THESIS APPROVAL

GRADUATE SCHOOL, KASETSART UNIVERSITY

Master of Engineering (Information and Communication Technology for Embedded Systems)
DEGREE

Information and Communication Technology for Embedded Systems | Electrical Engineering
FIELD | **DEPARTMENT**

TITLE: A Novel Method for Extrinsic Parameter Estimation between a Single-Line Scan LiDAR and a Camera

NAME: Mr. Pakapoj Tulsuk

THIS THESIS HAS BEEN ACCEPTED BY

(Assistant Professor Miti Ruchanurucks, Ph.D.)
THESIS ADVISOR

(Mr. Teera Phatrapornnant, Ph.D.)
THESIS CO-ADVISOR

(Mr. Panu Srestasathiern, Ph.D.)
THESIS CO-ADVISOR

(Professor Hiroshi Nagahashi, D.Eng.)
THESIS CO-ADVISOR

(Assistant Professor Miti Ruchanurucks, Ph.D.)
DEPARTMENT HEAD

APPROVED BY THE GRADUATE SCHOOL ON _____

(Associate Professor Gunjana Theeragool, D.Agr.)
DEAN

THESIS

A NOVEL METHOD FOR EXTRINSIC PARAMETER ESTIMATION BETWEEN A SINGLE-LINE SCAN LiDAR AND A CAMERA

PAKAPOJ TULSUK

A Thesis Submitted in Partial Fulfillment of
the Requirements for the Degree of
Master of Engineering (Information and Communication Technology for Embedded Systems)
Graduate School, Kasetsart University
2014

Pakapoj Tulsuk 2014: A Novel Method for Extrinsic Parameter Estimation between a Single-Line Scan LiDAR and a Camera. Master of Engineering (Information and Communication Technology for Embedded Systems), Major Field: Information and Communication Technology for Embedded Systems, Department of Electrical Engineering. Thesis Advisor:
Assistant Professor Miti Ruchanurucks, Ph.D. 96 pages.

In this research, we present a new algorithm to calibrate the pose between a LiDAR and a camera. LiDAR in the thesis is a single line scan LiDAR. It scan its surrounding perimeter using a scan plane. The pose can be explained by a rotation matrix and a translation vector which describe the orientation and the position between the devices respectively. The calibration process estimate the parameters, the rotation matrix and translation vector, using perceived shared feature of each devices. The checkerboard is the feature because it can define easily in mathematic. The checkerboard are detected in the image using corner detection and LiDAR as a line since the intersection of planes is a line. The described geometry is realize into mathematic model, called a geometry constraint model. Therefore, the estimation model is introduced as the distance of the checkerboard scanned points and its intersection line.

The solution to solve the estimation model contains with two part, i.e. linear least-square and non-linear optimization. Linear least-square answer are used as initial guess for non-linear optimization since the acquired data have an uncertainty. The obtained answer is resolved according to rotation matrix properties, to achieve the answer that fit to the assumption.

The result shows that our estimation model is outperform that of Zhang and Pless. Our analysis shows that ours proposed model effectively estimates extrinsic parameters between a LiDAR and a camera.

_____ _____ __ / __ / __

Student's signature Thesis Advisor's signature

ACKNOWLEDGEMENTS

I would like to express my gratitude to my thesis advisor, Assist. Prof. Miti Ruchanurucks, whose expertise, understanding, and patience, added considerably to my graduate experience. I would like to thank the other members of my committee, Dr. Teera Phatrapornnant and Prof. Hiroshi Nagahashi for valuable comments and suggestion of the research project. Finally, I would like to gratefully thank Dr. Panu Srestasathiern from the Geo-Informatics and Space Technology Development Agency (GISTDA) for practicing me and giving advises of all the knowledges.

In conclusion, I recognize that this research would not have been possible without the financial assistance of Thailand Advanced Institute of Science and Technology - Tokyo Institute of Technology (TAIST-Tokyo Tech), National Science and Technology Development Agency (NSTDA), Tokyo Institute of Technology (Tokyo Tech) and Geo-Informatics and Space Technology Development Agency (GISTDA) and Kasetsart University (KU).

Pakapoj Tulsuk

July 2014

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF FIGURES (Continued)

A NOVEL METHOD FOR EXTRINSIC PARAMETER ESTIMATION BETWEEN A SINGLE-LINE SCAN LiDAR AND A CAMERA

INTRODUCTION

Nowadays, there are many applications that rely on both camera and LiDAR for data acquisition. In order to measure and use the data correctly, the alignment between the camera and the LiDAR must be accurately estimated. Many researchers have introduced methods for estimating extrinsic parameters between a LiDAR and a camera. Zhang and Pless (Zhang & Pless, 2004) proposed an algorithm for estimating orientation and position of a single-line scan LiDAR and a camera using checkerboard calibration pattern. Their algorithm and experimental setup are simple and practical. The estimating cost function is the orthogonal distances from the calibration pattern to LiDAR points. Our proposed (Tulsuk, Srestasathiern, Ruchanurucks, Phatrapornnant, & Nagahashi, 2014) method is similar to the work of Zhang and Pless (Zhang & Pless, 2004). Namely, we utilize the checkerboard as a calibration pattern because it can simply define and also practical. The contribution of this work is a novel cost function for extrinsic parameters estimation of a LiDAR and a camera. The proposed cost function is formulated as the orthogonal distances between 2D LiDAR points and a 2D line which is the intersection of LiDAR scan plane and the calibration plane. In order to detect LiDAR points which lie on the checkerboard automatically and robustly, we employ Hough transform to detect LiDAR points and RANSAC to filter out outliers. To obtain the optimal solution, non-linear optimization is performed by applying Levenberg-Marquardt algorithm.

OBJECTIVES

The purpose of this study is to derive extrinsic parameters which are orientation and position between a camera and a LiDAR accurately. Specifically, the objective of this thesis is as follows:

1. To study the calibration method between a LiDAR and a camera.

2. To compare the efficiency of the proposed method and that of Zhang and Pless (Zhang and Pless, 2004).

LITERATURE REVIEW

This chapter presents a fundamental idea of this thesis, related works and applied applications. Our work is a camera calibration, to find the precise location of the LiDAR with respect to the camera. Therefore, we review the camera model of Zhang (Zhang, 2000). Mainly, the work of Zhang and Pless (Zhang and Plass, 2004) is discussed because it is widely used, and use to compare in many works. The calibration of the camera and the LiDAR is mostly applied to survey work fields to align 3D points into images. Then, survey works are shown as applied applications to illustrate the usage of this work.

The related works explain the previous proposed methods, examine their geometry constraints and the solving methods. The applied applications clarifies the usage of our work.

Related Works

This section reviews all the related works of extrinsic parameters estimation between a camera and a LiDAR. They contain different kind of LiDAR such as a single line scan, a multiple line scan and a velodyne. The vision system is a pinhole camera model, mostly. The estimation methods are explained in many aspects of reviewer, including the advantages and the usability of the works.

Camera Calibration Zhang Model (Zhang, 2000)

Camera Calibration is an algorithm to estimate parameters of the camera model. The parameters include a rotation matrix and a translation vector for 3D transformation as known as extrinsic parameters. Furthermore, the camera model also include lens distortion, focal lengths and center of camera, called intrinsic parameters. Those parameters are estimated using information from real world and images, the corners on the checkerboard pattern. The estimation is a parameter minimization in least-square sense using a plane property.

He proposed a constraint of an observed plane in an image as follow:

$$s\tilde{\boldsymbol{m}} = \mathrm{A}[\mathrm{R} \quad \boldsymbol{t}]\tilde{\boldsymbol{M}} \quad , \tag{1}$$

where $\tilde{\boldsymbol{m}} = [u \quad v \quad 1]^{\mathrm{T}}$ is a 2D point in image coordinate system, $\tilde{\boldsymbol{M}} = [x \quad y \quad z \quad 1]$ 3D points in real world coordinate system, and s is an arbitrary scale factor. The plane coordinate systems can be transformed to the camera coordinate system using rotation matrix R and translation vector $\boldsymbol{t}$. After the transformation, the coordinate system is going through camera lens. Then, intrinsic matrix

$$\mathrm{A} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

is applied, given (u_0, v_0) is the principle point of the camera, α and β are the scale factors in image u and v axes, and γ the skewness between two axes.

With the definition of plane, the representation of point $\tilde{\boldsymbol{M}}$ can be reduced to $[x \quad y \quad 1]^{\mathrm{T}}$; $z = 0$. Therefore, the constraint can be reformed as $s\tilde{\boldsymbol{m}} = \mathrm{H}\tilde{\boldsymbol{M}}$, given $\mathrm{H} = \mathrm{A}[\mathrm{r}_1 \quad \mathrm{r}_2 \quad \boldsymbol{t}]$. Since z is zero, the third column of $\mathrm{R} = [\mathrm{r}_1 \quad \mathrm{r}_2 \quad \mathrm{r}_3]$ is removed.

Solved the constraint (1) for H and recovered the removed column with orthogonal property, then all of the parameters are obtained. This work is applied into the procedure of our work to estimate the transformations of the checkerboard to the camera. The camera model is also applied to project 3D points into images. For the reason that we can roughly confirm the estimation results using this projection.

Wasielewski and Strauss (Wasielewski & Strauss, 1995)

This work is the very first study for extrinsic parameter estimation between a camera and a LiDAR. (Wasielewski & Strauss, 1995) explained that “A calibration process based upon a specific calibration pattern is used to identify such a

transformation. Adapted procedures define a geometric relationship between these two sensors", see Figure 1.

The constraint of this experimental setup is created from a unique individual of the calibration pattern which is a composition of two plane. The calibration pattern composes with mainly two characteristics, see Figure 1. First, each plane have difference color (black and white). The line at the separation of those two planes is a co-feature point. Last, the angle between two planes is specified to achieve the best estimation result. With these characteristics, both the camera and the LiDAR can detect the separation line.

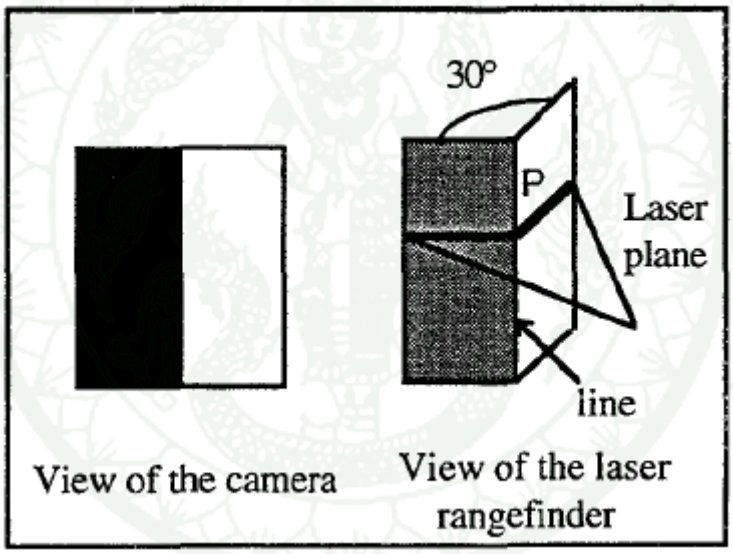


Figure 1 V-shape calibration pattern.

Source: Wasielewski and Strauss (1995)

The geometric transformation of camera and LiDAR is introduced with three rotation angles θ_1, θ_2, θ_3 (row, pitch, yaw) and three translation t_x, t_y, t_z. Let $w = f(\theta_1, \theta_2, \theta_3)$ be the matrix of rotation. Then the transformation from LiDAR coordinate system t to camera coordinate system c can be presented as:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad . \tag{3}$$

Since LiDAR data represents in polar coordinate system, see Figure 2, the data must be converted to cartesian coordinate system. Then LiDAR point can be defined as:

$$\begin{cases} x_t = \rho \cdot cos\theta \\ y_t = \rho \cdot sin\theta \\ z_t = 0 \end{cases} , \tag{4}$$

where ρ is a distance to the scanned object and θ an angle reference from X axis.

