2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Infrastructure Based Calibration of a Multi-Camera and Multi-LiDAR System Using Apriltags

Yuanfan Xie[1,2], Rui Shao[1,2], Popo Gui[1,2] , Bo Li[1,2] and Liang Wang[1]

Abstract— **This paper proposes a framework for the calibration of a multi-camera and multi-LiDAR system. It utilizes the Apriltags to build a calibration environment to solve the poses and extrinsics of cameras and then deploy ICP-like algorithm to solve the extrinsics of LiDARs. Compared to previous extrinsic calibration methods, this proposed framework naturally applies to systems with different numbers and configurations of cameras and LiDARs. The calibration procedure produces extrinsics not only accurately and robustly, but efficiently without inconvenient human manipulation. The framework provides general, efficient and standardizable calibration solution for autonomous driving platforms with cameras and LiDARs.**

I. Introduction

Modern robotic systems including autonomous driving vehicles and unmaned aerial vehicles tend to equip multiple cameras or LiDARs for environment perception like SLAM or object detection. These tasks require accurate extrinsic calibration between different cameras and LiDARs. Extrinsic calibration estimates the relative extrinsic transform between sensors such that data captured by different sensors can be interpreted under a unified coordinate system. This procedure plays an important role in robotic applications. For example, provided accurate extrinsics, LiDAR captured point cloud data can be aligned with images to obtain depth or localization info of objects detected in images.

As will be mentioned in Section II, although a variety of methods have been proposed to calibrate camera-camera, camera-LiDAR or LiDAR-LiDAR extrinsics, in practice it is still inconvenient to calibrate a general multi-camera and multi-LiDAR system. Existing approaches mostly assemble different methods to calibrate different components of the system separately and requires many manual work. For example, one may have to randomly move a chessboard to calibrate the extrinsics of stereo cameras first and then a polygon to calibrate the extrinsics between cameras and LiDARs [1]. For sensors without overlapping FoV, the hand-eye calibration method might also be used [2]. In the practical development of robotic systems, e.g. the autonomous driving vehicles, we find such calibration procedure is not only inconvenient for academic research, but also difficult to standardize for industrial production. Therefore in this paper, we seek to develop a unified extrinsic calibration method which easily adapts to different multi-camera and multi-LiDAR systems and can be deployed in a standardized procedure.

[1] The authors are with the Baidu Inc. {shaorui, xieyuanfan, guipopo, libo24, wangliang18}@baidu.com
[2] Equal contribution as joint first authors.

II. Related Work

A number of approaches have been proposed in the past decades to calibrate robotic systems with different configurations of cameras and LiDARs. The so-called supervised calibration is a main category of these methods. The calibration procedures exploit certain reference objects with known shape or pattern to formulate constraints on the extrinsic parameters. For example, the well known chessboard object are widely used not only for camera intrinsic calibration [3] but also for camera-LiDAR extrinsic calibration[4][5], [6]. Besides chessboards, textured image [7], polygons [1] and Apriltags [8] are also suggested in previous studies in calibrating camera-camera or camera-LiDAR extrinsics. For LiDAR-LiDAR extrinsic calibration, shape primitives like planes [9] and poles [10] can be used as reference objects.

In addition to supervised calibration, researches also have been drawn on exploiting natural scenes for calibration in an unsupervised manner. Heng et. al. estimate camera-camera extrinsics by structure from motion with the extrnisics as part of bundle adjustment parameters[11]. With specific environment model available as infrastructure, the precision can be further enhanced [12]. For camera-LiDAR extrinsic calibration, some unsupervised calibration works [13], [14], [15] project point cloud into image space and optimize the alignment between point projection and the color image. Hand-eye calibration and its variations are also used in some previous works to estimate extrinsics from sensor trajectories [2].

The supervised calibration with carefully designed reference objects has several advantages over unsupervised methods in precision and robustness. Compared with keypoint detection on nature features e.g. [11], [12], the detection of reference object such as chessboard corners in a controlled environment is much more robust. The sub-pixel level coordinates of the detected corners is generally more accurate than those of the Harris corners from nature images. In addition, the known sizes and coordinates of the reference objects provide extra constraints and information for solving the calibration extrinsics.

The previous extrinsic calibration methods mostly focus on calibrating a specific portion of the multiple sensor systems, e.g. camera-camera extrinsic calibration [7], [4] or camera-LiDAR extrinsic calibration [5], [6]. However, for a general sensor system with multiple cameras and multiple LiDARs, it is usually inconvenient and even impractical to deploy different lab-scale hand-held reference objects and calibrate camera-camera pairs and camera-LiDAR pairs separately.

