IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 8, 2020, accepted October 2, 2020, date of publication October 7, 2020, date of current version October 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3029267

Extrinsic Calibration of a Camera and a 2D LiDAR Using a Dummy Camera With IR Cut Filter Removed

JAE-YEUL KIM[1] **AND JONG-EUN HA**[2]

[1]Graduate School of Automotive Engineering, Seoul National University of Science and Technology, Seoul 01841, South Korea
[2]Department of Mechanics and Automotive Engineering, Seoul National University of Science and Technology, Seoul 01841, South Korea

Corresponding author: Jong-Eun Ha (jeha@seoultech.ac.kr)

This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2017R1D1A1B03028400, and in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIT) under Grant 2020R1A2C1013335.

ABSTRACT Extrinsic calibration of a camera and a LiDAR is necessary to fuse information from each sensor. The real trajectory of the LiDAR is not visible on an image, therefore the accuracy of the extrinsic calibration is usually checked by evaluating residuals of constraints. In this paper, we present an improved extrinsic calibration algorithm of a camera and a 2D LiDAR using an additional dummy camera removing IR cut filter, which make it possible to observe the real trajectory of LiDAR. Some previous algorithms used the real trajectory of LiDAR for the extrinsic calibration. However, they used IR filter directly on the calibrating camera by adjusting exposure time, which can affect the result of the extrinsic calibration. We use an initial solution using the Hu algorithm which makes extrinsic calibration possible by using just one shot of data. The Hu algorithm gives a sensitive result according to pose variation between a system consisted of a camera and a LiDAR and a calibration structure, which is verified using the real trajectory of LiDAR. We cope with this problem by refining the initial solution through nonlinear minimization in a 3D space using the real trajectory of LiDAR. Experimental results show that the proposed algorithm gives an improved solution.

INDEX TERMS Extrinsic calibration, sensor fusion, camera, LiDAR, IR cut filter.

I. INTRODUCTION

Extrinsic calibration between a camera and a LiDAR is a prerequisite step to fuse information from each sensor. Through the extrinsic calibration, it is possible to represent the information of two sensors under the same coordinate system. In general, the sensor fusion between a camera and a LiDAR is done by projecting LiDAR data onto the corresponding image point, through it, it is possible to combine 3D information by the LiDAR and intensity information by the camera. The goal of sensor fusion is to extract a more improved result than the case when we use each sensor individually. For accurate sensor fusion, correspondences to the same points must be established, which is possible using the information from the extrinsic calibration. The trajectory of LiDAR data is not visible on an image because LiDAR use a light source in an infrared band. For this reason, the evaluation of extrinsic calibration of the camera and the LiDAR is usually done by checking how well the constraints are satisfied, and the amount of the residual of constraints is usually investigated. This may invoke a problem that a solution with a small residual does not always give a correct extrinsic calibration.

Wasielewski and Strauss [1] proposed an extrinsic calibration algorithm for multiple cameras and LiDARs using a V-shaped calibration structure. After fitting the two straight lines using the data of the LiDAR irradiated to the V-shaped structure, they find the intersection of the two straight lines. They use the constraint that the intersection point should exist in the center straight line of the V-shaped structure on the image. The linear solution for the extrinsic calibration is not noted in the paper, and the above constraint is used for nonlinear minimization. In Zhang and Press [2], a method for obtaining a linear solution of extrinsic calibration is proposed using the constraint that the points of the LiDAR data are present in the calibration structure of plane. The improved solution is obtained by nonlinear minimization which uses the distance between the point and the plane as a cost function.

The associate editor coordinating the review of this manuscript and approving it for publication was Yakoub Bazi.

Li *et al.* [3] proposed extrinsic calibration algorithm which uses a triangular structure at right angles to each other is proposed. An initial solution is specified manually and a solution is obtained by a nonlinear minimization using constraints between points and straight lines. Kassir and Peynot [4] suggests an automated method using the algorithm of Zhang and Press [2]. They automated the entire process of related LiDAR data processing and camera calibration. Bok *et al.* [5] proposed a linear solution for the extrinsic calibration of a camera and a LiDAR using point-line constraints. In order to improve the preceding solution, a nonlinear minimization method using a distance between a point and a straight line is used. In Kwak *et al.* [6], a V-shaped structure is used, and constraints between points and straight lines are used. A distance between a point and a straight line was used as variable weights to remove outliers, which improved the performance of extrinsic calibration.

In Yang *et al.* [7], an infrared filter is used to find a LiDAR trajectory on an image, and an extrinsic calibration algorithm of a camera and a LiDAR using a Perspective-n-Point (PnP) method [8] is proposed. They are similar to the proposed algorithm in that both use the real trajectory of LiDAR on an image. They attach IR filter to a camera under calibration and obtain the trajectory of LiDAR by adjusting exposure time, while the proposed algorithm uses additional camera removing IR cut filter. Gomez-Ojeda *et al.* [9] suggested a method using triangular forms perpendicular to each other that can be easily found in indoor environments. It uses constraints derived from line-plane and point-plane. Vasconcelos *et al.* [10] proposed a method using a correspondence relationship between a plane and a straight line having a maximum of eight solutions. A linear solution based on the PnP-based method is presented.

In Zhou [11], an extrinsic calibration method using three planes and three straight lines corresponding to it was proposed. They proposed a method for solving algebraic solutions from polynomials by constraints, and first estimate the rotational component and then the translational component. Hu *et al.* [12] proposed an algorithm that adopts a PnP method using three mutually perpendicular planes, which make extrinsic calibration possible using one shot. Li *et al.* [13] proposed an algorithm that uses a V-shape board with controllable angle between two planes, and they also proposed an analytical solution for the extrinsic calibration with a single observation. Fan *et al.* [14] proposed an algorithm for the extrinsic calibration of a camera and a LiDAR using a photogrammetric control field, which requires even distribution of control points in a 3D space. They decouple extrinsic calibration by independently calibrating the camera and the LiDAR with respect to the photometric control field. Briales and Gonzalez-Jimenez [15] used corners of an orthogonal trihedron which is easily found in many buildings for the extrinsic calibration of a camera and a 2D LiDAR. Zhu *et al.* [16] proposed an extrinsic calibration algorithm among multiple LiDARs, and they used corners of three perpendicular planes for the extrinsic calibration. Tian *et al.* [17] used checkerboard trihedron for the extrinsic calibration of a camera and a 2D LiDAR, and they proposed an algorithm consisted of three steps.