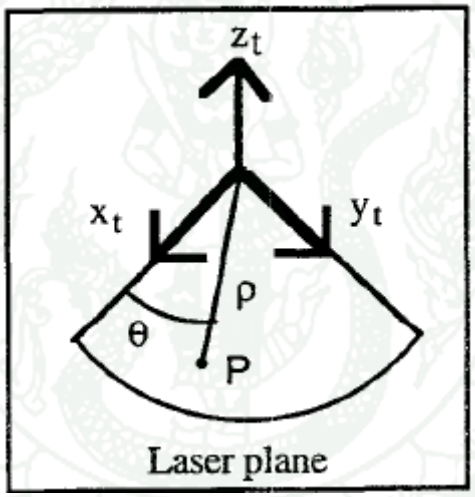


Figure 2 Polar coordinate system.

Source: Wasielewski and Strauss (1995)

Furthermore, there is a projective transformation to projecting a 3D point onto image plane (u, v), which is

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{f}{z_c} \begin{bmatrix} k_u & 0 & \frac{u_0}{f} \\ 0 & k_v & \frac{v_0}{f} \\ 0 & 0 & \frac{1}{f} \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \quad , \tag{5}$$

where f is focal length, k_u and k_v the scale factors along the axes, u_0 and v_0 the principal point. The parameters related to the camera model are estimated using camera calibration. Then LiDAR points can be transformed onto image using:

$$\begin{cases} x_c = a_0 x_t + b_0 y_t + c_0 \\ y_c = a_1 x_t + b_1 y_t + c_1 \\ z_c = a_2 x_t + b_2 y_t + t_3 \end{cases} \quad , \tag{6}$$

where a_i, b_i and c_i are nonlinear functions of the extrinsic parameters.

The separation line is detected in image, and defined as:

$$(A \cdot u) + (B \cdot v) = 0 \quad , \tag{7}$$

where A and B are the line parameters. In the other hands, LiDAR detects the points at the intersection of the planes as a point that maximizes the curvature of LiDAR data. Assuming that the point lies on the separation line, then the constraint is created by combining (5), (6) and (7) so they had:

$$\left.\begin{array}{c} A \cdot a_0 \cdot x_t + A \cdot b_0 \cdot y_t + A \cdot c_0 \\ +B \cdot a_1 \cdot x_t + B \cdot b_1 \cdot y_t + B \cdot c_1 \\ -a_2 \cdot x_t - b_2 \cdot y_t - t_3 \\ = 0 \end{array}\right\} \quad . \tag{8}$$

They conclude that “The originality of this calibration method is to match the image of a line in the 3D space with the point resulting in the intersection between the line and the laser plane”. This work shows several results of the minimized constraint equation; the result of nonlinear least squares (Gauss Newton), recursive least squares, recursive weighed least squares. In conclusion, the problem of extrinsic

parameter estimation between a camera and a LiDAR can be solved using their geometric constraint, and the nonlinear least squares minimizing model yields the best result out of the other minimizing models.

(Li *et al*., 2007), They describe their proposed method that "On the basis of an analysis of three possible features, we propose to design a right-angled triangular checkerboard and to employ the invisible intersection points of the laser range finder's slice plane with the edges of the checkerboard to set up the constraints equations. The extrinsic parameters are then calibrated by minimizing the algebraic errors between the measured intersections points and their corresponding projections on the image plane of the camera." Stereo vision system allows them to investigate 3D point location, only for reference. The reliability of the calibration result can be compared using the stereo vision system. The checkerboard is shown in Figure 3.

The distortion model is explained as follow: a point $E_c(x_E, y_E, z_E)$ in the camera coordinate system can be projected into image plane as a point in the image $e_c(x_e, y_e)$, and $d_c(x_d, y_d)$ is normalized coordinate system when lens distortion is included. The model can be explained using distortion coefficients k$(k_1, k_2, k_3, k_4, k_5)$ and intrinsic parameters K with the following equation:

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \left((1 + k_1)r^2 + k_2 r^4 + K_5 r^6\right) \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2k_3 xy + k_4(r^2 + 2x^2) \\ k_3(r^2 + 2y^2) + 2k_4 xy \end{bmatrix} \quad , \quad (9)$$

where $x = \frac{x_E}{z_E}$, $x = \frac{y_E}{z_E}$ and $r = \sqrt{x^2 + y^2}$. The undistorted point can be represented in image plane as follow:

$$\begin{bmatrix} x_e \\ y_e \\ 1 \end{bmatrix} = K \begin{bmatrix} x_d \\ y_d \\ 1 \end{bmatrix} \quad . \quad (10)$$

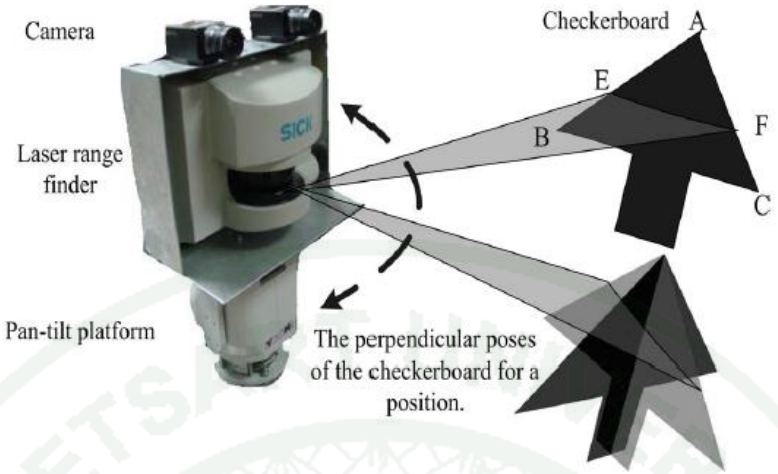


Figure 3 The LiDAR observes the arrow shape checkerboard.

Source: Li *et al*. (2007)

As shown in Figure 3, the geometric constraint is created. Edges AB and AC are projected into image plane as edges ab and ac respectively which can be detected using image processing. E_L is a point observed by the LiDAR. e is the projected point in the camera coordinate system using rotation matrix and translation vector with respect to E in the world coordinate system. Thus, the distance $d(e, ab)$ of point e to the line ab can be calculated using the following equation:

$$d(e, ab) = \frac{\left\|\overrightarrow{eb} \times \overrightarrow{ab}\right\|}{\left\|\overrightarrow{ab}\right\|} \quad , \tag{11}$$

where $\times$ is the cross product, $\vec{\cdot}$ a vector notation and $\|\cdot\|$ the norm of vector. Similarly, the other edge can also be calculated such as $d(f, ac)$. Then, a rotation matrix R and a translation vector $\boldsymbol{T}$, whose minimize the equation

$$\min_{\mathrm{R},\boldsymbol{T}} \left\{ \sum_{i=1}^{N} [d(e_i, a_i b_i) + d(f_i, a_i c_i)]^2 \right\} \tag{12}$$

are the optimum pose of the LiDAR, where N is the number of calibration pattern poses.

The minimizing equation is solved using Gauss Newton algorithm, and yields a fine result refer to the comparison with stereo vision and simulated data. Intrinsic parameters and distortion coefficients are given. There are many of advantages of using a simulation such as environment controlling, additive noise level and noise distribution. The input for the calibration method are the edges of the calibration pattern and the points located at the edges. The camera detected the edges and the points are detected by the LiDAR.

(Huang and Barth, 2009), This work propose a calibration method for a multi-planar LiDAR. Multi-planar LiDAR is a LiDAR with multiple scan planes, see Figure 4. It allows the device acquire more information. Each plane is set in different scan angles along vertical axis of the device. They explained that "Geometric constraints of the 'views' from the LIDAR and camera images are resolved as the coordinate transformation coefficients". Their constraints minimize the distance of all scanning points and the calibration plane.

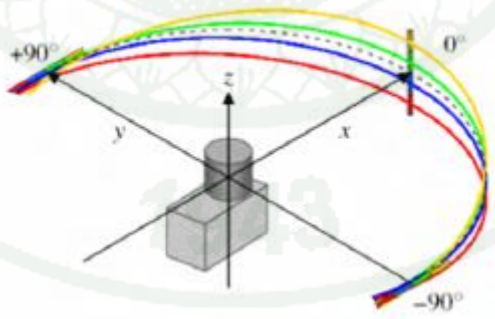


Figure 4 Multi-planar LiDAR.

Source: Huang and Barth (2009)

They describe that the checkerboard plane $z = 0$ in the world coordinate system c can be transformed from the world coordinate system w to another coordinate system by rotation R and translation $\boldsymbol{T}$. Rotation matrix R is a combination of three columns vector $\mathrm{R} = [\boldsymbol{r}_1 \quad \boldsymbol{r}_2 \quad \boldsymbol{r}_3]$. The third column $\boldsymbol{r}_3$ is the surface normal vector of the checkerboard in the camera coordinate system. Point $\boldsymbol{P}^c$ is a point on the checkerboard observed by the camera. Since $\boldsymbol{T}$ is known as the vector from the origin of camera coordinate system to the world's, then they introduced vector $\vec{v} = \boldsymbol{P}^c - \boldsymbol{T}$ that define as a vector on the checkerboard plane, see Figure 5. Recall that $\boldsymbol{r}_3$ is also a vector but orthogonal to the plane. Thus they induced the constraint as an inner product of two vectors $\boldsymbol{r}_3 \cdot \vec{v} = 0$ since they are orthogonal. The constraint had been reformed as:

$$\boldsymbol{r}_3 \cdot \mathrm{R}_{\mathrm{l}}^{\mathrm{c}} \boldsymbol{P}^l + \boldsymbol{T}_{\mathrm{l}}^{\mathrm{c}} - \boldsymbol{T} = 0 \quad , \tag{13}$$

where $\mathrm{R}_{\mathrm{l}}^{\mathrm{c}}$ and $\boldsymbol{T}_{\mathrm{l}}^{\mathrm{c}}$ are rotation matrix and translation vector that define the transformation from the LiDAR coordinate system to the camera coordinate system.

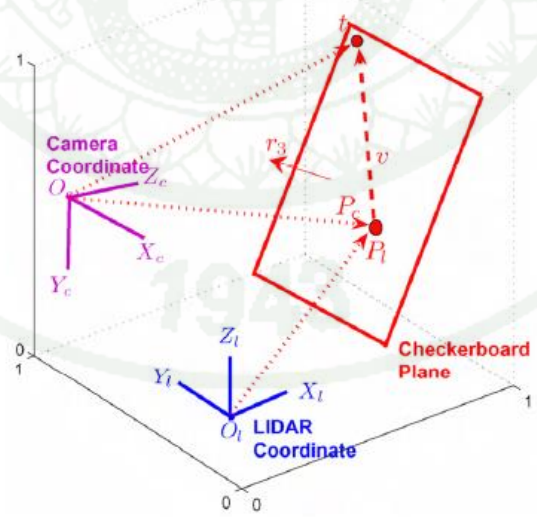


Figure 5 The experiment is described by vectors.

Source: Huang and Barth (2009)

The results from the simulated data shows that the more efficiency we need, the number of checkerboard poses is require. Lagrange method is used to derive the constraint in closed-form solution. Maximum Likelihood is included to minimize the constraint and yield the optimum result.