978-1-5386-4451-5/18/$31.00 ©2018 IEEE

For possible industrial application, e.g. the production of autonomous driving vehicles in the future, the previous methods are far inefficient and difficult for standardization.

In this paper, we illustrate the recent progress of our exploration in developing a standardizable extrinsic calibration procedure for multi-camera and multi-LiDAR system especially the autonomous driving vehicles. Its contribution comes mainly in two aspects:

1) We design a unified reference environment (object) which supports the extrinsic calibration of both cameras and LiDARs. The environment is easy to build by just pasting printed bar codes, e.g. AprilTags [16] on multiple walls in a garage alike room. The AprilTag is a series of bar code widely used in robotics with sub-pixel detection accuracy.
2) We design a calibration framework which naturally supports the extrinsic calibration of general multi-camera and multi-LiDAR systems based on the proposed reference environment. The framework applies to any mounting configurations of the sensors positions and orientation. Specifically, different from previous methods [7], [5], [6], it does not require overlapping FoV between any sensors.

By exploiting a large number of reference points from the environment, the proposed procedure calibrates sensor extrinsics with high precision compared to related methods. In addition, the procedure can be standardized easily for industrial production without requiring nonstandard human manipulation. In the following of this paper, we first illustrate the setup of the reference environment and the calibration operation procedure in Section III. Section V and VI illustrate the extrinsic calibration of cameras and LiDARs respectively. Experiments are presented in Section VII.

III. Environment Setup and Calibration Operation

The proposed calibration environment is built in a garage alike room, with printed AprilTags pasted on the walls. To guarantee that sensors oriented at different directions can observe AprilTags (not necessarily the same tags) at the same time, it is suggested that three or more walls of different directions to be pasted with AprilTags. Figure 3a shows a sample environment built in our experiment.

Note that in some cases, the printed AprilTags might appear too large in some long focal lens cameras, which results in very few observed tags in images. This issue can be solved by pasting Apriltags of different scales mixed. Alternatively, bar codes made up of multiple scaled components, e.g. the factual tags [17] can be deployed.

To obtain the 3D coordinates of all the AprilTags in a unified coordinate system, we use a 3D scanner to capture the full 3D model of the room. AprilTags are then labeled from the model and thus the 3D coordinates of all corners are obtained. This paper uses $X \in \Theta$ to denote the 3D coordinates of an AprilTag corner and Θ as the set of all corners. All 3D corner coordinates are indexed by the code of its belonging AprilTag. Thus when a AprilTag is detected, its corresponding 3D coordinates in the environment can be retrieved. In the case without a 3D scanner, the 3D coordinates of AprilTag corners can be reconstructed by techniques such as Structure from Motion. Note that the proposed setup does not require the planarity of the wall, unlike the chessboard based calibration method, which assumes that corners all lies on the $z = 0$ plane.

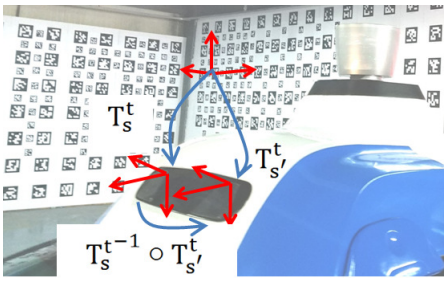


Fig. 1. A sample setup of the calibration environment. AprilTags are posted on walls to produce accurate features. See Section IV for the detailed definition of the sensor coordinate system.

The calibration procedure of an autonomous vehicle in such environment is executed by driving the vehicle inside the room around and capturing synchronized sensor data. For sensors without synchronized clock signal or trigger, synchronized data can be captured by fully stopping the vehicle at each suitable position and then triggering the capture. As the reference environment is fixed, specified vehicle trajectory and data capturing position can be designed, which allows standardized calibration operation for manufacture production.

IV. Approach Overview

Denote T^t as the transform from world frame to the vehicle body frame at time t. T_s as the extrinsics of sensor S_s. Thus the transform from world frame to the sensor frame s at time t can be written as:

$$T_s^t = T_s \circ T^t \tag{1}$$

Without losing generality, in this paper we assume the vehicle body frame to be aligned with the first camera, i.e. $T_0 = I$. Thus the sensor extrinsics T_s denotes the relative extrinsic pose between sensor S_0 and sensor S_s. Figure 1 provides a simple illustration of the coordinate systems at a given timestamp t.