Recently, with the advance of autonomous navigation, 3D LiDAR is used as an essential sensor, and it is also used together with cameras. 3D LiDAR covers more area and gives more point cloud than 2D LiDAR, but it is more expensive than 2D LiDAR. In a point of extrinsic calibration, extrinsic calibration between a 3D LiDAR and a camera is rather easy than that of 2D LiDAR and camera, because they provides more correspondences between the LiDAR data and image [18]–[20]. Sui and Wang [20] uses a specially designed calibration structure consists of a 3D marker having four circles and a chessboard. Pentek *et al.* [21] proposed an automatic and targetless calibration algorithm of LiDAR-GNSS/INS-camera system. They compute calibration parameters by optimizing correspondences by a camera and a LiDAR.

In the case of the existing studies, calibration structures made of planes or combinations of planes are used, and extrinsic calibration of a camera and a LiDAR is performed using geometric constraints derived from these calibration structures. Since the actual position where the LiDAR is projected on the image is unknown, the evaluation of the extrinsic calibration is done based on the error due to a given constraint or qualitatively by checking whether or not the depth discontinuities match.

In this paper, we propose an algorithm to improve the performance of the Hu algorithm [12] by using an additional dummy camera, which removed the infrared cut filter to observe the actual trajectory of the LiDAR on an image. First, we show that the Hu algorithm [12] gives a sensitive calibration result according to the variation of pose between a system consisted of a camera and a LiDAR and a calibration structure. It is shown in quantitative manner by using actual trajectory of a LiDAR. Second, we present a method to improve the performance of the Hu algorithm [12], which does a nonlinear minimization in a 3D space using the actual trajectory of a LiDAR by the dummy camera.

II. PROPOSED METHOD

Figure 1 shows a system consisted of two cameras and one LiDAR used in the experiment. In this paper, the camera with the infrared cut filter removed is called a dummy camera, and the camera needs to be calibrated is called a normal camera. A dummy camera with an infrared cut filter removed is used to verify the actual trajectory of LiDAR on an image.

The following equations hold among three coordinate systems related to a normal camera, a dummy camera, and a LiDAR.

$$\mathbf{P}_A^w = \mathbf{R}_{dc}^w \mathbf{P}_A^{dc} + \mathbf{T}_{dc}^w \tag{1}$$

$$\mathbf{P}_A^w = \mathbf{R}_{nc}^w \mathbf{P}_A^{nc} + \mathbf{T}_{nc}^w \tag{2}$$

$$\mathbf{P}_A^w = \mathbf{R}_{\mathbf{l}}^w \mathbf{P}_A^l + \mathbf{T}_l^w \tag{3}$$

R represents a 3 × 3 rotation matrix and **T** represents a 3 × 1 translation vector. The superscript or subscript *dc*, *nc*, *l*, and *w* represent the dummy camera, the normal camera, the LiDAR and the world coordinate system, respectively. $\mathbf{P}_A^w$, $\mathbf{P}_A^{dc}$, $\mathbf{P}_A^{gc}$, and $\mathbf{P}_A^l$ represent the coordinates of a point *A* in the world coordinate system, the dummy camera coordinate system, the normal camera coordinate system, and the LiDAR coordinate system, respectively. Through extrinsic calibration, we obtain all **R** and **T**.

The proposed method first performs extrinsic calibration between a dummy camera and a LiDAR, and finally performs extrinsic calibration between a normal camera and a LiDAR. The pose between a dummy camera and a LiDAR can be converted to the pose between a normal camera and a LiDAR using the pose between a normal camera and a dummy camera.

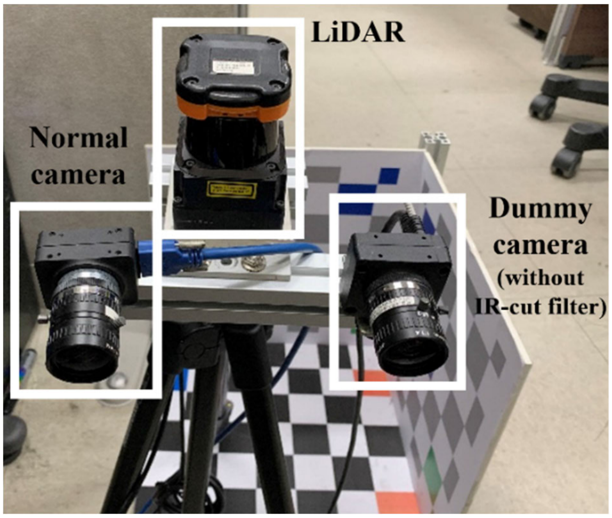

FIGURE 1. System configuration: normal camera, LiDAR, and dummy camera.

A. OVERVIEW OF HU ALGORITHM [12]

The proposed algorithm obtains the initial solution for extrinsic calibration of a dummy camera and a LiDAR using the Hu algorithm [12], and the outline of the algorithm is as follows. 3D-3D correspondence based on Perspective 3 Points (P3P) [22] is used for the extrinsic calibration between a LiDAR coordinate system and a world coordinate system on a calibration structure, as shown in Figure 2.

In Figure 2, there are three triangles $\Delta\mathbf{P}_0\mathbf{P}_1\mathbf{P}_2$, $\Delta\mathbf{P}_0\mathbf{P}_2\mathbf{P}_3$, and $\Delta\mathbf{P}_0\mathbf{P}_1\mathbf{P}_3$. Applying cosine's second law to the three triangles gives the following equation.

$$d_{12}^2 = d_{01}^2 + d_{02}^2 - 2d_{01}d_{02}cos\theta_{12} \quad (4)$$

$$d_{23}^2 = d_{02}^2 + d_{03}^2 - 2d_{02}d_{03}cos\theta_{23} \quad (5)$$

$$d_{13}^2 = d_{01}^2 + d_{03}^2 - 2d_{01}d_{03}cos\theta_{13} \quad (6)$$

d_{ij} denotes the distance between points $\mathbf{P}_i$ and $\mathbf{P}_j$, and θ_{ij} denotes the angle between a vector passing through points $\mathbf{P}_0$ and $\mathbf{P}_i$ and a vector passing through points $\mathbf{P}_0$ and $\mathbf{P}_j$. The coordinates of the three points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ in a LiDAR coordinate system can be obtained through LiDAR data processing, which involves finding depth discontinuities. When three points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ are known, it is possible to obtain a distance between two points, e.g., d_{12}, d_{23}, and d_{13}. When we use a calibration structure that is orthogonal to each other as shown in Figure 3, θ_{12}, θ_{23}, and θ_{13} is 90°. We obtain following equations after applying them Eq. (4) to Eq. (6).