This work represents the simplicity of the geometric interpretation i.e. the definition of checkerboard plane $z = 0$, and an alternative solution for least square problem. Unfortunately, the different hardware still be a problem in term of comparison.

(Kwak *et al*., 2011), They introduced an extended work of Wasielewski and Strauss which is the solution to the parameter estimation between a LiDAR and a camera using a V-shape calibration pattern. They introduce the geometry constraint as "The extrinsic calibration parameters are estimated by minimizing the distance between corresponding features projected onto the image plane. The features are edge and centerline features on a v-shaped calibration target". The main contribution of this are two way to improve the calibration accuracy. First, "use different weights to distance between a point and a line feature according to the correspondence accuracy of the features". Second, "apply a penalizing function to exclude the influence of outliers in the calibration data sets".

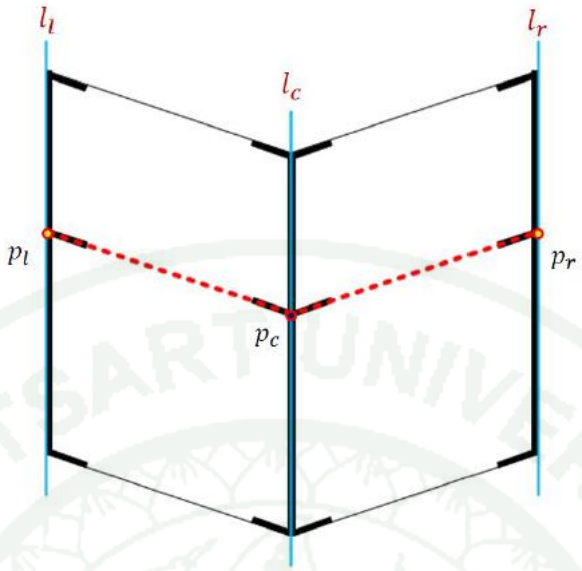


Figure 6 V-shape calibration pattern with three stipes.

Source: Kwak *et al.* (2011)

The cost function, the distance between point p_i and line l_i must be zero when no noise appears, of each feature point and detected line $d(p_i, l_i)$ is zero. Then, it can be written as:

$$E(\mathrm{R}, \boldsymbol{T}) = \sum_{i=1}^{N} d^i(p_l, l_l)^2 + d^i(p_c, l_c)^2 + d^i(p_r, l_r)^2 = 0 \quad , \qquad (14)$$

where $E(\mathrm{R}, \boldsymbol{T})$ is error function that optimizing rotation matrix R and translation $\boldsymbol{T}$ vector, the transformation between the camera and the LiDAR, and N the number of calibration pattern poses.

There are two sub-methods to improve the result. First, they applied different weights to the constraint at different feature points. Since the LiDAR points can be sparse, the feature points can be far from the true position. Points at the edges, p_l and p_r, are replaced with expected values of the neighboring points see Figure 7. The expected value $E(\cdot)$ is computed using two consecutive points around the edge, e.g.

$p_l = E(p_L^j, p_L^{j+1})$. The farther point is used to average the true location of the edge point. The center point p_c also replaced with the intersected point of two fitted line, i.e. left and right line, which precisely be at the center of the pattern, see Figure 8. Thus every feature points have difference uncertainties. Weighting them differently then yield a better result. The weights w are inverse of residual of each feature points so the error function can be rewritten as:

$$E(\mathrm{R}, \boldsymbol{T}) = \sum_{i=1}^{N} w_l \cdot d^i(p_l, l_l)^2 + w_c \cdot d^i(p_c, l_c)^2 + w_r \cdot d^i(p_r, l_r)^2 = 0 \quad . \quad (15)$$

Second, a penalizing function is included to reduce the influence of outliers in the calibration data set. Huber penalty function $\phi_{huber}(\cdot)$ is used to reweight each of LiDAR data set since the data set may contains outliers which affect to the estimation model drastically. The penalty function is applied to the cost function d as follow:

$$\phi_{huber}(d) = \begin{cases} d^2 & if\ |d| \le d_{max} \\ d_{max}(2 \cdot |d| - d_{max}) & if\ |d| > d_{max} \end{cases} \quad , \quad 16)$$

where d_{max} is a threshold for weighted points that lies beyond. The threshold can be computed by: $d_l, d_r \lesssim d_{max} = f\tan\left(\frac{\theta}{2}\right)$, where f is the focal length of the camera and θ is the angular resolution between successive LiDAR points.

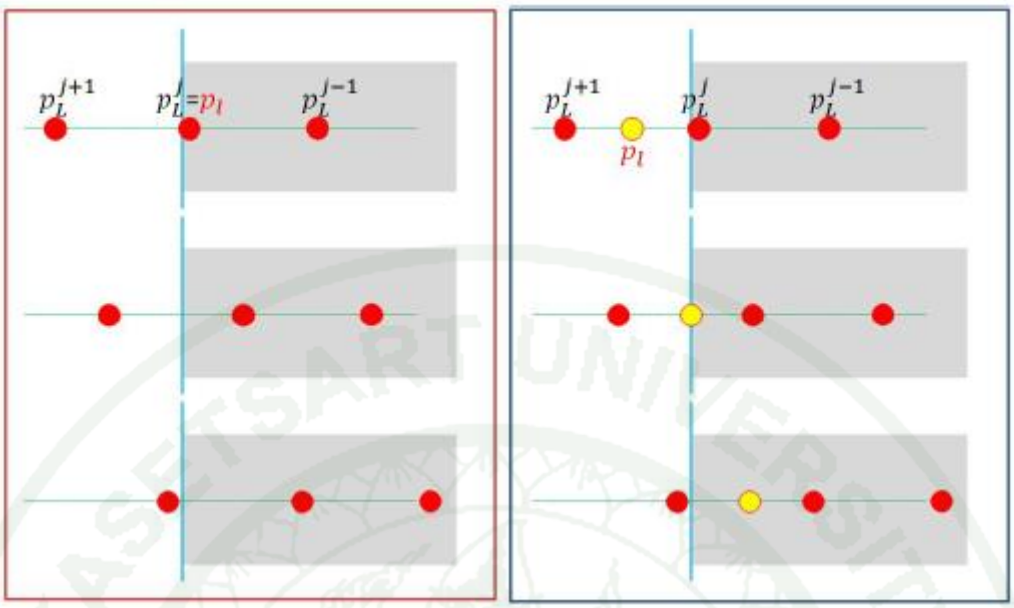


Figure 7 The problem of point displacement at the edges.

Sorece: Kwak *et al.* (2011)

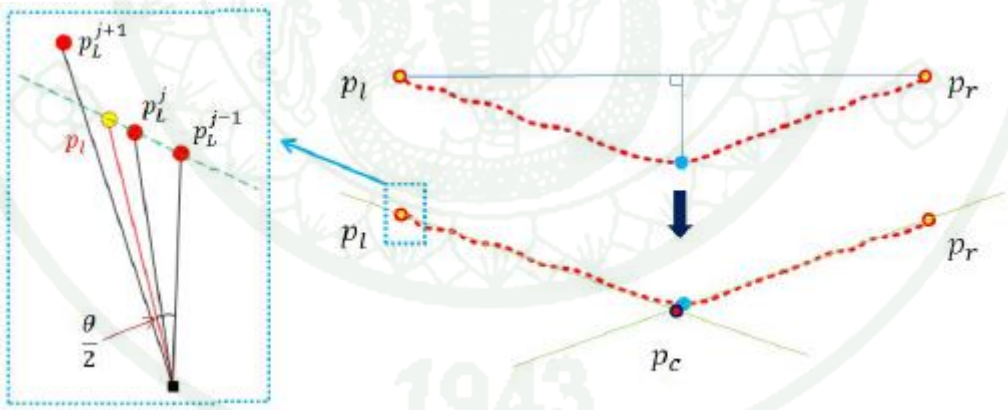


Figure 8 The approximation of new points.

Source: Kwak *et al.* (2011)

In conclusion, they summarize that “the alignment accuracy of our method is two times better than the compared state of the art algorithms”, “our method requires

fewer image/scan pairs to achieve the same calibration accuracy than the previous methods" and "our method has robust performance regardless of pose and range of the calibration target".

(Zhang and Pless, 2004**),** This work presents an estimation method for extrinsic parameters between a camera and a single line scan LiDAR which is the most familiar work with ours. They explained that "The calibration is based on observing a planar checkerboard pattern and salving for constraints between the views of a planar checkerboard calibration pattern from a camera and laser range finder". And the proposed solution is given that is "a direct solution that minimizes an algebraic emr from the constraint, and subsequent nonlinear refinement minimizes a re-projection error". The results of the estimation are rotation matrix and translation vector between the camera and the LiDAR. The results are then compared with ground truth in a simulation to show the proposed can be effectively used.

They define a constraint of the plane that visible to both devices using homography H which is a composition of rotation matrix and translation vector. The checkerboard are detected as 2D LiDAR scanning points manually. In the images, the plane is detected using camera calibration algorithm of Zhang. Consequently, the transformations from the planes to the camera are obtained.

The transformation from plane coordinate system w to image coordinate system i is defined by $s\boldsymbol{p}^i = \mathrm{K}(\mathrm{R}\boldsymbol{P}^w + \boldsymbol{t})$, Zhang model, (Zhang, 2000) where K is intrinsic parameters matrix. The transformation from camera coordinate system c to LiDAR coordinate system f is defined by $\boldsymbol{P}^f = \Phi\boldsymbol{P}^c + \boldsymbol{\Delta}$, where Φ and $\boldsymbol{\Delta}$ are rotation matrix and translation vector between the camera and the LiDAR. To prevent a confusion, we define the transformation from world coordinate system to camera coordinate system as: $\boldsymbol{P}^c = \mathrm{R}\boldsymbol{P}^w + \boldsymbol{t}$.

The checkerboard plane can be defined by vector $\boldsymbol{N}$ which is a parallel vector to the normal and its magnitude $\|\boldsymbol{N}\|$ equals to the distance from the camera to the plane. Therefore, $\boldsymbol{N} = -\boldsymbol{r_3}(\boldsymbol{r_3}^{\mathrm{T}} \cdot \boldsymbol{t})$ is given. Then the constraint

$$\boldsymbol{N} \cdot \Phi(\boldsymbol{P}^f - \boldsymbol{\Delta}) = \|\boldsymbol{N}\|^2 \tag{17}$$

is proposed since the scanned points lie on the plane. By solving the constraint for Φ and $\boldsymbol{\Delta}$, the pose between the camera and the LiDAR is obtained. Moreover, they also refine the obtained parameters using Levenberg-Marquardt to reduce error of noise in LiDAR and image, which are called global optimization.

This work shows the effective and practical method to solve the problem thus it is the most wildly used and worth to compare with. Our proposed method is compared with this work. We also learn that the simulation is a good choice to analyze the displacement of the estimated results to the ground truth. For those reasons, this work will be called as the state of the art for all of the thesis.

(Vasconcelos *et al*., 2012), They describe that "the problem of estimating the rigid displacement between the two sensors is formulated as the one of registering a set of planes and lines in the 3D space. It is proved for the first time that the alignment of 3 plane-line correspondences has at most 8 solutions, that can be determined by solving a standard *p3p* problem and a linear system of equations". That leads to the main contribution of this work that to show the minimal solution for extrinsic parameters estimation of between the LiDAR and the camera

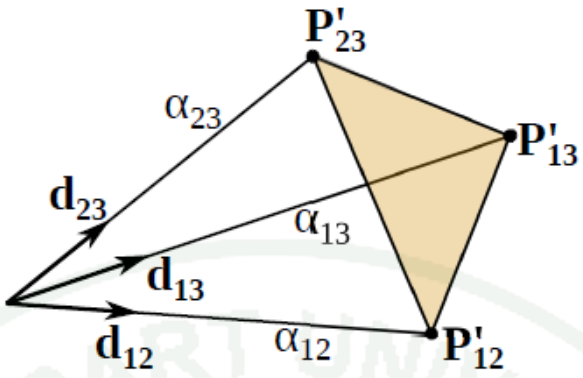


Figure 9 P3P configuration.