Denote $\ell(T^t, T_s, \Theta)$ as the observation error of sensor S_s for environment Θ at time t. The calibration procedure is formed to minimize the summation observation error over sensors and data frames:

$$\sum_s \sum_t \ell(T^t, T_s, \Theta) \tag{2}$$

The proposed approach optimizes the extrinsics of cameras and LiDARs respectively, with losses designed for camera data and LiDAR data. In the following sections, the calibration procedure is illustrated as the initialization stage and the optimization stage for T^t and T_s.

V. Camera Extrinsics Calibration

A. Initialization

Consider the 3D coordinates of each AprilTag corner $X \in \Theta$. At time t its detected corresponding 2D corners in camera s is denoted as x_s^t. Recall the pinhole camera projection model with K as intrinsics. Denote the projection function as p for example:

$$\begin{aligned} x_s^t &\sim p(K, T_s^t, X) = K \circ (R_s^t \circ X + t_s^t) \\ T_s^t &= \begin{bmatrix} R_s^t & t_s^t \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{3}$$

x_s^t is a 2D homogeneous coordinates with the last element as 1. The coordinates X of the AprilTag corner are known from the scanned 3D environment model. The symbol $\sim$ here denotes equality up to a scale. K is assumed to be known using e.g. [18]. Note that for fish-eye or catadioptric camera distortion model, the projection procedure is analogous and can be found from e.g. [7].

At each timestamp t, we denote the frame of data captured by sensor S_s as f_s^t and denote $F^t = \{f_s^t | s = 1, 2, \cdots\}$ as the set of data frame of all sensors at time t. If a camera observes a 2D AprilTag corner x_s^t, its 3D correspondence X can be easily queried using its AprilTag code. Given enough correspondence pairs, it is not difficult to exploit the PnP algorithm [19] to solve T_s^t from (3). Consider two cameras s and s', both of which observe enough AprilTag corners to solve T_s^t and $T_{s'}^t$. The relative extrinsic pose between camera s and s' can thus be obtained as ${T_s^t}^{-1} \circ T_{s'}^t$ as shown in Figure 1.

We extend this procedure to compute the extrinsics between multiple cameras by constructing a bipartite pose graph as shown in Figure 2. This graph denotes similar concepts with [7]. Denote each camera S_s as a camera vertex, which uniquely corresponds to its extrinsics T^s. Denote F_t, the set of the data captured by all cameras at time t as a data vertex. F_t uniquely corresponds to T^t. If a camera s observes enough corners in f_s^t such that T_s^t can be solved, an edge representing f_s^t is connected between F_t and camera s. Each edge in the graph corresponds to the relative pose T_s^t. Composing relative poses corresponding to the edges along a path in the graph produces the relative pose between the two ending vertices of the path. Thus as in the above case, when two cameras s and s' are connected to a same vertex F_t, their relative pose can be obtained along the path S_s–F_t–$S_{s'}$, as $T_{s'}^t \circ {T_s^t}^{-1}$.

The initialization of camera extrinsics requires that all sensor vertices should be connected with each others in a common graph, such that the the relative pose between sensors can be computed along the connected edges. For example in Figure 2, the initial extrinsics T_2 can be obtain as $T_2^2 \circ {T_1^2}^{-1} \circ T_1^0 \circ {T_0^0}^{-1}$ along the path S_0–F_0–S_1–F_2–S_2.

This initialization requires that enough portion of sensors should observe AprilTags at each timestamp such that all cameras can be connected to S_0 in the graph. This is usually not a difficult issue when a vehicle can move freely in the environment with AprilTags pasted on three or more walls.