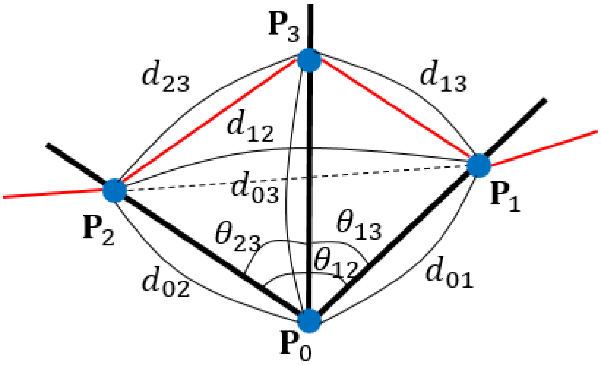

FIGURE 2. Extrinsic calibration of a camera and a LiDAR using PnP (Perspective-n-Point).

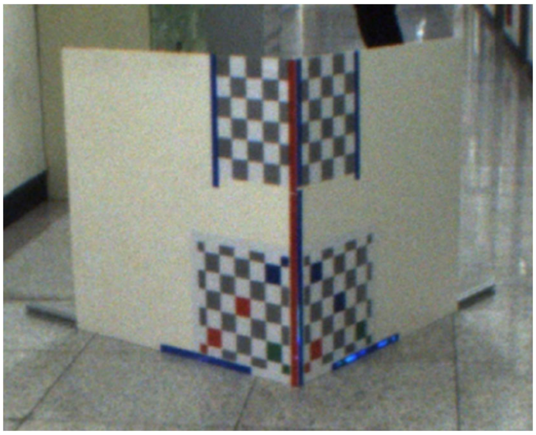

FIGURE 3. A calibration structure used for extrinsic calibration.

$$d_{12}^2 = d_{01}^2 + d_{02}^2 \quad (7)$$

$$d_{23}^2 = d_{02}^2 + d_{03}^2 \quad (8)$$

$$d_{13}^2 = d_{01}^2 + d_{03}^2 \quad (9)$$

Solving the above system of equations for d_{01}, d_{02}, and d_{03}, we have following equations.

$$d_{01} = \sqrt{\frac{d_{12}^2 + d_{13}^2 - d_{23}^2}{2}} \quad (10)$$

$$d_{02} = \sqrt{\frac{d_{12}^2 + d_{23}^2 - d_{13}^2}{2}} \quad (11)$$

$$d_{03} = \sqrt{\frac{d_{13}^2 + d_{23}^2 - d_{12}^2}{2}} \quad (12)$$

Knowing d_{01}, d_{02}, and d_{03} means that we have three points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ in a world coordinate system. Therefore, extrinsic calibration between a world coordinate system and a

LiDAR coordinate system can be computed. Throughout this process, we can represent the same point under two different coordinate systems. By using these 3D-3D correspondence relationships, it is possible to obtain a conversion relationship between a world coordinate system and a LiDAR coordinate system. Various methods [22] have been proposed for obtaining the relationship between two coordinate systems using 3D-3D correspondence. Extrinsic calibration between a world coordinate system and a camera coordinate system can be solved using camera calibration algorithms [23]–[25]. Finally, it is possible to obtain extrinsic calibration between a camera coordinate system and a LiDAR coordinate system by using two results of extrinsic calibration. One is obtained between a world coordinate system and a LiDAR coordinate system, and the other is obtained between a world coordinate system and a camera coordinate system.

B. EXTRINSIC CALIBRATION USING REAL LiDAR TRAJECTORY BY A DUMMY CAMERA

In this paper, the real trajectory of the LiDAR is extracted using a dummy camera with the infrared cut filter removed, and this is used for extrinsic calibration of a camera and a LiDAR. For extrinsic calibration, a structure composed of two mutually perpendicular planes was used, as shown in Figure 3. The upper chessboard pattern was used for the validation of extrinsic calibration, and the lower chessboard pattern was used for extrinsic calibration of a camera and a LiDAR. If the lower part of the structure in Figure 3 is used, the geometric constraint in Figure 2 can be used for the extrinsic calibration of a dummy camera and a LiDAR.

The reflective tape on the calibration structure was attached to facilitate the detection of the actual trajectory of the LiDAR with ease. If we remove IR cut filter from a camera, it is possible to observe the real trajectory of a LiDAR on an image when a distance from a dummy camera to a calibration structure is near or environmental lighting is weak. When the distance is large or the surrounding lighting is strong, it is difficult to extract the real trajectory of a LiDAR on an image. A reflective tape was used to mitigate this problem. When a reflective tape is attached, more strong reflected light is generated at the location where a LiDAR passes, making it easier to detect a trajectory on an image. Therefore, it is possible to detect the actual trajectory of a LiDAR at a relatively long distance.

As shown in Figure 4, we can observe the actual trajectory of a LiDAR on an image by the camera with the infrared cut filter removed. It was possible to observe the trajectory of a LiDAR on image at near position under normal lighting in indoor. At far position up to 5m, it is possible to observe the trajectory of a LiDAR on image by turning off lighting in indoor. In Figure 4, the actual trajectory of a LiDAR correspond to the white line on the image. After extrinsic calibration between a world coordinate system and a dummy camera coordinate system, we can convert the trajectory of a LiDAR on an image into a world coordinate system.

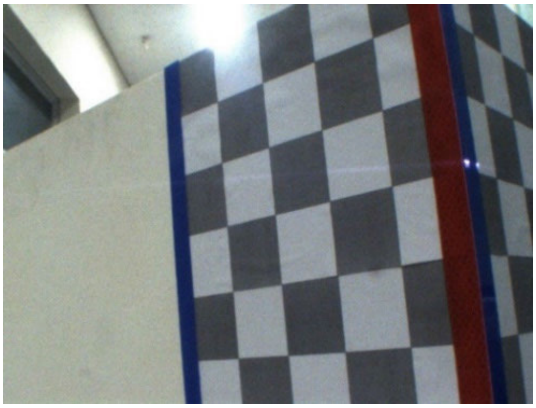

FIGURE 4. The actual trajectory of LiDAR on an image by a dummy camera.

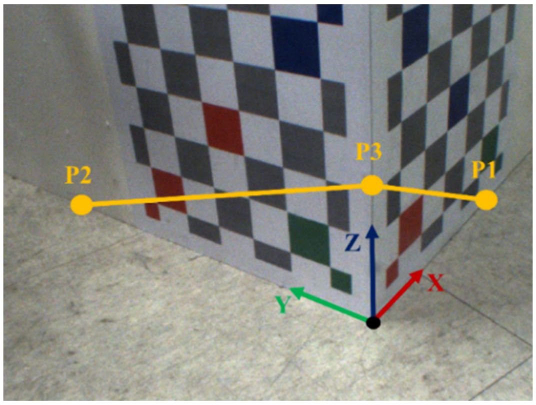

FIGURE 5. Three control points used in extrinsic calibration.