Source: Vasconcelos *et al.* (2012)

Perspective-3-points (P3P) problem is similar to the solution to find the pose of the object pose by solving fourth order equation. The geometric constraint is formed as P3P like to present the problem alternatively. Although P3P problem already have the solution, the difficulty and novelty are in the adaptation and reformation of the constrain equation. The coordinate systems transformation is defined as: $\boldsymbol{Q}' = \mathrm{H}\boldsymbol{Q}$, where $\boldsymbol{Q}$ and $\boldsymbol{Q}'$ are notations for the old point coordinate system and the new one respectively. The point coordinate system is denoted by $[x \quad y \quad z \quad 1]^{\mathrm{T}}$. Homography $\mathrm{H} = \begin{bmatrix} \mathrm{R} & \boldsymbol{T} \\ \mathbf{0}_3{}^{\mathrm{T}} & 1 \end{bmatrix}$ is the transformation matrix. Plane transformation also defined as: $\boldsymbol{\Pi}' = \mathrm{H}^{-\mathrm{T}}\boldsymbol{\Pi}$. Plane $\boldsymbol{\Pi}$ can be transformed to new coordinate system by inverse transpose of homography, where plane is $\boldsymbol{\Pi} = [\mathbf{n}^{\mathrm{T}} \quad 1]^{\mathrm{T}}$, and $\mathbf{n}$ is plane parameters.

Splitting the homography into two separated part; a rotation and a translation, then reformed the constraint equations as P3P problem to find each part separately. The new geometric constraint is proposed as: $\alpha_{ij}\boldsymbol{d}_{ij} = \mathrm{R}^{\mathrm{T}}(\boldsymbol{P}_{ij}{}' + \boldsymbol{m})$, where α is a distance to a point, $\boldsymbol{d}$ a direction to the point, $\boldsymbol{m}$ the transformed plane parameters and $\boldsymbol{P}$ a variable that denoted the transformed point. The subscripts i and j express the direction from source i to destination j, as shown in Figure 9. For further details

please look (Vasconcelos *et al*., 2012). At least three points are required to solve the new constraint for the rotation matrix. Then the minimal solution can be written as:

$$\begin{cases} \alpha_{12}\boldsymbol{d}_{12} = \mathrm{R}^{\mathbf{T}}(\boldsymbol{P}_{12}{}' + \boldsymbol{m}) \\ \alpha_{13}\boldsymbol{d}_{13} = \mathrm{R}^{\mathbf{T}}(\boldsymbol{P}_{13}{}' + \boldsymbol{m}) \\ \alpha_{23}\boldsymbol{d}_{23} = \mathrm{R}^{\mathbf{T}}(\boldsymbol{P}_{23}{}' + \boldsymbol{m}) \end{cases} . \quad (18)$$

The rotation matrix R can be estimated using the solution of P3P. Then, the equations are rearranged into P3P form as:

$$\begin{cases} \|\alpha_{12}\boldsymbol{d}_{12} - \alpha_{13}\boldsymbol{d}_{13}\| = \|\boldsymbol{P}_{12}{}' - \boldsymbol{P}_{13}{}'\| \\ \|\alpha_{12}\boldsymbol{d}_{12} - \alpha_{23}\boldsymbol{d}_{23}\| = \|\boldsymbol{P}_{12}{}' - \boldsymbol{P}_{23}{}'\| \\ \|\alpha_{13}\boldsymbol{d}_{13} - \alpha_{23}\boldsymbol{d}_{23}\| = \|\boldsymbol{P}_{13}{}' - \boldsymbol{P}_{23}{}'\| \end{cases} . \quad (19)$$

Since $\boldsymbol{d}$ is a unit direction vector, its square equals to one. By apply the law of cosines, the equations become P3P form;

$$\begin{cases} {\alpha_{12}}^2 + {\alpha_{13}}^2 - \alpha_{12}\alpha_{13}{\boldsymbol{d}_{12}}^{\mathrm{T}}\boldsymbol{d}_{13} = \|\boldsymbol{P}_{12}{}' - \boldsymbol{P}_{13}{}'\|^2 \\ {\alpha_{12}}^2 + {\alpha_{23}}^2 - \alpha_{12}\alpha_{23}{\boldsymbol{d}_{12}}^{\mathrm{T}}\boldsymbol{d}_{23} = \|\boldsymbol{P}_{12}{}' - \boldsymbol{P}_{23}{}'\|^2 \\ {\alpha_{13}}^2 + {\alpha_{23}}^2 - \alpha_{13}\alpha_{23}{\boldsymbol{d}_{13}}^{\mathrm{T}}\boldsymbol{d}_{23} = \|\boldsymbol{P}_{13}{}' - \boldsymbol{P}_{23}{}'\|^2 \end{cases} . \quad (20)$$

The rotation matrix should be obtained, after solving Equation (20). Consequently, the rotation matrix is used to estimate the translation vector in order.

The experiment collects series of checkerboard poses, which is visible to both devices. All of the checkerboard poses are used to resolve the constraint in PnP manner. Furthermore, the iterative process is applied to accurately locate the optimum result. They claim that their constraint will reach to the optimum answer with any initial guess, the post the LiDAR reference to the camera. Although five checkerboard poses can yield the proper parameters, theoretically, it is shown that only three poses are enough to solve the constraint. Above of all, the algorithm shows that it yields more accuracy than the previous studies in minimum case.

(Zhou and Deng, 2012), They proposed two different ways to calibrate a LiDAR using a checkerboard. The LiDAR is a velodyne LiDAR. Its field of scan is half sphere shape. This device allows user to fully construct 3D structure as a set of 3D points simultaneously. However, the points can be colorize by combining camera data. In order to colorize the data of the camera, the data must be transformed using rotation matrix and translation vector. Since the transformation from the world coordinate system to the camera coordinate system is known (Zhang, 2000), the transformation between the camera and the LiDAR must be solved.

Study Number One (Zhou an Deng, 2012)

The study shows a novel interpretation of geometry between the camera, the LiDAR and the checkerboard. They said that "Since Line correspondences are used to construct the geometric constraint in our algorithm, the calibration target is flexible and any object with straight line boundary can be an appropriate calibration target". A 3D line $\boldsymbol{L}_{3D}$ in the world coordinate system is represented in image as 2D line $\boldsymbol{l}_{2D}$ then the geometry $p_c^{\mathrm{T}}\boldsymbol{l}_{2D} = 0$ is satisfied where p_c is a point in the camera coordinate system. A point in the LiDAR coordinate system l is transformed to the image coordinate system u using rotation matrix R, translation vector $\boldsymbol{T}$ and intrinsic matrix K with transformation equation as follow:

$$\boldsymbol{p}_u = s\mathrm{K}[\mathrm{R} \quad \boldsymbol{T}][\boldsymbol{p}_l^{\mathrm{T}} \quad 1]^{\mathrm{T}} \quad , \tag{21}$$

where s is a scale value. In short, they given $\mathrm{P} = s\mathrm{K}[\mathrm{R} \quad \boldsymbol{T}]$, and then obtain the geometric constraint

$$\boldsymbol{l}_{2D}^{\mathrm{T}}\mathrm{P}[\boldsymbol{p}_l^{\mathrm{T}} \quad 1]^{\mathrm{T}} = 0 \quad . \tag{22}$$

As shown in Figure 10, the lines are checkerboard edges that observed by the camera so these line can be detected by an existing algorithm. In the other hands, the

LiDAR also observed the same checkerboard. Two ending points of the checkerboard edges are detected. Then, the points are used for the calibration.

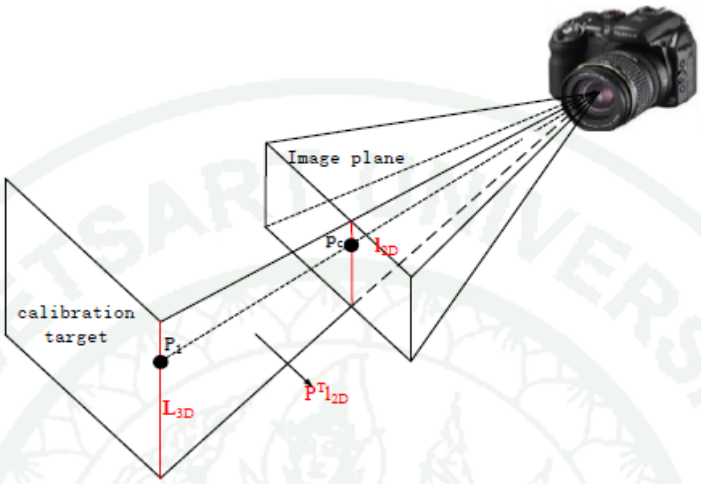


Figure 10 The projection of 3D line on image is represented by 2D line instead.

Source: Zhou *et al.* (2012)

The constraint is reformed into $A\boldsymbol{x} = 0$. Since the data for calibration is acquired from N checkerboard poses and each pose i have 2 ending points. Each set yields one constraint, the constraint is stack over each another, then became matrix A. Vector x contained the combination of parameters of extrinsic and intrinsic parameter. By using of SVD over the constraint with the condition $\|\boldsymbol{x}\| = 1$, the parameters are obtained. Since the intrinsic parameters are known (Zhang, 2000), the extrinsic parameter can be resolved from vector $\boldsymbol{x}$.

This work shows the novel method to estimate the extrinsic parameters based on the assumption of the distance between points and line, which can be useful and effectively used. Although SVD is perfectly used to solve an equation in least-square sense, the condition have cause a complexity in order to derive the true values.

Study Number Two (Zhou and Deng, 2012)

They claimed that "Since the interaction between the estimation errors of the plane parameters obtained from checkerboard images downgrades the quality of extrinsic calibration results, a new geometric constraint is presented to decouple the rotation from the translation so as to reduce the effect of such an interaction".

The new estimation model between the camera and the LiDAR is presented by resolving the geometry of the checkerboard. The checkerboard has it a normal vector $\boldsymbol{n}_i$ for the $i-th$ pose. The normal can be observed in LiDAR perspective, LiDAR coordinate system, using the relation

$$\boldsymbol{n}_i^l = \mathrm{R}^{\mathrm{T}} \cdot \boldsymbol{n}_i^c \quad , \tag{23}$$

where R is rotation matrix that transforms the camera coordinate system c to the LiDAR coordinate system l. Let vector $\mathbf{P}_{i,j}^{l'} = \mathbf{P}_{i,j}^{l} - \overline{\mathbf{P}}_i^l$, where $\overline{\mathbf{P}}_i^l$ is the centroid of the checkerboard and $\mathbf{P}_{i,j}^{l}$ the $j-th$ point that lies on the checkerboard scanned by LiDAR. Since the point $\mathbf{P}_{i,j}^{l'}$ is a point on the plane, then $\boldsymbol{n}_i^{l\mathrm{T}}\mathbf{P}_{i,j}^{l'} = 0$. Reformed the equation to:

$$(\boldsymbol{n}_i^c)^{\mathrm{T}}\left(\mathrm{R}^{\mathrm{T}} \cdot \left(\mathbf{P}_{i,j}^{l} - \overline{\mathbf{P}}_i^l\right)\right) = 0 \quad . \tag{24}$$

By deriving Equation (24), the rotation matrix can be obtained. In the other hands, the translation vector $\boldsymbol{t}$ can be resolved using a constraint

$$(\boldsymbol{n}_i^c)^{\mathrm{T}}\boldsymbol{t} = -(\boldsymbol{n}_i^c)^{\mathrm{T}}\left(\mathrm{R}\mathbf{P}_{i,j}^{l} - \boldsymbol{P}_i^c\right) \quad , \tag{25}$$

where $\boldsymbol{P}_i^c$ is the origin of checkerboard. Equation (25) is the solution of Equation (13), refer to (Huang & Barth, 2009). Figure 11 shows the geometric constraint.