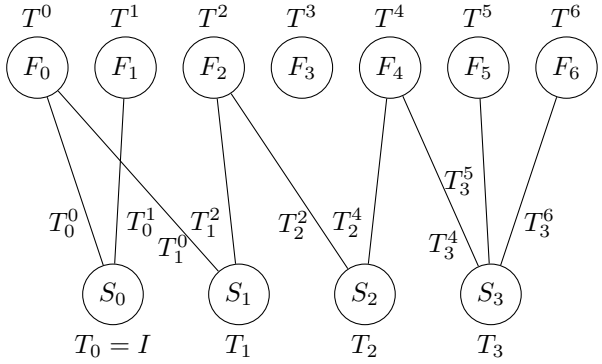


Fig. 2. Bipartite calibration pose graph, with similar definition as in [7]. F_t denotes the collection of the captured data at time t. Each F_t corresponds with a vehicle pose T^t. S_s denotes the sensors, each of which uniquely corresponds to its extrinsics T_s. When a sensor S_s observes enough corners in F_t and solves its pose T_s^t, edge representing T_s^t is drawn between F_t and S_s. If two sensors are connected to a common F_t vertex, e.g. S_0 and S_1 are connected to S_0, the extrinsics can be computed via $T_1^0 \circ {T_0^0}^{-1}$

Due to the estimation error, relative poses along different paths over the graph may not be consistent, which requires further pose graph optimization. However, for initialization of the sensor extrinsics, it is sufficient to greedily find a spanning tree connecting all sensors to sensor S_0 and compose the relative extrinsic pose T_s for other sensors along the tree.

B. Loss formulation

Similar to previous camera calibration methods, e.g. [7], the summarized image-based reprojection error is used as the optimization cost. Both the vehicle poses and camera extrinsics are optimized. Thus (2) is instantiated as

$$\begin{aligned} &\arg\min_{T^t, T_s} \sum_s \sum_t \ell(T^t, T_s, \Theta) \\ &\ell(T^t, T_s, \Theta) = \sum_{X \in \Theta} \|\hat{x}_s^t - x_s^t\|^2 \\ &x_s^t \sim p(K, T_s^t, X) \end{aligned} \tag{4}$$

where $\hat{x}_s^t$ is the 2D projection of a 3D AprilTag corner and x_s^t is the corresponding detected 2D AprilTag corner from the frame t of camera s. We minimize the loss using Levenberg-Marquart algorithm.

VI. LiDAR Extrinsics Calibration

A. Initialization

Following Section V, the pose of vehicle body frame T^t is solved and then can be exploited for solving the extrinsics of LiDARs. For a multi-beam 3D LiDAR s, T_s^t can be estimated via ICP-like algorithms by aligning the LiDAR point cloud data at timestamp t and environment model Θ. The ICP-like algorithm solves the following minimization problem:

$$\arg\min_{T_s^t} \sum_{X \in \Theta} d(T_s^t \circ X, Y) \tag{5}$$

where Y is the nearest LiDAR point to $T_s^t \circ X$ captured by LiDAR s at time t. $d(\cdot, \cdot)$ denotes a distance function

between the two points. We here use the distance

$$d(X,Y) = \begin{cases} n_Y^\top (X-Y) & \|X-Y\| < \sigma \\ 0 & \text{elsewise} \end{cases} \quad (6)$$

from the point-to-plane ICP algorithm, as suggested in [20]. σ is the threshold to exclude outlier correspondences. Since our proposed environment is made up with locally planar walls, this point-to-plane algorithm produces reasonably accurate alignment between the captured point cloud and the scanned 3D model.

Objective (5) is solved by gradient descent method or Newton's method from an initial guess of T_s^t. The initial guess of T_s^t can be obtained in several ways. For example, with T^t solved from Section V, T_s^t can be guessed using $T_s \circ T^t$ with a rough guess of T_s.

Similar to camera extrinsic calibration, the relative LiDAR extrinsics between s and s' can be estimated via $T_s^{t\,-1} \circ T_{s'}^t$. Incorporating with the bipartite calibration pose graph concept as mentioned in Section V, the initial extrinsics T_s for a LiDAR s can be obtained.

Note that (5) provides reasonably accurate initialization of T_s for multi-line 3D LiDARs. For single-line 2D LiDARs or 3D LiDARs with narrow vertical FoV however, ambiguous solutions might exist and (5) may not converge correctly. In this case, we find that a manually measured initialization of T_s can also work for the extrinsic refinement.

