×30 mm) for drones or robots, and only one LiDAR is mounted at the center of the rig. For automobiles, the Big rig has an extremely wide-baseline (1500×2300 mm), and two LiDARs are mounted near the center of each long axis. We use a checkerboard with 12×10 grids each of which is 60×60 mm for both data. We also generate a synthetic dataset for quantitative evaluation via a rendering software, Blender, and the dataset mimics our Big rig setup as well as the size of the checkerboard. The synthetic dataset consists of $4 \times (800 \times 768)$ fisheye images with 220° FOV and $2 \times (\sim$26K) LiDAR points with 16 vertical channels, and we apply the zero-mean Gaussian noise (with $\sigma = 30$ mm) to the LiDAR point's ground-truth depth for a realistic simulation.

We align the world coordinate system with the first camera coordinate system in the global pose initialization

TABLE I: **Quantitative comparison of optimization frameworks.** We compare our method with multi-stage framework: optimizing pairwise poses and merging them. After calibrating intrinsic, we optimize pairwise poses (cam1-2-3-4, cam2-LiDAR2, cam4-LiDAR1) and merge them into the global (cam1) coordinate system. We also calibrate cameras pose via proposed global optimization and merge cam-LiDAR poses.

	Translation error E_t (mm)						Rotation error E_r (°)					
	cam$_2$	cam$_3$	cam$_4$	LiDAR$_1$	LiDAR$_2$	Avg.	cam$_2$	cam$_3$	cam$_4$	LiDAR$_1$	LiDAR$_2$	Avg.
Global pose initialization	3.10	12.11	8.41	777.67	780.82	-	0.20	0.32	0.55	38.76	145.11	-
Merging all pairwise poses	**2.10**	6.28	9.67	18.67	8.82	9.11	**0.17**	0.29	0.54	**0.23**	0.95	**0.43**
Merging camera rig to LiDARs	3.08	5.59	**5.14**	17.64	10.08	8.31	0.18	0.30	0.58	0.32	**0.94**	0.47
Global optimization	3.15	**4.80**	5.86	**12.20**	**4.82**	**6.17**	0.21	**0.23**	**0.46**	0.24	1.93	0.61

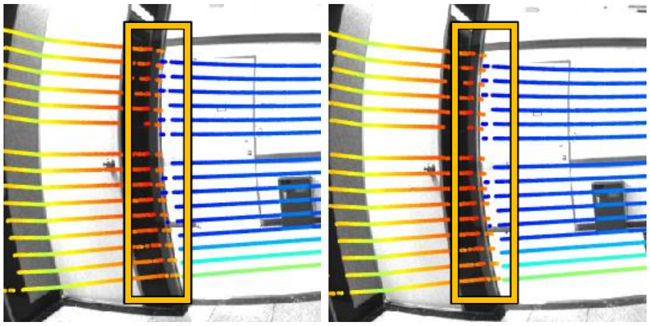


Fig. 3: **Qualitative results of our global optimization (left) and simple composition of the pairwise poses (right) on Small dataset.** The LiDAR points on the front pillar are correctly projected (orange points) after global optimization.

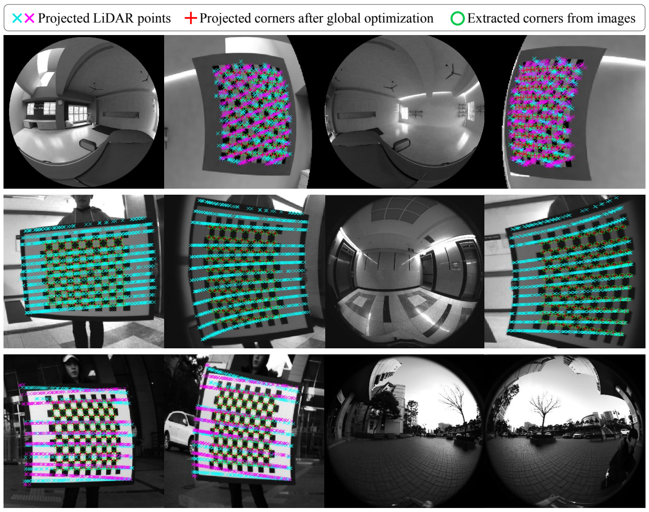

××Projected LiDAR points
+Projected corners after global optimization
○Extracted corners from images