First, the extrinsic calibration of a dummy camera and a LiDAR is performed using the Hu algorithm [12], which is used as an initial solution. In the case of Hu algorithm [12], extrinsic calibration can be performed using only three control points. As shown in Figure 5, in the case of the calibration structure used in this paper, it is possible to acquire three control points with one shot. Extrinsic calibration between a world coordinate system and a camera coordinate system is performed using a chessboard pattern attached to a calibration structure. Therefore, extrinsic calibration between a camera and a LiDAR is possible through one data acquisition. This is regarded as an initial solution, and a final solution is obtained through nonlinear minimization using an actual trajectory on an image by a dummy camera.

We manually select three control points, which correspond to $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$, from an actual LiDAR trajectory on an image by a dummy camera. LiDAR data between two control points is used for a nonlinear minimization. In the proposed method, two nonlinear minimization methods were compared. The first method does a nonlinear minimization by converting LiDAR data in an image coordinate system into a world coordinate system. The second method does a nonlinear minimization under an image coordinate system by projecting LiDAR data onto an image. The second method is similar to an algorithm, which is widely used in camera calibration. A distance from a point to a line is used as cost in both methods. Experimental results show that performing a nonlinear minimization in a 3D space provides good results.

III. EXPERIMENTAL RESULTS

Experiments are done using a system as shown in Figure 1. We used FLIR's Chameleon3 USB3 camera and USB2 camera as a normal camera and a dummy camera, respectively. Hokuyo UTM-30LX is used as a 2D LiDAR.

We performed extrinsic calibration multiple times for the same system configuration as shown in Figure 1. A total of nine extrinsic calibration were performed by varying a distance and a pose between a system and a calibration structure. Figure 6 shows images in each case used for extrinsic calibration.

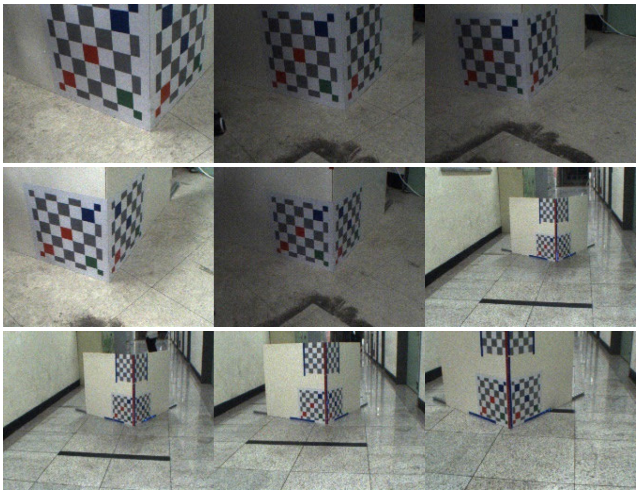

FIGURE 6. Images of calibration structure for all 9 cases.

The evaluation of the result of extrinsic calibration was done using the actual trajectory of a LiDAR. Using a dummy camera, we can observe the actual trajectory of a LiDAR on an image. A crossing point where LiDAR data meets at the intersection of two planes is found, and it is used for the evaluation of extrinsic calibration. In Figure 7, the orange circle shows a validation data obtained from an actual LiDAR trajectory on an image by a dummy camera. The axis of X, Y, and Z of a world coordinate system is set as shown in Figure 5. We convert a control point in an image coordinate system into a point, $\mathbf{P}_t^w$, in a world coordinate system, and we regard it as a ground truth for evaluation.

A corresponding LiDAR point to a point $\mathbf{P}_t^w$ is denoted as $\mathbf{P}_e^l$. It is a LiDAR point that meet at the intersection of two planes. After we obtain a point $\mathbf{P}_e^l$, it is used to evaluate the result of extrinsic calibration. A point $\mathbf{P}_e^l$ in a LiDAR coordinate system is converted into a point $\mathbf{P}_e^w$ in a world coordinate system using computed value of extrinsic calibration. A distance between two points $\mathbf{P}_t^w$ and $\mathbf{P}_e^w$in a 3D space is considered as an error. In the case of a calibration structure, since two planes are vertically arranged, the LiDAR data passing through them can be approximated by two straight lines. In this paper, in order to obtain accurate control point coordinates, the coordinates of the raw LiDAR data are not used. The intersection of the two lines is used as control point after estimating the two lines. Figure 8 shows the process of extracting a LiDAR control point using two fitted lines, which is used for verification. The upper part of a calibration structure in Figure 3 was used for the evaluation of extrinsic calibration. At a fixed distance of 0.5m, 1.0m, 1.5m, 2m, 3m, 4m, and 5m, we obtain three control points from the left, center, and right position. A total of 21 control points were used for the evaluation of extrinsic calibration. From Table 1 to Table 4 validation error means the error computed using these 21 control points in a 3D space.

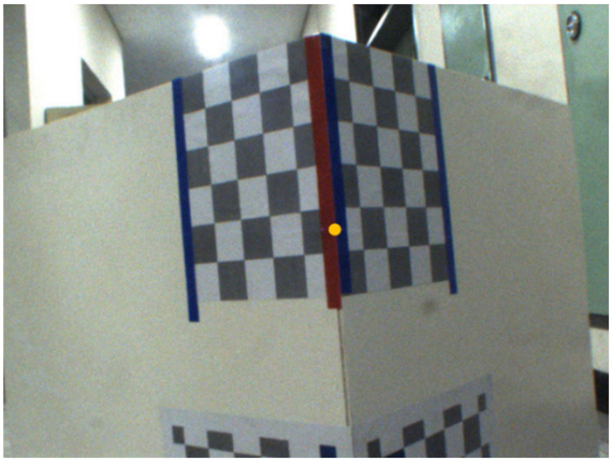

FIGURE 7. A control point on an image by a dummy camera which is used in the evaluation of extrinsic calibration.

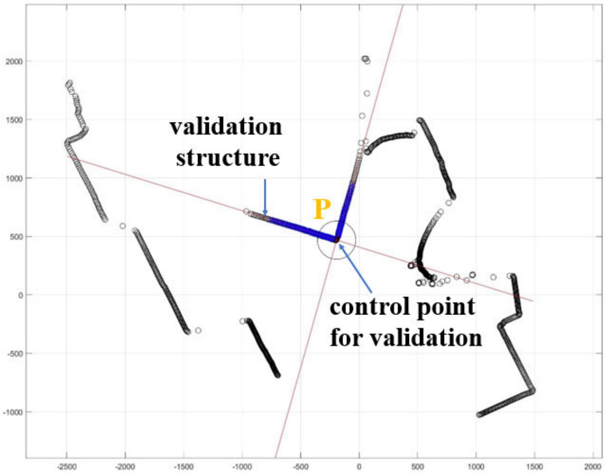

FIGURE 8. Computation of a LiDAR control point for validation using line fitting (black circle: original LiDAR data, blue circle: LiDAR data used in line fitting, red line: fitted line, yellow circle: control point found by crossing two lines).