B. Loss formulation

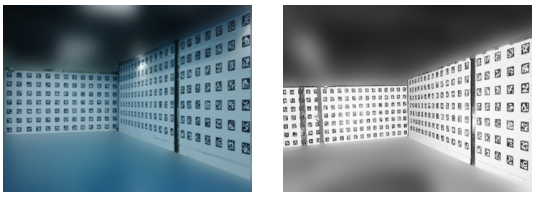
The LiDAR extrinsics are further refined to minimize the ICP-like cost over all LiDARs and all frames as follows.

$$\begin{aligned} &\arg\min_{T_s} \sum_s \sum_t \ell(T^t, T_s, \Theta) \\ &\ell(T^t, T_s, \Theta) = \sum_{X \in \Theta} d(T_s^t \circ X, Y) \\ &T_s^t = T_s \circ T^t \end{aligned} \quad (7)$$

where Y is the nearest LiDAR point to $T_s^t \circ X$ captured by LiDAR s at time t. This cost shares similar form with (5) but accumulates point correspondence distances over multiple LiDARs and multiple timestamps. Point cloud from 2D single-line LiDARs can also be included in the loss formulation. This cost enforces the LiDAR points captured at different time form a environment map which is compact and consistent with Θ. As shown in [9], [21], by capturing data at different vehicle orientation and position, the solution ambiguity for single-line LiDARs mentioned above can be also solved.

VII. Experiments

The proposed approach is experimented in an environment built according to Section III. About 500 AprilTags of series 36h11 are used to build the environment. For better calibration precision and performance analysis, we use high-definition laser scanner to build up the 3D model of the calibration environment. The 3D position of the tag corners are obtained in the 3D model exploiting the intensity of tags. Figure 3 shows an example of the built environment.



(a) (b)

Fig. 3. Sample of Calibration Environment. (a) A photography of our calibration environment. (b) Scanned point cloud.

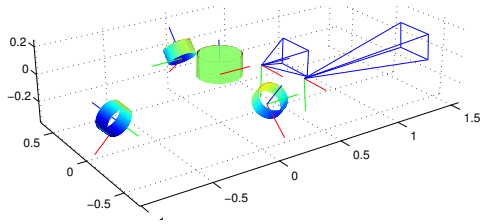


Fig. 4. Sensor pose visualization for an autonomous vehicle with two cameras, one HDL 64 LiDAR and three VLP 16 LiDARs.

The vehicle platform consists of a Velodyne HDL 64E LiDAR, three Velodyne VLP 16 LiDARs and two cameras with one long focal and one short focal. The resolution of both cameras is 1280×960. The intrinsics are well calibrated before installation using [18]. The calibration procedure uses data frames captured from 30 arbitrary vehicle poses in the environment.

Figure 4 visualizes the calibrated sensor poses. To evaluate the stability of the approach, we repeat the calibration procedure with different data for 100 times. Figure 5 shows the statistics of the calibrated extrinsic parameters. The standard deviation of calibrated translation is less than 2 cm and that of orientation angles less than 0.5 deg.

The proposed approach calibrates the sensor system in a single unified framework. Since most of the previous calibration methods only calibrate a certain portion of the multi-camera and multi-LiDAR system, in the experiments we compare each part of calibration results with corresponding reference literature respectively.

A. Camera-Camera Extrinsic Calibration

A main category of modern multi-camera extrinsic calibration approaches [18], [7] solve extrinsics by minimizing reprojection error in a manner similar to (4). Thus with keypoints of similar quality and distribution, these methods calibrate in very similar precision while the difference mainly comes in the applicability for different sensor configuration.

In the experiments, we compare the stereo camera extrinsic calibration with the most commonly used stereo calibration function in OpenCV. Both of the methods directly minimize the objective cost made of reprojection errors. The OpenCV method uses chessboard as the reference object and assumes corners to be on the same plane. The proposed method measures the full 3D coordinates of the AprilTag corners.

For the above camera pair, the OpenCV approach requires that a chessboard to be detected by both the long focal

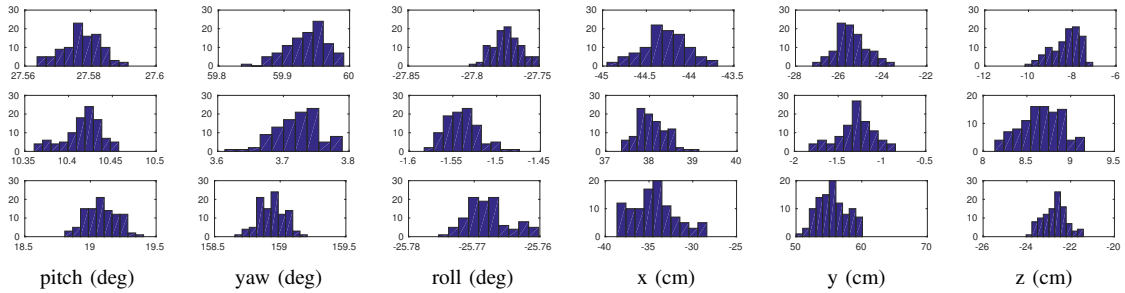
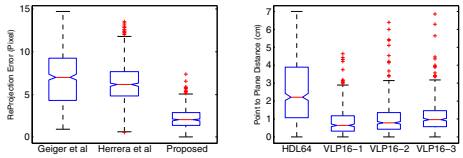