Fig. 4: **Projected checkerboard points after optimization.** The LiDAR points in labeled planes and the checkerboard points are correctly projected onto the input images after calibration. (Cyan: LiDAR$_1$, Magenta: LiDAR$_2$)

TABLE II: After global optimization, we validate the calibration results by the root mean squared error (RMSE) of the corner pixel distances (Eq. 3), and the mean absolute error (MAE) of the point-to-plane distances (Eq. 5).

Dataset	Reprojection RMSE (px)				Distance MAE (mm)	
	cam$_1$	cam$_2$	cam$_3$	cam$_4$	LiDAR$_1$	LiDAR$_2$
Small	0.41	0.42	0.34	0.47	6.43	-
Big	0.40	0.47	0.25	0.19	39.37	29.08
Synthetic	0.14	0.14	0.16	0.17	6.65	11.02

(Sec. III-C) so that ${}^{\mathrm{s}_1}_{\mathrm{w}}\mathbf{T}$ is the identity transformation, and fix it in the optimization process. The initial poses of LiDARs are set to the center of the initialized rig pose. The weight parameters $\lambda_{\hat{\mathbf{p}}}$ and λ_L for corner reprojection errors are set to 5×10^6, and $\lambda_{\hat{\boldsymbol{\rho}}}$ is set to 10^{-4}.

For quantitative evaluation, the translation error E_t is measured by the euclidean distance between the estimated and ground-truth sensor position, and the rotation error E_r is measured by the magnitude of the relative rotation parameter between the estimated and ground-truth rotation:

$$E_t = \|\mathbf{t} - \mathbf{t}^*\| \text{ and } \; E_r = \|\bar{\mathbf{r}}\|, R(\bar{\mathbf{r}}) = R(\mathbf{r})^\top R(\mathbf{r}^*), \quad (7)$$

where $(\mathbf{r}^\top, \mathbf{t}^\top)^\top$ is the estimated pose and $(\mathbf{r}^{*\top}, \mathbf{t}^{*\top})^\top$ is the ground-truth pose.

B. Evaluation of Global Optimization

We evaluate our proposed unified global optimization framework quantitatively on the synthetic dataset and qualitatively on the Small dataset. Similar to the traditional multi-sensor calibrations, we estimate the pairwise poses $\{{}^{\mathrm{s}_1}_{\mathrm{s}_2}\mathbf{T}, {}^{\mathrm{s}_2}_{\mathrm{s}_3}\mathbf{T}, {}^{\mathrm{s}_3}_{\mathrm{s}_4}\mathbf{T}, ...\}$ and combine them to compute the global poses of the sensors. We also test the intermediate version that optimizes the camera parameters only. Tbl. I shows quantitative results of each optimization framework. After applying our framework to cameras, the translation error of the lastly merged cam$_4$ is greatly decreased, and the translation errors of LiDARs are also decreased after applying global optimization. Fig. 3 shows qualitative comparison of our method on Small dataset. The 3D LiDAR points on the front pillar are projected correctly in the image after the proposed global optimization, whereas using extrinsic from combined pairwise poses they are misaligned.