Table 1 shows the results of extrinsic calibration between a dummy camera and a LiDAR obtained by the Hu algorithm [12]. In three trials among total nine cases, extrinsic calibration was not possible as shown in Table 1. Hu algorithm uses geometric constraints inherent on a calibration structure. In the case of the calibration structure in Figure 3, since each surface is made to be perpendicular to each other and a floor is flat, a calibration structure itself has no problem.

TABLE 1. The result of extrinsic calibration by Hu algorithm [12] using raw LiDAR data.

	Computed **R** $(\theta_x, \theta_y, \theta_z)\ [deg]$	Computed **T** $(T_x, T_y, T_z)\ [mm]$	$\|\|$**T**$\|\|$ [mm]	Validation Error [mm]
T1	(-89.89, -0.37, 0.38)	(81.84, 10.53, -71.76)	109.4	99.37
T2	(-92.38, 0.30, -0.30)	(97.62, 72.20, -66.28)	138.3	154.16
T3	(-108.72, 1.95, -0.92)	(115.70, 448.91, 10.98)	463.7	535.54
T4	(-88.93, -0.40, 0.49)	(79.17, 3.17, -67.57)	102.6	70.22
T5	(-95.15, -0.47, 0.02)	(93.44, 183.18, -58.84)	213.9	202.99
T6	x	x	x	x
T7	x	x	x	x
T8	x	x	x	x
T9	(-97.73, 0.13, -1.20)	(143.61, 368.15, -18.25)	395.6	256.29

TABLE 2. The result of extrinsic calibration by Hu algorithm [12] using line fitting.

	Computed **R** $(\theta_x, \theta_y, \theta_z)\ [deg]$	Computed **T** $(T_x, T_y, T_z)\ [mm]$	$\|\|$**T**$\|\|$ [mm]	Validation Error [mm]
T1	(-87.04, -0.58, 0.62)	(74.99, -37.92, -74.96)	112.6	39.54
T2	(-83.76, -0.89, 0.48)	(75.98, -106.07, -70.01)	148.1	76.86
T3	(-85.23, -0.59, 1.13)	(62.64, -80.43, -73.43)	125.6	50.37
T4	(-84.86, -0.47, 0.84)	(68.79, -87.08, -72.61)	132.6	54.70
T5	(-87.86, -0.44, 0.94)	(65.24, -14.86, -74.96)	100.5	51.17
T6	(-81.41, -1.65, 2.49)	(-65.65, -400.72, 4.33)	406.1	189.58
T7	(-90.40, -0.04, -0.30)	(119.49, 179.31, -9.67)	215.7	116.65
T8	(-90.10, -0.20, 2.05)	(-11.77, 134.15, -49.14)	143.4	103.45
T9	(-78.83, -0.05, 0.44)	(77.94, -352.33, -20.22)	361.4	174.26

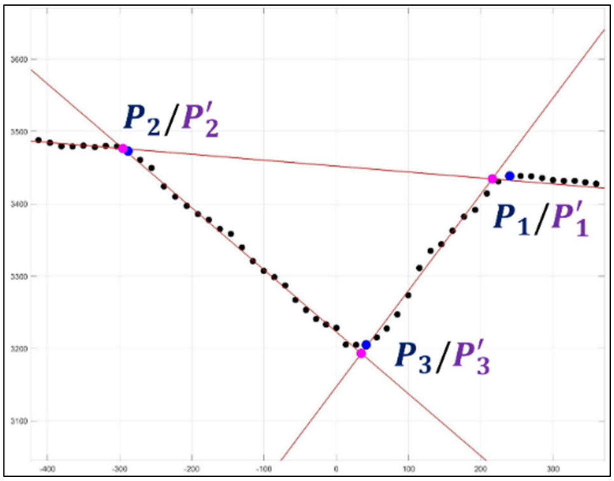

FIGURE 9. Computation of control points using line fitting (Black dots: raw LiDAR data, red lines: fitted lines, blue circles: control points selected using raw LiDAR data, red circles: control points by fitted lines).

TABLE 3. the Result of extrinsic calibration by nonlinear minimization in the world coordinate system.

	Computed **R** $(\theta_x, \theta_y, \theta_z)\ [deg]$	Computed **T** $(T_x, T_y, T_z)\ [mm]$	$\|\|$**T**$\|\|$ [mm]	Val. Err. [mm]
T1	(-87.74, 0.68, 0.92)	(71.20, -33.49, -77.85)	110.7	55.54
T2	(-86.56, -0.02, 0.21)	(82.66, -47.98, -70.32)	118.6	33.03
T3	(-86.82, -0.55, 0.70)	(72.91, -39.62, -71.39)	109.5	34.96
T4	(-87.37, -0.35, 0.21)	(82.80, -31.37, -69.27)	112.4	40.60
T5	(-86.92, 0.39, 1.44)	(52.53, -40.04, -74.72)	99.7	44.73
T6	(-85.72, -0.93, 2.04)	(-35.17, -96.32, -20.46)	104.6	68.73
T7	(-85.91, -0.70, 0.08)	(95.62, -76.96, 0.02)	122.7	45.16
T8	(-86.18, -1.01, 1.83)	(-0.01, -63.63, -42.49)	76.5	46.85
T9	(-84.79, 0.16, -0.31)	(108.88, -108.45, -39.38)	158.6	47.49

Hu algorithm [12] uses three LiDAR control points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ present on the corners of the calibration structure, as shown in Figure 5. Due to the LiDAR measurement errors amount to ±10 mm, it caused a problem of imaginary number in Eq. (10) to Eq. (12).

In order to solve this problem caused by the measurement error of a LiDAR, more accurate LiDAR control points were computed using line fitting. After the selection of three points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$, straight lines were fitted using LiDAR data between two points. The new LiDAR control points $\mathbf{P}'_1$, $\mathbf{P}'_2$, and $\mathbf{P}'_3$ were obtained by finding crossing points of fitted lines, and these were used for extrinsic calibration. Figure 9 shows the comparison result between control points obtained manually using a raw LiDAR data and control points obtained using line fitting. Table 2 shows the result of extrinsic calibration using new control points. In the case of using LiDAR control points obtained through line fitting, extrinsic calibration was possible for all data sets, as shown in Table 2. In addition, it gives the lower validation error than that of using control points selected from a raw LiDAR data.