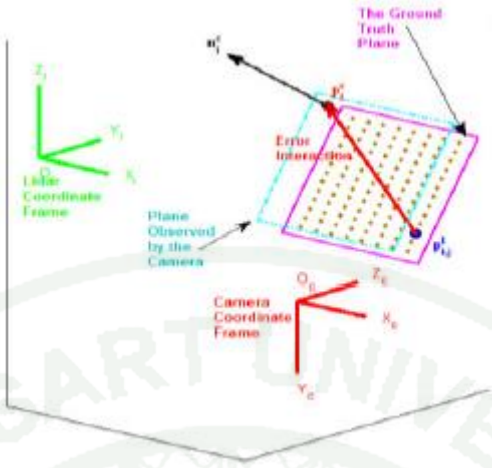


Figure 11 The experimental configuration for the interpretation of geometry constraint equation.

Source: Zhou and Deng (2012)

By solving equation (24) and (25), the rotation matrix and translation vector are obtained respectively. Equation (24) is solved by SDV, after reformed (24) into $A\boldsymbol{x} = \mathbf{0}$. Equation (25) is reformed into $A\boldsymbol{x} = \boldsymbol{B}$, then applied pseudo inverse. The obtained rotation matrix have to be resolved because it does not implied to the characteristic of rotation matrix. Furthermore, RANSAC is included into the process to remove all outliers such as invalid LiDAR points or invalid checkerboard poses. Levenberg-Marquardt is applied to optimize the rotation matrix and translation vector.

Although the result shows that the method yielded significant result than the work of Huang and Barth (Huang and Barth, 2009) because the rotation are estimated separately. Translation vector are not estimated under the assumption $\|\boldsymbol{x}\| = 1$. The size of the answers will not mixed up, since rotation parameters and translation parameters can have difference size.

Applied Application

This section helps to clarify the usability of our work. Many of application applied LiDAR in order to investigated 3D position. By including a camera, color perception is integrated into the system. That is, the observed 3D position is literally classified. The classified point has many advantages such as analyzing, detection or displaying. The devices must be mounted accurately in order to transform the data come one device to another. The mounting process contains significant error from uncontrollable environment, then the alternative way to handle such task could be finding the location using the observe data.

(Douillard *et al*., 2008), This work is Simultaneous Localization and Mapping (SLAM) system. A single line scan LiDAR and a camera are integrated in to the system. The system is mounted on a vehicle to collect the data. The acquired data is computed simultaneously to construct a map of the surveyed paths. The system computed the odometry of the vehicle by solving the relation of the moving object, 3D point. Therefore, the points must be unique to distinguish each object in the real world, and use them for odometry estimation.

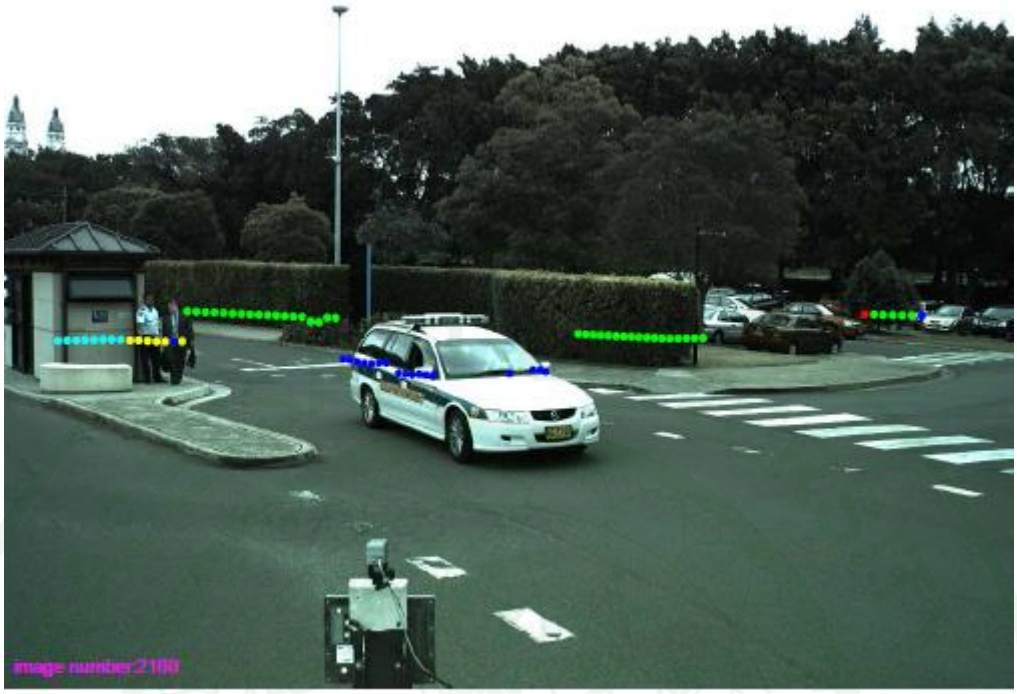


Figure 12 The collected data used to estimate the odometry for SLAM.

Source: Douillard *et al.* (2008)

Figure 12 show the result of the classification. Image data is used to classify each object while LiDAR data is transform into image. The projected point is colorized with respect to the cluster of classified objects. The transformation of LiDAR to the system must be known by the calibration process.

(Premebida *et al*., 2007)**,** The study presents a method for pedestrian detection by integrating depth information to expand the understanding to the detected object. The devices, a LiDAR and a camera, are mounted on a vehicle designing for intelligent vehicle system. The geometry of the devices must be derived in order to combine their data together. The geometry can be described by a basic transformation, rotation and translation. Then, the calibration method is the essential issued for the task.

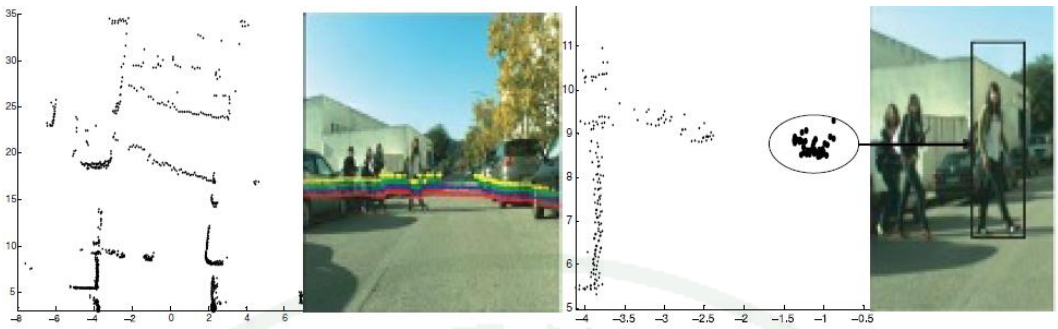


Figure 13 The fusing data from a camera and a multi-line scan LiDAR.

Source: Premebida *et al.* (2007)

Figure 13 shows the fusing data from a camera and a multi-line scan LiDAR. Human is detected in images by adding an information of depth into the system. It helps to emphasize the meaning of the detected object.

(Romos *et al*., 2007)**,** They proposed a SLAM model to reconstruct a map by using the reprojected poses of the specific feature which is a tree, as shown in Figure 14. To detect a tree, image processing is required. The pose of a single line scan LiDAR and a camera must be known to reproject a point from LiDAR coordinate system into an image. Feature points, trees, are detected then the location of the robot can be found by register those trees into a specific coordinate system, as shown in Figure 15. The location of the robot can be tracked by resolving their SLAM model using the previous pose of the robot and the new one.

Figure 14 The LiDAR data on feature points.

Source: Romos *et al.* (2007)

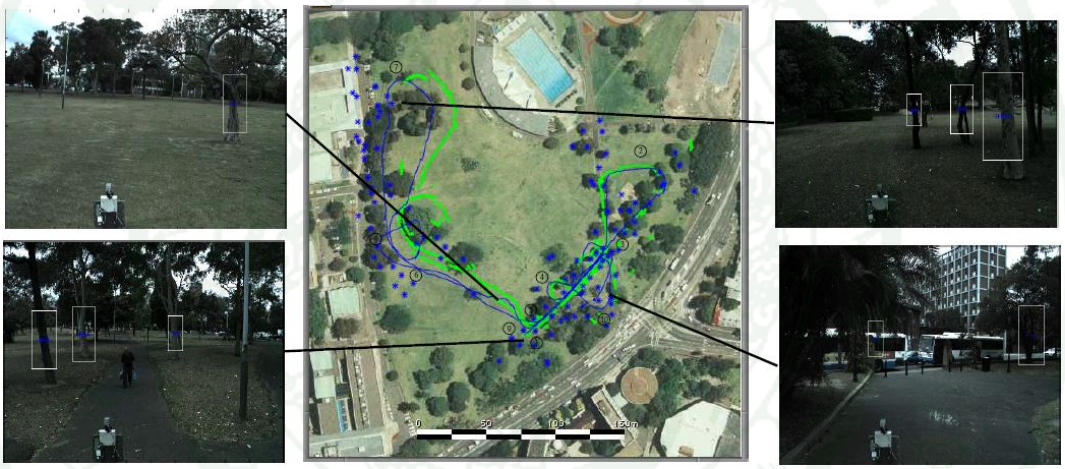


Figure 15 The result of the estimation using detected trees as feature points.

Sourec: Romos *et al.* (2007)

MATERIALS AND METHODS

This chapter is started by materials used in this study. Then, the problem statement and the purposed algorithm are presented.

Materials

Instrument

1. SICK LMS511 LiDAR
2. Color webcam
3. IR webcam
4. Checker board

Tools

1. MATLAB
2. SOPAS (supported SICK LMS511)

Library

1. Camera calibration (Zhang model)
2. RANSAC

Methods

Problem Statement

Solving extrinsic parameters, orientation and position, between a camera and a LiDAR using checkerboard. The checkerboard is placed in front of the camera and the LiDAR, and detected by those instruments for parameters estimation process. Therefore we also suggest the checkerboard detection process in LiDAR data. A constrain equation is formed from the pose of checkerboard and the pose of instruments. The configuration is defined by points, lines and planes. The parameters is estimated by solving the constraint in Least-Square sense. The constraint is the

rotation matrix and transition vector between the LiDAR and the camera that minimize the distance of an intersection line, the intersection of scanning plane and the checkerboard plane, and LiDAR scanning point. The solution contains two parts which are linear solution and non-linear solution. The result of linear solution is suffered from noise, in practice. Therefore, Non-linear solution helps to eliminate the suffering by estimating the true parameters from the answers of linear solution.

LiDAR

SICK LMS511 LiDAR is a measurement sensor that scans the surrounding perimeter in a single plane. It scans in two dimensional radial coordinates (r, θ); distance and angle. It works on the principle of time of flight. The distance is calculated using the traveling time of laser light since the LiDAR is emitted until it reflect from the object and back to the receiver. Note that, this LiDAR shows no visible trace of laser light. Figure 16 shows how the device works.