Fig. 5. Stability statistics of the calibrated extrinsic parameters in repeated experiments. The first row corresponds to the HDL 64E LiDAR. The second row corresponds to the long focal camera and the third row to the left VLP 16 LiDAR. Due to the space limitation, the statistics of the rest sensors are not shown but similar with the above.

(a) Geiger et al (b) Herrera et al (c) Proposed

Fig. 6. Sample data used for experiments of camera-LiDAR calibration.



(a) (b)

Fig. 7. (a) Statistics of the reprojection errors from LiDAR to camera.(b) Statistics of point-plane distance from VLP 16 to HDL 64E.

camera and the short focal camera. Thus the chessboard appears very small in the short focal images. In this case, it is difficult to guarantee the chessboard corners to be uniformly distributed in the short focal images. In our experiment, 20 frames of extra data are collected from the environment and used as test data to evaluate the reprojection error of the camera extrinsic calibration. The average reprojection error on the short focal images for OpenCV calibration functionality is 3.6pix and 1.4pix for the proposed approach. The reprojection error on the long focal images for OpenCV is 7.1pix and 2.6pix for the proposed approach.

Besides the above camera pair, the proposed approach is also compared against OpenCV on a standard BumbleBee 3 stereo camera set, with baseline of 24 cm from manual. The two methods performs similarly as shown in Table I.

Method	Baseline (cm)	Reprojection Error (px)	
		Calib Data	Valid Data
OpenCV	24.076	0.64	1.27
Proposed	23.999	0.98	1.23

TABLE I
STEREO CALIBRATION ERROR FOR BUMBLEBEE 3

B. Camera-LiDAR Extrinsic Calibration

We compare the camera-LiDAR calibration precision of the proposed approach with [4], [5]. [4] calibrates camera-LiDAR extrinsics with one shot of data frame. We built the scene with similar configuration of 6 chessboards with different poses, as shown in Figure 6a. [5] is originally proposed for camera-kinect calibration and adapted to LiDAR data in the experiments. It uses planar chessboard as reference object as shown in Figure 6b. 30 frames are collected as input for [5]. Sample image data used for the three methods are shown in Figure 6.

The three approaches are compared by evaluating the reprojection error from the Velodyne HDL 64E LiDAR to the cameras. 20 extra frames of *validation data* are captured from the proposed Apriltags based calibration environment for evalutaion. The Velodyne HDL 64E point cloud in each test data frame is firstly aligned with the environment point cloud model Θ so that coordinates of the Apriltag corners are transformed to the HDL 64E frame. The coordinates of the Apriltag corners are then projected into cameras using the calibrated extrinsics and the reprojection error with the detected Apriltag corners can be evaluated. Note that the alignment error between the HDL 64E point cloud and the environment model is comparably small and does not affect the comparison results.

Figure 7a shows the statistics of the reprojection error of the three methods. [4], [5] calibrate camera-LiDAR extrinsics in a similar manner of registering plane structures between cameras and LiDARs directly. [4] uses only one frame of data which inevitably affects the calibration precision, since it is difficult to construct a scene with chessboard corners uniformaly and densely distributed. [5] produces stabler by taking more data frames. The proposed approach makes use of the known environment point cloud model by registering camera data and LiDAR data to the model respectively. The HDL 64 point cloud data are registered to the environment model by ICP-like manner of (5), which generates much more constraints and stabler results than [5] by registering planes.