After an initial solution is obtained using the Hu algorithm [12], a nonlinear minimization is performed using the actual trajectory of a LiDAR on an image by a dummy camera. Three control points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$in Figure 5 are manually extracted from an image by a dummy camera with a IR cut filter removed. A fitted line using points between $\mathbf{P}_1$ and $\mathbf{P}_3$ is denoted as l_1, and a fitted line using points between $\mathbf{P}_2$ and $\mathbf{P}_3$ is denoted as l_2. The sum of a distance between a line l_1 and a point along $\mathbf{P}_1$ and $\mathbf{P}_3$ and a distance between a line l_2 and a point along $\mathbf{P}_2$ and $\mathbf{P}_3$ is used as cost in a nonlinear minimization. Nonlinear minimization was done using MATLAB fmincon function.

A nonlinear minimization using a real LiDAR trajectory by a dummy camera can be proceeded in two ways. The first method performs a nonlinear minimization in a 3D space. The second method performs a nonlinear minimization in a 2D space of image. As shown in Figure 5, in the case of a calibration structure used in this paper, each face is vertical and two planes represent the XZ and YZ planes, respectively. Therefore, it is simple to convert detected control points on an image into points in a world coordinate system. In addition, a LiDAR data can also be expressed as coordinates in a world coordinate system using **R** and **T** between a LiDAR coordinate system and a world coordinate system. It is possible to minimize cost in a world coordinate system using a dummy camera. Table 3 shows result by a nonlinear minimization in a world coordinate system, and we can notice that error is reduced in all cases except T1.

TABLE 4. The result of extrinsic calibration by nonlinear minimization in the pixel coordinate system.

	Computed **R** $(\theta_x, \theta_y, \theta_z)$ $[deg]$	Computed **T** (T_x, T_y, T_z) $[mm]$	Validation Error [mm]
T1	(-88.61, 1.65, 1.68)	(64.6, -23.1, -48.5)	83.66
T2	(-90.48, -5.50, 65.31)	(-649.5, -14.0, 5057.4)	4455.9
T3	(-86.67, -0.25, 0.97)	(64.0, -43.2, -49.5)	26.9
T4	(-90.57, 1.02, -35.24)	(-3559.8, 1355.6, 8420.2)	8387.9
T5	(-86.88, 0.34, 0.93)	(63.9, -40.4, -49.6)	28.3
T6	(-87.29, -1.81, 55.44)	(-2254.6, 677.0, 15603.6)	14839.0
T7	(-87.26, -1.62, 48.57)	(-1903.2, 404.5, 9989.2)	9440.8
T8	(-86.68, -0.33, 0.35)	(64.0, -39.8, -49.8)	27.2
T9	(-88.19, 2.12, -41.10)	(1106.2, 26.2, 2016.4)	1863.7

Also, it is possible to do a nonlinear minimization in a 2D pixel coordinate system, not in a 3D world coordinate system. In this case, LiDAR control points are projected onto an image using **R** and **T** between a LiDAR coordinate system and a camera coordinate system. The lines l_1 and l_2 are defined in a pixel coordinate system. Table 4 shows results by a nonlinear minimization on a pixel coordinate system. 5 cases among total 9 cases show completely wrong results. The reason for showing this result is that cost is defined as a distance between a LiDAR data and a line on a pixel coordinate system. For example, if we denote the intersection point of two lines l_1 and l_2 as (x, y), all LiDAR data that are projected onto (x, y) would have zero cost. For all the wrong cases in Table 4, the estimated translation vector has large magnitude, and this confirm our reasoning about failure cases. This problem does not happen in a nonlinear minimization on a 3D space.

In Table 3 and Table 4, validation error is computed using the real trajectory of LiDAR. Therefore, we can consider it as the correct evaluation of extrinsic calibration. In Table 4, all nine cases converged, and eventually showing small residual after a nonlinear minimization. Using the real trajectory of a LiDAR like the proposed algorithm makes it possible to check whether the result of extrinsic calibration is good or bad in quantitative way.

Table 5 shows the result of a quantitative error evaluation in the validation set of each extrinsic calibration result by the Hu [12] and the proposed algorithm. In Table 5, the distance between a camera and a calibration structure is also indicated for each case. In Table 5, Hu algorithm gives a better result than proposed algorithm only in T1 case. In other cases of T2 and T3, we could have improved result compared to T1 result by Hu algorithm. Also, error value for each location is shown. A 2D LiDAR used in this paper is the Hokuyo UTM-30LX, which has an error of ±30mm in the range of 0.1m to 10m. Considering these errors, we can conclude that the proposed method provides proper extrinsic calibration results in all cases.

In the case of T6 by the Hu algorithm, it shows a large projection error of 189.6mm on average, but it shows a small error of 30∼50mm at a distance of 3∼4m. In the case of T6 ∼ T9 by the Hu algorithm [12], extrinsic calibration is done using data acquired at a relatively large distance, they showed a large error of more than 100mm at 0.5m, 1.0m, and 1.5m positions. In the case of the Hu algorithm [12], although a small error is shown at a distance close to a position where

TABLE 5. Comparison of error per position of validation data by the Hu [12] and the proposed algorithm.

Case/Distance to calibration structure [m]		Error per position (0.5m, 1.0m, 1.5m, 2m, 3m, 4m, 5m) [mm]	Mean [mm]
T1/0.90	Hu [12]	(**10**, 15, 21, 34, 62, 58, 76)	39.5
	Proposed	(**9**, 18, 29, 45, 80, 91, 118)	55.5
T2/1.20	Hu [12]	(36, **20**, 24, 48, 99, 130, 181)	76.9
	Proposed	(12, **12**, 17, 30, 55, 45, 60)	33.0
T3/1.34	Hu [12]	(24, **19**, 21, 36, 70, 77, 106)	50.4
	Proposed	(13, **12**, 17, 29, 56, 50, 66)	35.0
T4/1.27	Hu [12]	(26, **19**, 21, 38, 74, 86, 119)	54.7
	Proposed	(17, **10**, 17, 31, 61, 63, 85)	40.6
T5/1.52	Hu [12]	(25, **17**, 19, 34, 70, 82, 112)	51.2
	Proposed	(20, **15**, 19, 33, 65, 69, 93)	68.7
T6/3.79	Hu [12]	(344, 295, 246, 201, 110, **37**, 95)	189.6
	Proposed	(124, 100, 83, 66, 37, **23**, 48)	68.7
T7/3.37	Hu [12]	(212, 173, 136, 102, **30**, 50, 114)	116.7
	Proposed	(86, 64, 56, 42, **17**, 29, 23)	45.2
T8/3.00	Hu [12]	(174, 136, 100, 69, **25**, 74, 146)	103.5
	Proposed	(78, 56, 43, 31, **28**, 32, 60)	46.9
T9/2.14	Hu [12]	(234, 168, 98, **40**, 104, 219, 356)	174.3
	Proposed	(65, 42, 24, **12**, 40, 58, 92)	47.5
Mean/std	Hu [12]	81.14/43.15	
	Proposed	46.20/10.79	