C. Validation of the Proposed Framework

In the previous subsection, we quantitatively measure the calibration error of our algorithm on the synthetic dataset. Fig. 5 shows qualitative calibration results on synthetic and real-world datasets. We project 3D LiDAR points onto the input images, and as shown in Fig. 5, most of the points from close to far depths are well aligned, whereas in Big dataset, a small number of misaligned points exists due to the extremely wide-baseline setup of the sensors. The optimization result is also detailed in Tbl. II and Fig. 4, and similar to the qualitative results, the plane distance errors

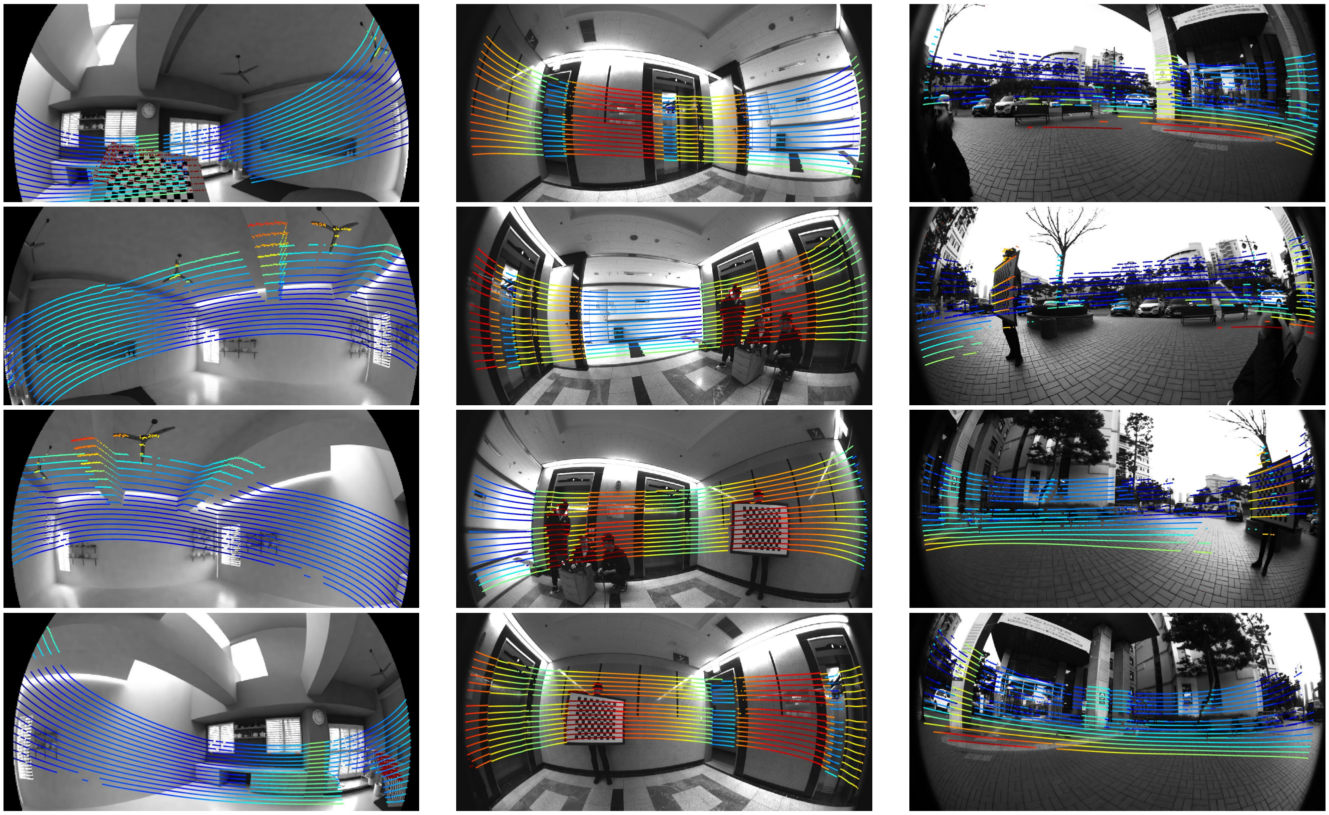


Fig. 5: **Qualitative calibration results.** From left: synthetic, Small, and Big datasets. The LiDAR points are projected onto the images, and the colors of the projected points encode depths. Note that the points from the background can be projected onto occluding foreground objects due to the wide-baseline setup.