C. LiDAR-LiDAR Extrinsic Calibration

The extrinsics between HDL 64E LiDAR and three VLP 16 LiDARs are evaluated by point-plane distance. Point cloud from all LiDARs are transformed into the calibration environment frame according to the calibrated extrinsics. We

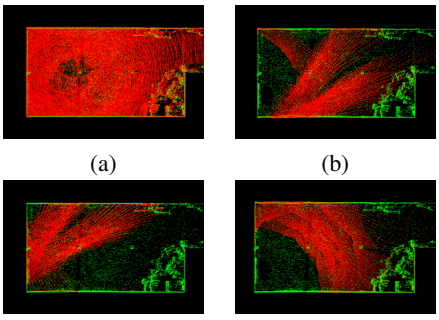


Fig. 8. Calibration Results of a HDL 64E LiDAR and Three VLP 16 LiDARs. Visualized by stitching 5 LiDAR point cloud frames using the estimated pose and extrinsics as red dots. The environment point cloud model is also visualized as green dots for reference. (a) HDL 64E LiDAR. (b) VLP 16 Left LiDAR. (c) VLP 16 Right LiDAR. (d) VLP 16 Back LiDAR.

use the environment point cloud as ground truth. Then the walls in calibration environment can be estimated as planes the point cloud. Figure 8 visualize the calibration result by transforming the LiDAR point cloud (red) into the environment coordinate system (green) using the estimated vehicle poses and extrinsics. The distance between the LiDAR points to the estimated planes is then evaluated. Figure 7b shows the boxplot of the point-plane distance. While the typical distance measurement accuracy of Velodyne VLP 16 is 3cm, the mean distance is 1.9 cm. Note that the outlier noise in Figure 7b is mainly due to the reflection noise.

VIII. Conclusion

Although many methods have been proposed in the recent years to calibrate camera and LiDAR systems on lab scale, it is widely admitted that calibrating a complex sensor system is non-trivial and requires specific algorithm adaption. In the scenario of autonomous driving, this inevitably becomes a bottleneck for the standardization of the vehicle platforms production. In this paper, we explore to standardize the calibration procedure of the multi-camera and multi-LiDAR system in a unified framework. The framework not only produces reliable and accuracy calibration extrinsics but also naturally applies to arbitrary configuration of cameras and LiDARs. With more industrial organizations investing efforts in turning autonomous vehicle into commercial products, the proposed method can be turned into an automatic and standard procedure for industrial production.

Acknowledgment

The authors would like to thank the help from Yanwei Du, Jing Zhang and Fei Qing.

References

[1] 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.

[2] C. X. Guo and S. I. Roumeliotis, "An analytical least-squares solution to the line scan lidar-camera extrinsic calibration problem," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2943–2948.

[3] C. Zhang and Z. Zhang, "Calibration between depth and color sensors for commodity depth cameras," in *Computer Vision and Machine Learning with RGB-D Sensors*. Springer, 2014, pp. 47–64.

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

[5] 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.

[6] G. Pandey, J. McBride, S. Savarese, and R. Eustice, "Extrinsic calibration of a 3d laser scanner and an omnidirectional camera," *IFAC Proceedings Volumes*, vol. 43, no. 16, pp. 336–341, 2010.

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

[8] 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.

[9] M. He, H. Zhao, J. Cui, and H. Zha, "Calibration method for multiple 2d lidars system," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3034–3041.

[10] C. Gao and J. R. Spletzer, "On-line calibration of multiple lidars on a mobile vehicle platform," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 279–284.

[11] L. Heng, B. Li, and M. Pollefeys, "Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1793–1800.

[12] L. Heng, M. Bürki, G. H. Lee, P. Furgale, R. Siegwart, and M. Pollefeys, "Infrastructure-based calibration of a multi-camera rig," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 4912–4919.

[13] J. Levinson and S. Thrun, "Automatic calibration of cameras and lasers in arbitrary scenes," in *International Symposium on Experimental Robotics*, 2012, pp. 1–6.

[14] Z. Taylor, J. Nieto, and D. Johnson, "Multi-modal sensor calibration using a gradient orientation measure," *Journal of Field Robotics*, vol. 32, no. 5, pp. 675–695, 2015.

[15] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, "Automatic targetless extrinsic calibration of a 3d lidar and camera by maximizing mutual information." in *AAAI*, 2012.

[16] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 3400–3407.

[17] A. Herout, M. Zachariáš, M. Dubska, and J. Havel, "Fractal marker fields: No more scale limitations for fiduciary markers," in *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 285–286.

[18] 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.

[19] Y. Wu and Z. Hu, "Pnp problem revisited," *Journal of Mathematical Imaging and Vision*, vol. 24, no. 1, pp. 131–141, 2006.

[20] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and vision computing*, vol. 10, no. 3, pp. 145–155, 1992.

[21] W. Maddern, A. Harrison, and P. Newman, "Lost in translation (and rotation): Rapid extrinsic calibration for 2d and 3d lidars," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3096–3102.