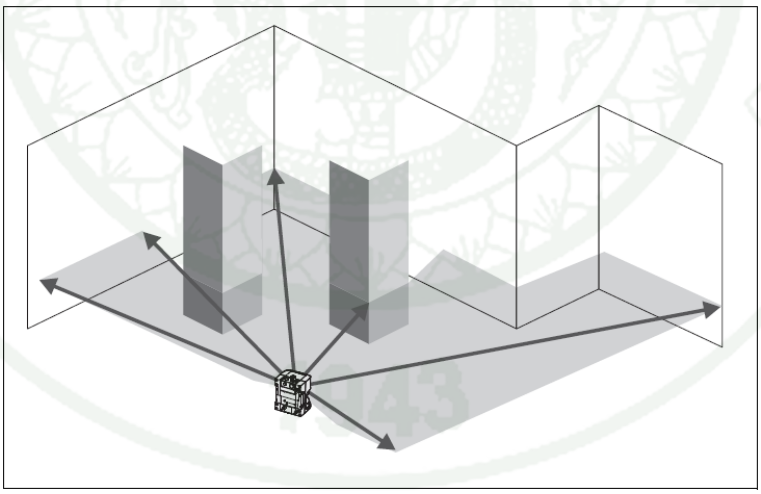


Figure 16 The functionality of a single line scan LiDAR, that is plane intersection.

Source: SICK Sensor Intelligence (n.d.)

In the experiment, we set up the device as follow: Its scans angle is from -5 degrees to 185 degrees with 0.5 degrees angle resolution, Its scan range is 1000 millimeters to 3000 millimeters. LiDAR is set to scan the checkerboard in order to calibrate the pose between a camera and a LiDAR.

Notation

In this section, we discuss about the notation and basic concepts used in this thesis. Vector is written in boldface e.g. $\boldsymbol{K}$ and matrix in typewriter font e.g. K. Extrinsic parameters are values that define the pose of an object. They usually represent in rotation matrix and translation vector. Homography helps to express those parameters to be more convenient for the calculation, the parameter estimation.

Basic Geometry Definition

We define $\boldsymbol{P} = [x \quad y \quad z]^T$ as a point in 3D and $\widehat{\boldsymbol{P}} = [x \quad y \quad z \quad 1]^T$. A 2D line is represented by $\boldsymbol{L} = [\alpha \quad \beta \quad \gamma]^T$, where α, β and γ are the parameters of line equation i.e. $0 = \alpha x + \beta y + \gamma$. A plane is represented by $\boldsymbol{\pi} = [a \quad b \quad c \quad d]^T$, where a, b, c and d are the parameters of plane equation i.e. $0 = ax + by + cz + d$. Superscript i indicates the coordinate system. For instance, $\boldsymbol{P}_i$ is the coordinate of the point $\boldsymbol{P}$ in i coordinate system.

Homography Rigid Body Transformation

Homography H is a matrix for rigid body transformation which define by orientation and position from one coordinate system to another. Orientation is presented by a rotation matrix

$$\mathrm{R}(\omega, \phi, \kappa) = \mathrm{R}_z(\kappa)\mathrm{R}_y(\phi)\mathrm{R}_x(\omega) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad , \qquad (26)$$

where

$$\mathrm{R}_x(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\omega) & -\sin(\omega) \\ 0 & \sin(\omega) & \cos(\omega) \end{bmatrix} \quad , \tag{27}$$

$$\mathrm{R}_y(\phi) = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) \\ 0 & 1 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) \end{bmatrix} \tag{28}$$

and

$$\mathrm{R}_z(\kappa) = \begin{bmatrix} \cos(\kappa) & -\sin(\kappa) & 0 \\ \sin(\kappa) & \cos(\kappa) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad . \tag{29}$$

Note that, in this the proposed method, we define the rotation in counterclockwise system, as shown in Figure 17. Position is explained by translation vector $T = [t_x \quad t_y \quad t_z]^T$. Therefore a homography can be explained by six variables which are three rotation angles: ω, ϕ, κ and three distances: t_x, t_y, t_z along x, y and z axes respectively. The 3D homography transformation can be represented by a 4 by 4 matrix: $\mathrm{H}_j^i = \begin{bmatrix} \mathrm{R}_j^i & -\mathrm{R}_j^i \boldsymbol{T}_{ji}^j \\ \mathbf{0}_3{}^T & 1 \end{bmatrix}$, where H_j^i represents the transformation from j coordinate to i coordinate systems , R_j^i rotates j coordinate to i coordinate systems, and T_{ji}^j is a vector in j coordinate system from the origin of j coordinate system to that of i coordinate system.

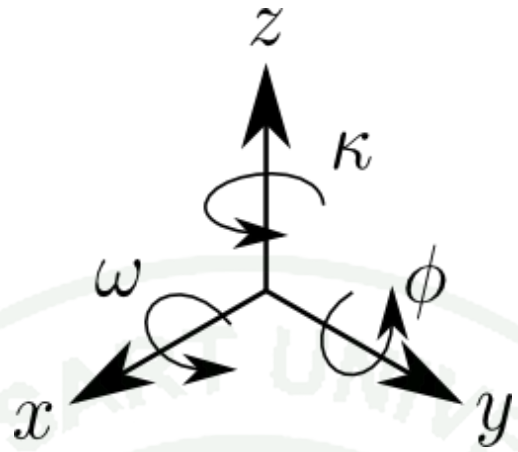


Figure 17 The definition of rotation matrix for our proposed method.

Basic Transformation

Homogeneous coordinate system is used to support a transformation of coordinate systems which have the same dimension, such 3D to 3D coordinate system. It is used along with homography transformation. Figure 18 shows the relation of homography transformation and two homogeneous coordinate systems, *i* coordinate system and *j* coordinate system.

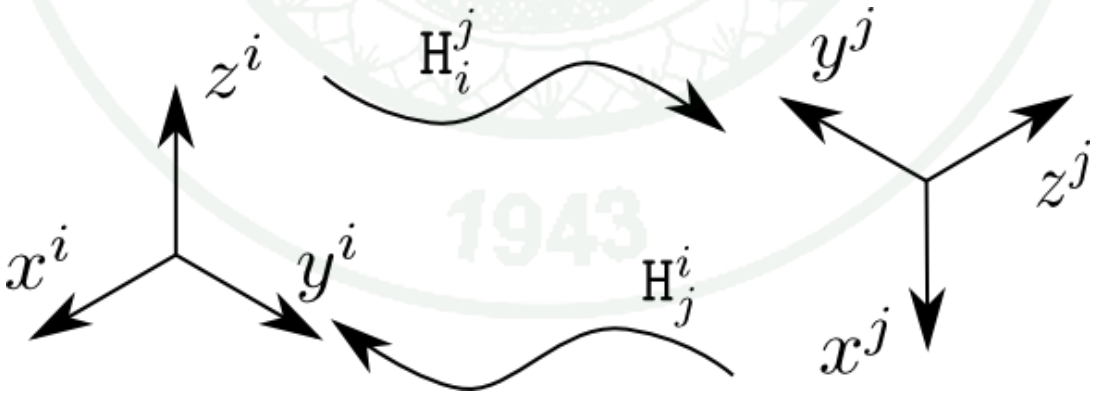


Figure 18 The transformation of *i* coordinate system and *j* coordinate system using homography.

Homography H is a transformation from one coordinate system to another. The 3D homography transformation can be represented by a 4 by 4 matrix:

$$\mathrm{H}_j^i = \begin{bmatrix} \mathrm{R}_j^i & -\mathrm{R}_j^i \boldsymbol{T}_{ji}^j \\ \mathbf{0}_3{}^{\mathrm{T}} & 1 \end{bmatrix} \quad , \tag{30}$$

where H_j^i represents the transformation from j coordinate system to i coordinate system and T_{ji}^j is a vector in j coordinate system from the origin of j coordinate system to that of i coordinate system. As a consequence, the coordinate of point $\boldsymbol{P}$ in j coordinate system can be transformed to i coordinate system by $\widehat{\boldsymbol{P}}^{\boldsymbol{i}} = \mathrm{H}_j^i \widehat{\boldsymbol{P}}^{\boldsymbol{j}}$. Although plane transformation can also uses homography. That is, the plane $\boldsymbol{\pi}$ in j coordinate system can be transformed to i coordinate system by $\boldsymbol{\pi}^{\boldsymbol{i}} = \mathrm{H}_j^{i-T} \boldsymbol{\pi}^{\boldsymbol{j}}$, where H_j^{i-T} is the inverse transpose of the homography, see Appendix A for further details of inverse transpose of homography.

Proposed Geometric Constraint

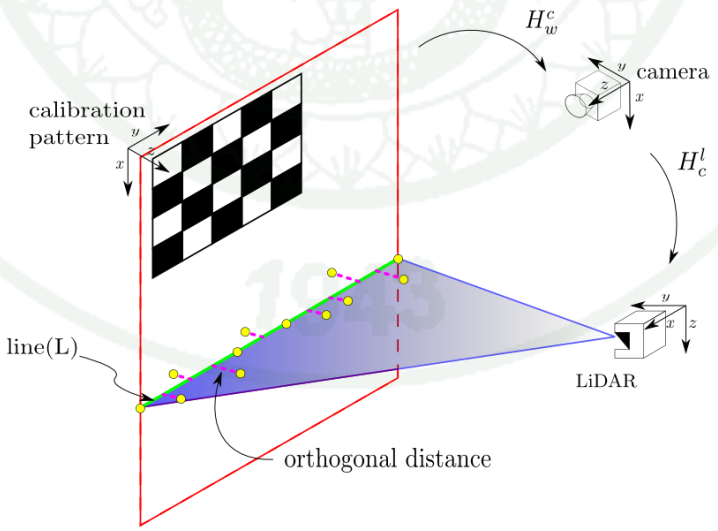


Figure 19 Our proposed geometric constraint.

A calibration plane that visible to both camera and LiDAR is used to create the geometric constraint. We assume that there is an intersection of two planes which are checkerboard plane $\boldsymbol{\pi}_{chk}$ and LiDAR scan plane $\boldsymbol{\pi}_{Li}$. Since both planes have the plane $z = 0$ in their coordinate systems, i.e. calibration plane coordinate system w and LiDAR coordinate system l, they are represented by $\boldsymbol{\pi}^{w}_{chk} = \boldsymbol{\pi}^{l}_{Li} = [0 \quad 0 \quad 1 \quad 0]^{T}$.