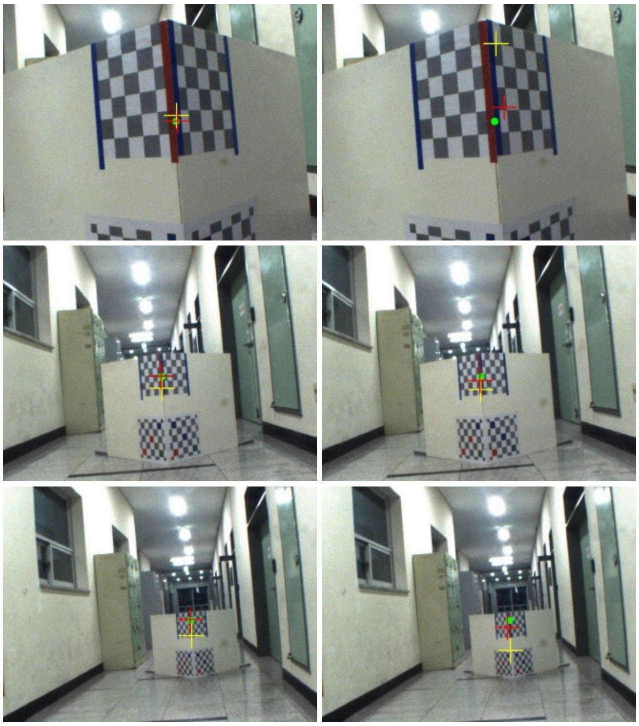
(a) (b)

FIGURE 10. Comparison of projection error on a dummy camera (green circle: ground truth, yellow cross: by the Hu [12] algorithm, red cross: by the proposed algorithm, first row: 1m, second row: 3m, third row: 5m) (a) by result of T2 in Table 5 (b) by result of T9 in Table 5.

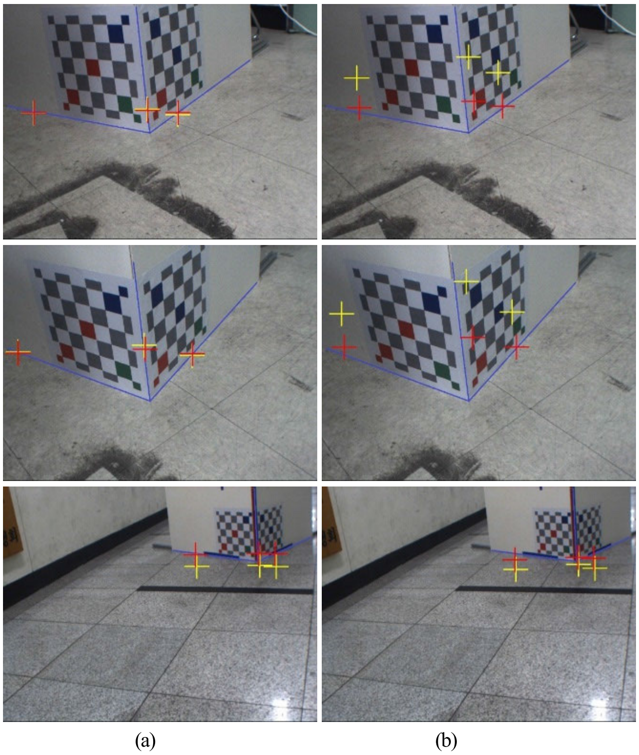

FIGURE 11. Comparison of projection error on a normal camera (blue line: calibration structure corner, yellow cross: by the Hu [12] algorithm, red cross: by the proposed algorithm, first row: calibration structure used in T3, second row: calibration structure used in T4, third row: calibration structure used in T8) (a) by result of T2 in Table 5 (b) by result of T9 in Table 5.

extrinsic calibration is done, we can notice that an error is amplified according to the change of distance. We can notice that the proposed method can solve this problem. In average, the Hu algorithm [12] and the proposed algorithm gives error statistics of 81.14 ± 43.15 mm and 46.20 ± 10.79 mm. The proposed algorithm gives a solution having a small error and variation compared to the Hu algorithm. The Hu algorithm [12] gives a sensitive result according to pose and distance between a system and a calibration structure, while the proposed algorithm gives a consistent result.

Experiments are done under the same configuration as shown in Figure 1 without changing relative pose among a normal camera, a dummy camera, and a 2D LiDAR, therefore the ground truth value of **R** and **T** is unique. It is difficult to find the ground truth value of **R** and **T** of extrinsic calibration. The ground value of **R** is rather difficult than that of **T**. The norm of **T** is a Euclidean distance between the origin of a camera and a LiDAR, respectively. We can roughly measure this distance and can used it as reference value for checking. From Table 1 to Table 3, $\|\mathbf{T}\|$ is also displayed for comparison. The proposed algorithm gives a $\|\mathbf{T}\|$ value with less variation than Hu algorithm.

Qualitative comparison was performed by projecting the control points used for extrinsic calibration onto an image using the result of the extrinsic calibration. Figure 10 shows the result of projecting control points used for verification onto an image by a dummy camera using the result of extrinsic calibration by the Hu and the proposed algorithm. The green circle represents the position of a control point manually extracted using an actual LiDAR trajectory. A yellow cross and a red cross represent a projected point of a control point using extrinsic calibration result by the Hu algorithm and the proposed algorithm, respectively. Figure 10(a) is the result by T2 case and Figure 10(b) is the result by T9 case in Table 2 and Table 3.

Figure 11 shows the result of projecting three control points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ used in a dummy camera onto a normal camera using a relative pose between a dummy camera and a normal camera. In Figure 10, a blue straight line represents the corner of a calibration structure. If extrinsic calibration result is correct, three control points should be projected at corner. In Figure 11, a yellow cross is result by the Hu algorithm [12] and a blue cross is result by the proposed algorithm. Figure 11(a) is the result by T2 case in Table 5, and Figure 11(b) is the result by T9 case in Table 5. From the first row to the third row in Figure 10, the result of projecting control points $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ onto a corresponding image of a normal camera used in T3, T4, and T8, respectively, is shown. From the result of Figure 10 and Figure 11, we can conclude that the proposed algorithm gives improved results compared to the Hu algorithm [12].

We think that a LiDAR modeling similar to camera modeling during camera calibration is required to improve the extrinsic calibration of a camera and a 2D LiDAR. Also, bundle adjustment which is widely used in 3D reconstruction also could be used in the extrinsic calibration of a camera and a 2D LiDAR to improve the accuracy.