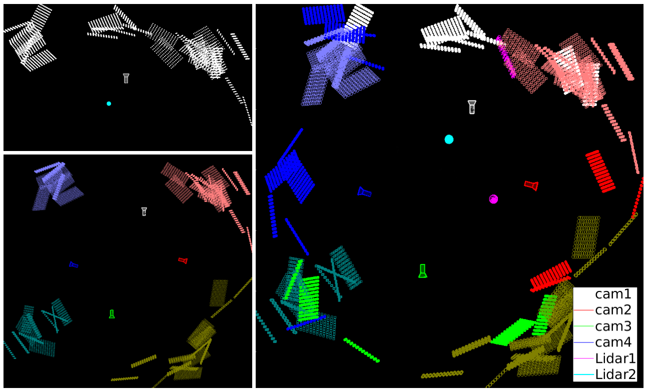
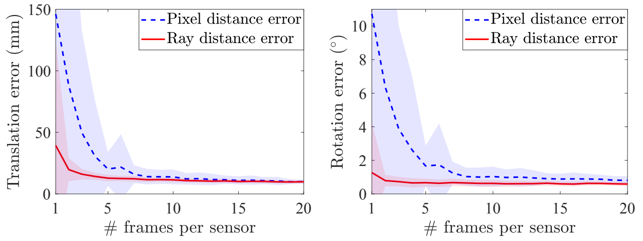


Fig. 6: **Calibration results on various sensor layouts.** Left: camera to LiDAR (top), multi-camera (bottom). Right: multi-camera multi-LiDAR. Global poses of the sensors and observed global checkerboard frames are illustrated.

in Big dataset is higher than the other datasets. Fig. 6 also shows qualitative results of our methods on various sensor layouts. Calibrated global poses of the sensors and the global checkerboards frames are illustrated.

We also measure the calibration errors according to the number of observed frames — we randomly sample a subset of the observed frames while preserving connectivity between the sensors and measure the extrinsic errors. The extrinsic errors are averaged over randomly sampled 100 results. As shown in Fig. 7, the errors converge if more than 10 frames per sensor are used. We also compare the pixel distance error and the ray distance error used in optimization (Sec. III-D), and the ray distance error yields better performance especially for the rotation.



(a) Translation error

(b) Rotation error

Fig. 7: **Mean and standard deviation of the extrinsic errors on the synthetic dataset according to the number of frames.** The errors decrease as more frames are used, and they converge after using 10 frames per sensor. The blue and red denotes the ray distance error and the pixel distance error respectively. The results show that the ray distance is more accurate and stable.

V. CONCLUSIONS

In this paper, we present a novel approach to unified calibration for multi-camera and multi-LiDAR sensor systems using a single checkerboard. The proposed framework can handle the complete calibration of various sensor layouts and camera models. The intrinsic and extrinsic parameters of the sensors estimated by the conventional methods are used as the initial values of our non-linear optimization problem, and all observations are formulated as the reprojection and point-to-plane error terms. The final calibration parameters are found by solving the optimization problem. Through this approach, the calibration errors are reduced significantly compared to the traditional calibration approaches. Through extensive experiments using synthetic and real-world datasets, we show that the proposed unified calibration method works very well for challenging configurations only using a single checkerboard.

ACKNOWLEDGMENT

This research was supported in part by Next-Generation Information Computing Development program through National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT (NRF-2017M3C4A7069369), and the NRF grant funded by the Korea government (MSIT) (NRF-2019R1A4A1029800).

REFERENCES

[1] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1907–1915.

[2] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3d proposal generation and object detection from view aggregation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–8.

[3] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "Std: Sparse-to-dense 3d object detector for point cloud," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 1951–1960.

[4] J. Graeter, A. Wilczynski, and M. Lauer, "Limo: Lidar-monocular visual odometry," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 7872–7879.