The LiDAR observes the checkerboard plane by sampling the intersection line as a set of points. We hypothesize that if there is no noise in the system, the orthogonal distances between LiDAR points and the intersection line of the checkerboard and LiDAR scanning plane should be minimized. This geometric constraint is illustrated in Figure 19. In order to formulate the constraint, the checkerboard plane is firstly transformed from its coordinate system to the camera coordinate system c: $\boldsymbol{\pi}^{c}_{chk} = \mathrm{H}^{c}_{w}{}^{-\mathrm{T}}\boldsymbol{\pi}^{w}_{chk}$. As a result,

$$\boldsymbol{\pi}^{c}_{chk} = \begin{bmatrix} \boldsymbol{r}^{c}_{w(3)} \\ t^{w}_{wc(3)} \end{bmatrix} \quad , \tag{31}$$

where $\boldsymbol{r}^{c}_{w(3)}$ is the third column of rotation matrix R^{c}_{w} and $t^{w}_{wc(3)}$ the third member of translation vector T^{w}_{wc}. Then transformed the checkerboard plane from the camera coordinate system to the LiDAR coordinate system by $\boldsymbol{\pi}^{l}_{chk} = \mathrm{H}^{l}_{c}{}^{-\mathrm{T}}\boldsymbol{\pi}^{c}_{chk}$. Note that R^{l}_{c} and T^{c}_{cl} are estimated. Since LiDAR scanning plane $\boldsymbol{\pi}^{l}_{Li}$ is the plane $z = 0$, the intersection line $\boldsymbol{L}^{l}$ which is the intersection between plane $\boldsymbol{\pi}^{l}_{chk}$ and $\boldsymbol{\pi}^{l}_{Li}$, can be formulated by removing the third term of $\boldsymbol{\pi}^{l}_{chk}$. Compactly, the line $\boldsymbol{L}^{l}$ is then written as:

$$\boldsymbol{L}^{l} = \boldsymbol{\pi}^{l}_{Li} \cap \boldsymbol{\pi}^{l}_{chk} = D_{1}\mathrm{H}^{l}_{c}{}^{-\mathrm{T}} \begin{bmatrix} \boldsymbol{r}^{c}_{w(3)} \\ t^{w}_{wc(3)} \end{bmatrix} \quad , \tag{32}$$

where

$$\mathrm{D}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad . \tag{33}$$

Since the LiDAR scanning plane is plane $z = 0$, we represent a laser point by $\boldsymbol{p}^l = [x \quad y]^T$ and define $\hat{\boldsymbol{p}}^l = [x \quad y \quad 1]^T$. In the ideal case, where no noise is present, the orthogonal distance of each scanning point on the intersection line $\boldsymbol{L}^l$ should be zero. As a result, the geometric constraint is then:

$$0 = \frac{\hat{\boldsymbol{p}}^{l^T} \boldsymbol{L}^l}{\|\mathrm{D}_2 \boldsymbol{L}^l\|} \quad , \tag{34}$$

where

$$\mathrm{D}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad . \tag{35}$$

Therefore, Equation (26) is used as the estimation model in this thesis.

Feature Detection

The calibration process requires co-features which is the calibration plane that visible on both camera and LiDAR. We employ the checkerboard as a calibration pattern. The checkerboard can be automatically detected in the image using image processing technique. However, the automatic detection of the checkerboard in LiDAR data is also required in order to make the fully automated feature detection. As a result, we suggest Hough transform and RANSAC, which provide line segment localization and outlier removal respectively.

As it is already discussed, The LiDAR observes the checkerboard as a line because of the intersection between the checkerboard and the LiDAR scan plane. The checkerboard is placed as the closest object to the LiDAR. Therefore, the checkerboard can be detected by locating the closest line to the origin in LiDAR coordinate system.

Hough Transform

Hough transform is the most suitable method to locate lines (Burger and Burge, 2009). The concept is to transform every point in x-y space into parameter space. In order to detect lines, we define the parameter space as ρ and θ according to line equation $\rho = x\cos(\theta) + y\sin(\theta)$. After transform every point into the parameter space, local peaks in the parameter space that correspond to line candidates in x-y space can be noticed. The closest line, the checkerboard, is defined by the minimum distance from the center of line segment to the origin. In practical, corresponding points of the line segment might contain false detection from other line segments especially when the calibration is performed indoor. In this thesis, we remove the false detection using graph cut algorithm. That is, the points that are not in the densest point cluster are removed. The algorithm below shows hough transform for detecting the closest line in the LiDAR data (set of scanned points). The applied algorithm is shown in Figure 20.

1. $HoughLines(I, N_\theta, N_r, K)$

Computes the Hough transform to detect straight lines in the 2D dataset P_i, using N_θ, N_r discrete steps for the angle and radius, respectively. Returns the list of parameter pairs θ_i, r_i_for the K strongest lines found.

2. $(M, N) \leftarrow (x_{max} - x_{min}, y_{max} - y_{min})$
3. $(u_c, v_c) \leftarrow \left(\frac{M}{2}, \frac{N}{2}\right)$ averaging point
4. $r_{max} \leftarrow \sqrt{{u_c}^2 + {v_c}^2}$ radius is half the image diagonal
5. $\Delta_\theta \leftarrow \frac{\pi}{N_\theta}$ angular increment
6. $\Delta_r \leftarrow \frac{2 \cdot r_{max}}{N_r}$ radial increment
7. Create the accumulator array $Acc(i_\theta, i_r)$ of size $N_\theta \times N_r$
8. ***for all*** accumulator cells (i_θ, i_r) ***do***
9. $Acc(i_\theta, i_r) \leftarrow 0$ initialize the accumulator array
10. ***for all*** point in dataset P_i ***do*** scan the dataset
11. $(x, y) \leftarrow (P_i(1), P_i(2))$
12. $for\ i_\theta \leftarrow 0 \cdots N_\theta - 1\ do$ angular index i_θ
13. $\theta \leftarrow \Delta_\theta \cdot i_\theta$ real angle, $0 \leq \theta < \pi$
14. $r \leftarrow x \cos(\theta) + y \sin(\theta)$ real radius (pos./neg.)
15. $i_r \leftarrow \frac{N_r}{2} + round\left(\frac{r}{\Delta_r}\right)$ radial index i_r
16. $Acc(i_\theta, i_r) \leftarrow Acc(i_\theta, i_r) + 1$ increase $Acc(i_\theta, i_r)$

Find the parameters pairs $\langle \theta_j, r_j \rangle$_ for the K strongest lines:

17. $MaxLines \leftarrow FindMaxLines(Acc, K)$
18. ***return*** $MaxLines$.

Figure 20 Pseudo code of hough transform.

RANSAC

RANSAC is an iterative method to estimate parameters of a mathematical model, and remove outliers with respect to the model: $ax + by + c = 0$. Inliers are the majority of data that fit to the model, whereas outliers do not. The model can represent as matrices: $X^{\mathrm{T}}L = 0$, where $X = [x \quad y \quad 1]^{\mathrm{T}}$ and $L = [a \quad b \quad c]^{\mathrm{T}}$. Those outliers are from the confusion of the closest object cause by imperfections of experimental setup, such as the LiDAR scan on the hand of checkerboard holder. Moreover, due to the coarse step size in Hough parameter space, the obtained points possibly have outliers. Therefore, we use RANSAC to remove all the outliers. The

pseudo-code below shows a RANSAC algorithm to filters out outliers. Figure 20 is a applied algorithm of RANSAC for fitting 2D line using a data set of points.

1. $RANSACFitLine(X_i, S, T)$

RANSAC filters out unfitted points with respect to a line model: $ax + by + c = 0$. Set of points X_i is filtered using random process. Random S points from the dataset, then fit them to the model. Any fitted point, that yield a result greater than a given threshold T, will be filtered out.

2. $trial \leftarrow 0$ initial counter
3. $score \leftarrow 0$ initial score
4. **while** $N > trail$ **do** number of trials N
5. $x_j \leftarrow rand(X_i, S)$ random S samples(normal distribution) from dataset X_i
6. $(u, d, v) \leftarrow svd(x_j)$ fitting line model using svd
7. $m \leftarrow v_{end}$ the fitted model is the last column of v matrix
8. ***for all*** x_j ***do***
9. ***if*** $T > |dist(x_j, m)|$ measure a distance of a point x_j to a line m
10. $inliers \leftarrow x_j$ every point in the threshold is inlier
11. $M \leftarrow number\ of\ inliers$ score is the number of inliers
12. ***if*** $M > score$ find the highest score
13. $bestinliers \leftarrow inliers$ mostly of inliers contain in $bestinliers$
14. $score \leftarrow number\ of\ inliers$
15. ***return*** $bestinliers$.

Figure 21 Pseudo code of RANSAC.

Parameter Estimation

Since LiDAR data is prone to noise, in practice, the derived parameters cannot satisfy Equation (34). The optimal solution can be obtained by solving Equation (34) in least-square sense. That is, we hypothesize that the optimal solution minimizes Equation (34). In order to optimize the least-square problem, we firstly solve for the initial guess for the iterative process by relaxing the non-linearity of the Equation (34). The obtained initial guess is then passed to an iterative process called Levenberg-Marquardt algorithm in order to refines the initial guess.

Linear Least-Square

Recall the experimental geometry from proposed geometric constraint section. Removed the denominator from Equation (34), then obtained linear cost function:

$$0 = \hat{\boldsymbol{p}}^{l^T} \mathrm{D}_1 \mathrm{H}_c^{l^{-\mathrm{T}}} \boldsymbol{\pi}_{chk}^c \quad , \tag{36}$$

where

$$\mathrm{H}_c^{l^{-T}} = \begin{bmatrix} \mathrm{R}_c^l & \mathbf{0}_3 \\ \boldsymbol{T}_{cl}^c & 1 \end{bmatrix} \tag{37}$$

and

$$\boldsymbol{L}^l = D_1 \mathrm{H}_c^{l^{-\mathrm{T}}} \begin{bmatrix} \boldsymbol{r}_{w(3)}^c \\ t_{wc(3)}^w \end{bmatrix} \quad . \tag{38}$$

By apply $\boldsymbol{vec}$ operator to Equation (36), see Appendix A for further details of $\boldsymbol{vec}$ operator and Kronecker product, we obtain:

$$0 = \left(\left(\boldsymbol{\pi}_{chk}^{c^T} \right) \otimes \left(\hat{\boldsymbol{p}}^{l^T} \mathrm{D}_1 \right) \right) \boldsymbol{vec} \left(\mathrm{H}_c^{l^{-T}} \right) \quad , \tag{39}$$

where $\otimes$ is Kronecker product operator. Let $\boldsymbol{A} = \left(\boldsymbol{\pi}_{chk}^{c^T} \right) \otimes \left(\hat{\boldsymbol{p}}^{l^T} \mathrm{D}_1 \right)$ which have the size of $\left(\sum_{k=0}^{j} i_k \right) \times (m \times n)$, where i is the number of scan points in one pose, and j is the number of poses. Since $\mathrm{H}_c^{l^{-T}}$ have a size of 4×4, $\boldsymbol{vec} \left(\mathrm{H}_c^{l^{-T}} \right)$ is then expresses as follow:

$$\boldsymbol{vec}\left(\mathrm{H}_c^{l\,-T}\right) = \begin{bmatrix} r_{11} \\ r_{21} \\ r_{31} \\ t_x \\ \vdots \\ r_{13} \\ r_{23} \\ r_{33} \\ t_z \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad , \tag{40}$$

which have the size of 16×1. Then, Equation (36) can be rewrite as:

$$0 = \boldsymbol{A vec}\left(\mathrm{H}_c^{l\,-T}\right) \quad . \tag{41}$$

Term $\boldsymbol{vec}\left(\mathrm{H}_c^{l\,-T}\right)$ can reduce into since 0 and 1 are constant values in the matrix

$$\boldsymbol{vec}\left(\left[\mathrm{R}_c^{l\,\mathrm{T}} \middle| \boldsymbol{T}_{cl}^{c}\right]^{\mathbf{T}}\right) = \begin{bmatrix} r_{11} \\ r_{21} \\ r_{31} \\ t_x \\ \vdots \\ r_{13} \\ r_{23} \\ r_{33} \\ t_z \end{bmatrix} \quad . \tag{42}$$

As a consequence, the constant values are then multiple with matrix $\boldsymbol{A}$. Then, Equation (41) is reformulated as:

$$\boldsymbol{vec}\left(\left[\mathrm{R}_c^{l\,\mathrm{T}} \middle| \boldsymbol{T}_{cl}^{c}\right]^{\mathbf{T}}\right) = -[\boldsymbol{A}_1|\boldsymbol{A}_2|\cdots|\boldsymbol{A}_{12}]^{-1}\boldsymbol{A}_{16} \quad , \tag{43}$$

where $\boldsymbol{A}_i$ is the $i-$th column of $\boldsymbol{A}$. After solving Equation (43) in least-square sense using pseudo-inverse, see Appendix A for pseudo-inverse, reordered

$\boldsymbol{vec}\left(\left[\mathrm{R}_c^{l\,\mathrm{T}}\middle|\boldsymbol{T}_{cl}^{c}\right]^{\mathbf{T}}\right)$ into R_c^l matrix and $\boldsymbol{T}_{cl}^{c}$ vector. Then resolved R_c^l according to rotation matrix properties which will be described in the next paragraph. Therefore, we obtain extrinsic parameters, that are R_c^l and $\boldsymbol{T}_{cl}^{c}$ that suit to the proposed geometry.