IV. CONCLUSION

In this paper, we proposed an algorithm to improve the performance of the Hu algorithm [12], which has the advantage of being able to have an extrinsic calibration of a camera and a LiDAR using data with just one shot. Through quantitative evaluation using an actual LiDAR trajectory obtained from a dummy camera with IR cut filter removed, it was found that the Hu algorithm [12] provides sensitive results according to distance and pose between a system consisted of a camera and a LiDAR and a calibration structure. We show that this problem can be solved using the actual trajectory of a LiDAR on an image by a dummy camera. An initial solution was obtained using the Hu algorithm [12], and a method using nonlinear minimization using a real LiDAR trajectory was presented. A nonlinear minimization can be performed in both in a 2D pixel space and in a 3D space, and it has been shown that a nonlinear minimization in a 3D space gives a better solution.

REFERENCES

[1] S. Wasielewski and O. Strauss, "Calibration of a multi-sensor system laser rangefinder/camera," in *Proc. Intell. Vehicles Symp.*, Sep. 1995, pp. 472–477.

[2] Q. Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep./Oct. 2004, pp. 2301–2306.

[3] G. Li, Y. Liu, L. Dong, X. Cai, and D. Zhou, "An algorithm for extrinsic parameters calibration of a camera and a laser range finder using line features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2007, pp. 3854–3859.

[4] A. Kassir and T. Peynot, "Reliable automatic camera-laser calibration," in *Proc. Australas. Conf. Robot. Automat.*, 2010, pp. 1–10.

[5] Y. S. Bok, Y. K. Jeong, D. G. Choi, and I. S. Kweon, "Capturing village-level heritages with a hand-held camera-laser sensor," *Int. J. Comput. Vis.*, vol. 94, no. 1, pp. 36–53, 2011.

[6] K. Kwak, D. F. Huber, H. Badino, and T. Kanade, "Extrinsic calibration of a single line scanning lidar and a camera," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 3283–3289.

[7] H. Yang, X. Liu, and I. Patras, "A simple and effective extrinsic calibration method of a camera and a single line scanning LiDAR," in *Proc. Int. Conf. Pattern Recognit.*, Nov. 2012, pp. 1439–1442.

[8] L. Quan and Z. Lan, "Linear N-point camera pose determination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 774–780, Aug. 1999.

[9] R. Gomez-Ojeda, J. Briales, E. Fernandez-Moral, and J. Gonzalez-Jimenez, "Extrinsic calibration of a 2D laser-rangefinder and a camera based on scene corners," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 3611–3616.

[10] F. Vasconcelos, J. P. Barreto, and U. Nunes, "A minimal solution for the extrinsic calibration of a camera and a laser-rangefinder," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2097–2107, Nov. 2012.

[11] L. Zhou, "A new minimal solution for the extrinsic calibration of a 2D LIDAR and a camera using three plane-line correspondences," *IEEE Sensors J.*, vol. 14, no. 2, pp. 442–454, Feb. 2014.

[12] Z. Hu, Y. Li, N. Li, and B. Zhao, "Extrinsic calibration of 2-D laser rangefinder and camera from single shot based on minimal solution," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 4, pp. 915–929, Apr. 2016.

[13] N. Li, Z. Hu, and B. Zhao, "Flexible extrinsic calibration of a camera and a two-dimensional laser rangefinder with a folding pattern," *Appl. Opt.*, vol. 55, no. 9, pp. 2270–2280, 2016.

[14] J. Fan, Y. Huang, J. Shan, S. Zhang, and F. Zhu, "Extrinsic calibration between a camera and a 2D laser rangefinder using a photogrammetric control field," *Sensors*, vol. 19, no. 9, p. 2030, Apr. 2019.

[15] J. Briales and J. Gonzalez-Jimenez, "A minimal solution for the calibration of a 2D laser-rangefinder and a camera based on scene corners," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 1891–1896.

[16] F. Zhu, Y. Huang, Z. Tian, and Y. Ma, "Extrinsic calibration of multiple two-dimensional laser rangefinders based on a trihedron," *Sensors*, vol. 20, no. 7, p. 1837, Mar. 2020.

[17] Z. Tian, Y. Huang, F. Zhu, and Y. Ma, "The extrinsic calibration of area-scan camera and 2D laser rangefinder (LRF) using checkerboard trihedron," *IEEE Access*, vol. 8, pp. 36166–36179, 2020.

[18] C. Shi, K. Huang, Q. Yu, J. Xiao, H. Lu, and C. Xie, "Extrinsic calibration and odometry for camera-LiDAR systems," *IEEE Access*, vol. 7, pp. 120106–120116, 2019.

[19] L. Huang, F. Da, and S. Gai, "Research on multi-camera calibration and point cloud correction method based on three-dimensional calibration object," *Opt. Lasers Eng.*, vol. 115, pp. 32–41, Apr. 2019.

[20] J. Sui and S. Wang, "Extrinsic calibration of camera and 3D laser sensor system," in *Proc. 36th Chin. Control Conf. (CCC)*, Jul. 2017, pp. 6881–6886.

[21] Q. Pentek, P. Kennel, T. Allouis, C. Fiorio, and O. Strauss, "A flexible targetless LiDAR–GNSS/INS–camera calibration method for UAV platforms," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 294–307, Aug. 2020.

[22] D. W. Eggert, A. Lorusso, and R. B. Fisher, "Estimating 3-D rigid body transformations: A comparison of four major algorithms," *Mach. Vis. Appl.*, vol. 9, nos. 5–6, pp. 272–290, Mar. 1997.

[23] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE J. Robot. Autom.*, vol. 3, no. 4, pp. 323–344, Aug. 1987.

[24] J. Heikkila, "Geometric camera calibration using circular control points," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1066–1077, Oct. 2000.

[25] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

JAE-YEUL KIM received the B.S. degree in mechanical and automotive engineering from the Seoul National University of Science and Technology, Seoul, South Korea, in 2018, where he is currently pursuing the M.S. degree with the Graduate School of Automotive Engineering. His current research interests include visual surveillance using deep learning and scene understanding for autonomous navigation.

JONG-EUN HA received the B.S. and M.E. degrees in mechanical engineering from Seoul National University, Seoul, South Korea, in 1992 and 1994, respectively, and the Ph.D. degree in mechanical engineering from KAIST, Daejeon, South Korea, in 2000. From February 2000 to August 2002, he worked with Samsung Corning, where he developed an algorithm for a machine vision system. From 2002 to 2005, he worked with Multimedia Engineering, Tongmyong University. Since 2005, he has been a Professor with the Department of Mechanical and Automotive Engineering, Seoul National University of Science and Technology. His current research interests include deep learning, intelligent robots, and vehicles.