[5] Y.-S. Shin, Y. S. Park, and A. Kim, "Direct visual slam using sparse depth for camera-lidar system," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1–8.

[6] A. Geiger, F. Moosmann, Ö. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 3936–3943.

[7] Z. Xiao, H. Li, D. Zhou, Y. Dai, and B. Dai, "Accurate extrinsic calibration between monocular camera and sparse 3d lidar points without markers," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 424–429.

[8] Y. Park, S. Yun, C. S. Won, K. Cho, K. Um, and S. Sim, "Calibration between color camera and 3d lidar instruments with a polygonal planar board," *Sensors*, vol. 14, no. 3, pp. 5333–5353, 2014.

[9] D. Tang, T. Hu, L. Shen, Z. Ma, and C. Pan, "Apriltag array-aided extrinsic calibration of camera–laser multi-sensor system," *Robotics and biomimetics*, vol. 3, no. 1, p. 13, 2016.

[10] 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 *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 605–610.

[11] L. Heng, B. Choi, Z. Cui, M. Geppert, S. Hu, B. Kuan, P. Liu, R. Nguyen, Y. C. Yeo, A. Geiger, *et al.*, "Project autovision: Localization and 3d scene perception for an autonomous vehicle with a multi-camera system," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 4695–4702.

[12] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.

[13] Q. Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, 2004, pp. 2301–2306.

[14] D. Herrera, J. Kannala, and J. Heikkilä, "Joint depth and color camera calibration with distortion correction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 2058–2064, 2012.

[15] 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.

[16] B. Li, L. Heng, K. Koser, and M. Pollefeys, "A multiple-camera system calibration toolbox using a feature descriptor-based calibration pattern," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 1301–1307.

[17] Z. Chai, Y. Sun, and Z. Xiong, "A novel method for lidar camera calibration by plane fitting," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2018, pp. 286–291.

[18] J. Jeong, Y. Cho, and A. Kim, "The road is enough! extrinsic calibration of non-overlapping stereo camera and lidar using road information," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2831–2838, 2019.

[19] 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 *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 1499–1504.

[20] Z. Taylor and J. Nieto, "Automatic calibration of lidar and camera images using normalized mutual information," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[21] W. Zhao, D. Nister, and S. Hsu, "Alignment of continuous video onto 3d point clouds," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 8, pp. 1305–1318, 2005.

[22] S. Bileschi, "Fully automatic calibration of lidar and video streams from a vehicle," in *Proceedings of the IEEE International Conference on Computer Vision Workshops, ICCV Workshops (ICCVW)*, 2009, pp. 1457–1464.

[23] J. Levinson and S. Thrun, "Automatic online calibration of cameras and lasers." in *Proceedings of the Robotics: Science and Systems*, vol. 2, 2013, p. 7.

[24] F. Vasconcelos, J. P. Barreto, and U. Nunes, "A minimal solution for the extrinsic calibration of a camera and a laser-rangefinder," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2097–2107, 2012.

[25] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[26] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *Proceedings of the IEEE International Conference on Computer Vision Systems (ICVS)*, 2006, pp. 45–45.

[27] S. Urban, J. Leitloff, and S. Hinz, "Improved wide-angle, fisheye and omnidirectional camera calibration," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 108, pp. 72–79, 2015.

[28] Z. Zhang, H. Rebecq, C. Forster, and D. Scaramuzza, "Benefit of large field-of-view cameras for visual odometry," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 801–808.

[29] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[30] C. Won, J. Ryu, and J. Lim, "Sweepnet: Wide-baseline omnidirectional depth estimation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6073–6079.

[31] H. Seok and J. Lim, "Rovo: Robust omnidirectional visual odometry for wide-baseline wide-fov camera systems," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6344–6350.

[32] C. Won, J. Ryu, and J. Lim, "Omnimvs: End-to-end learning for omnidirectional stereo matching," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 8987–8996.