The obtained rotation matrix may not satisfy rotation matrix properties which are $\mathrm{R}_c^{l\,T} = \mathrm{R}_c^{l\,-1}$ and $\det(\mathrm{R}_c^l) = 1$. To obtain the valid rotation matrix that satisfies the aforementioned properties, using Singular Value Decomposition (SVD), $\mathrm{R}_c^l = \mathrm{UDV}^{\mathrm{T}}$, see further information about SVD and rotation matrix in Appendix A, and retrieve the true rotation matrix by applied the following constraint: $\mathrm{R}_c^l = \mathrm{USV}^{\mathrm{T}}$, where $\mathrm{S} = diag(1,1,\det(\mathrm{U})\det(\mathrm{V}^{\mathrm{T}}))$, according to (Moakher, 2002).

Non-Linear Optimization

Non-linear optimization requires an initial guess as an original searching area, which come from linear least-square Subsection. In other words, the initial solution is refined by non-linear least-square to find the most optimizing answer to the equation. The translation is simply parameterized by vector without imposing any constraint. Unfortunately, the properties of rotation matrix need to be imposed as constraints in the estimation model in order to obtain the correct rotation matrix. To waive the rotation matrix constraints, Triggs (Triggs *et al*., 2000) mentioned that rotation matrix should be parameterized using local perturbation $\mathrm{R} = \mathrm{R}_0\delta\mathrm{R}$ where R_0 is the original rotation matrix's state, $\delta\mathrm{R} = \exp([[\omega \quad \phi \quad \kappa]^T]_\times)$ the state update and $[[\omega \quad \phi \quad \kappa]^T]_\times$ a skew matrix of three rotation angles. Particularly, the rotation matrix is approximated about zero rotation angle $\omega = \phi = \kappa = 0$ to locate the optimum point around the original location. Additionally, we use Levenberg-Marquardt to increase iteration speed.

This process minimizes the objective function $f(v) = \frac{\hat{\boldsymbol{p}}^{l\,\mathrm{T}}\boldsymbol{L}^l}{\|\mathrm{D}_2\boldsymbol{L}^l\|}$ by refine the variable $v = [\omega, \phi, \kappa, t_x, t_y, t_z]$ which is literally R_c^l and $\boldsymbol{T}_{cl}^{c}$, until the objective

function yield an acceptable tolerance. To refine it, Equation (34) is expanded to first-order of Taylor's series (Gauss-Newton):

$$0 = f(v) + f'(v)dv \quad , \tag{44}$$

then dv is required to update the estimated parameter. It can be found by solve the following equation:

$$dv = -f'(v)^{-1}f(v) \quad . \tag{45}$$

First derivative of $f(v)$ is shown below:

$$\frac{df(v)}{dv} = \frac{d\hat{\boldsymbol{p}}^{l\mathrm{T}}\boldsymbol{L}^l}{dv\|\mathrm{D}_2\boldsymbol{L}^l\|} \quad , \tag{46}$$

by applied chain rule technique the first derivative is then reformulated as:

$$\frac{df(v)}{dv} = \frac{df(v)}{d\boldsymbol{L}^l} \cdot \frac{d\boldsymbol{L}^l}{dv} \quad . \tag{47}$$

$\frac{df(v)}{d\boldsymbol{L}^l}$ is resolved using quotient rule of derivative and shown as follow:

$$\frac{df(v)}{dv} = \frac{\|\mathrm{D}_2\boldsymbol{L}^l\| \frac{d\hat{\boldsymbol{p}}^{l\mathrm{T}}\boldsymbol{L}^l}{d\boldsymbol{L}^l} - \hat{\boldsymbol{p}}^{l\mathrm{T}}\boldsymbol{L}^l \frac{d\|\mathrm{D}_2\boldsymbol{L}^l\|}{d\boldsymbol{L}^l}}{\|\mathrm{D}_2\boldsymbol{L}^l\|^2} \cdot \frac{d\boldsymbol{L}^l}{dv} \quad . \tag{48}$$

$\frac{d\hat{\boldsymbol{p}}^{l\mathrm{T}}\boldsymbol{L}^l}{d\boldsymbol{L}^l}$ and $\frac{d\|\mathrm{D}_2\boldsymbol{L}^l\|}{d\boldsymbol{L}^l}$ are derived as:

$$\frac{df(v)}{dv} = \frac{\|\mathrm{D}_2\boldsymbol{L}^l\|\hat{\boldsymbol{p}}^{l\mathrm{T}} - \hat{\boldsymbol{p}}^{l\mathrm{T}}\boldsymbol{L}^l(\mathrm{D}_2\boldsymbol{L}^l)^{\mathrm{T}}\mathrm{D}_2}{\|\mathrm{D}_2\boldsymbol{L}^l\|^2} \cdot \frac{d\boldsymbol{L}^l}{dv} \quad . \tag{49}$$

Then, reformed Equation (49) in simpler order as:

$$\frac{df(v)}{dv} = \frac{\widehat{\boldsymbol{p}}^{l\mathrm{T}}}{\|\mathrm{D}_2\boldsymbol{L}^l\|} - \frac{\widehat{\boldsymbol{p}}^{l\mathrm{T}}\boldsymbol{L}^l\boldsymbol{L}^{l\mathbf{T}}\mathrm{D}_2{}^{\mathbf{T}}\mathrm{D}_2}{\|\mathrm{D}_2\boldsymbol{L}^l\|^2} \cdot \frac{d\boldsymbol{L}^l}{dv} \quad . \tag{50}$$

Finally, we obtain first derivative of $f(v)$: $f'(v)$, expressed in the following form:

$$\frac{df(v)}{dv} = \frac{\widehat{\boldsymbol{p}}^{l\mathrm{T}}}{\|\mathrm{D}_2\boldsymbol{L}^l\|}\left(\mathrm{I}_3 - \frac{\boldsymbol{L}^l\boldsymbol{L}^{l\mathbf{T}}\mathrm{D}_2}{\|\mathrm{D}_2\boldsymbol{L}^l\|^2}\right) \cdot \frac{d\boldsymbol{L}^l}{dv} \quad . \tag{51}$$

Derivative of matrix can be calculated using Jacobian matrix. Therefore $\frac{d\boldsymbol{L}^l}{dv}$ can be calculate using Jacobian and obtain:

$$\begin{aligned}\frac{d\boldsymbol{L}^l}{dv} = \Bigg[& D_1\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta\omega}\boldsymbol{\pi}_{chk}^c \quad D_1\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta\phi}\boldsymbol{\pi}_{chk}^c \quad D_1\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta\kappa}\boldsymbol{\pi}_{chk}^c \quad \cdots \\ & D_1\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta t_x}\boldsymbol{\pi}_{chk}^c \quad D_1\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta t_y}\boldsymbol{\pi}_{chk}^c \quad D_1\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta t_z}\boldsymbol{\pi}_{chk}^c\Bigg]\end{aligned} \tag{52}$$

since L^l is explained in Equation (32). By directly take the partial derivative inside $\mathrm{H}_c^{l\,-\mathrm{T}}$, its partial derivatives is then shown below:

$$\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta\omega} = \begin{bmatrix}\mathbf{0_3} & (\mathbf{r}_c^l)_3 & -(\mathbf{r}_c^l)_2 & \mathbf{0_3}\\ 0 & 0 & 0 & 0\end{bmatrix} \quad , \tag{53}$$

$$\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta\phi} = \begin{bmatrix}-(\mathbf{r}_c^l)_3 & \mathbf{0_3} & (\mathbf{r}_c^l)_1 & \mathbf{0_3}\\ 0 & 0 & 0 & 0\end{bmatrix} \quad , \tag{54}$$

$$\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta\kappa} = \begin{bmatrix}(\mathbf{r}_c^l)_2 & -(\mathbf{r}_c^l)_1 & \mathbf{0_3} & \mathbf{0_3}\\ 0 & 0 & 0 & 0\end{bmatrix} \quad , \tag{55}$$

$$\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta t_x} = \begin{bmatrix}\mathbf{0_3} & \mathbf{0_3} & \mathbf{0_3} & \mathbf{0_3}\\ 1 & 0 & 0 & 0\end{bmatrix} \quad , \tag{56}$$

$$\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta t_y} = \begin{bmatrix}\mathbf{0_3} & \mathbf{0_3} & \mathbf{0_3} & \mathbf{0_3}\\ 0 & 1 & 0 & 0\end{bmatrix} \quad , \tag{57}$$

$$\frac{\delta \mathrm{H}_c^{l\,-\mathrm{T}}}{\delta t_z} = \begin{bmatrix}\mathbf{0_3} & \mathbf{0_3} & \mathbf{0_3} & \mathbf{0_3}\\ 0 & 0 & 1 & 0\end{bmatrix} \quad . \tag{58}$$

Note that, the rotation matrix is given by exponential of skew matrix of three rotation angles multiple with R_0 as:

$$\begin{aligned} \mathrm{R}_c^l(\omega,\phi,\kappa) &= \mathrm{R}_0 exp\left(\begin{bmatrix}\omega\\ \phi\\ \kappa\end{bmatrix}\right) \\ &= \mathrm{R}_0 exp\left(\begin{bmatrix} 0 & -\kappa & \phi \\ \kappa & 0 & -\omega \\ -\phi & \omega & 0 \end{bmatrix}\right) \end{aligned} \quad , \qquad (59)$$

R_0 is an initial guess of rotation matrix which come from solving Equation (43) in least-square sense, which is the rotation matrix between the camera and the LiDAR. In the other hands, the position between them is basically defined as:

$$\boldsymbol{T}_{cl}^c(t_x,t_y,t_z) = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad , \qquad (60)$$

since there are no imposed constraint to the translation vector. The objective function and its first derivative are applied in Levenberg-Marquardt, see Appendix A for Levenberg-Marquardt optimization process, and then refined the result of R_c^l and $\boldsymbol{T}_{cl}^c$ to be optimized.

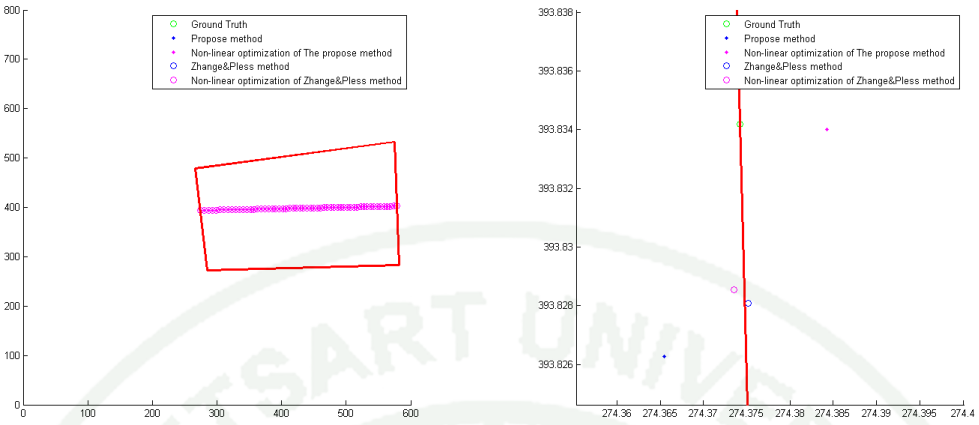
RESULTS AND DISCUSSION

Resules

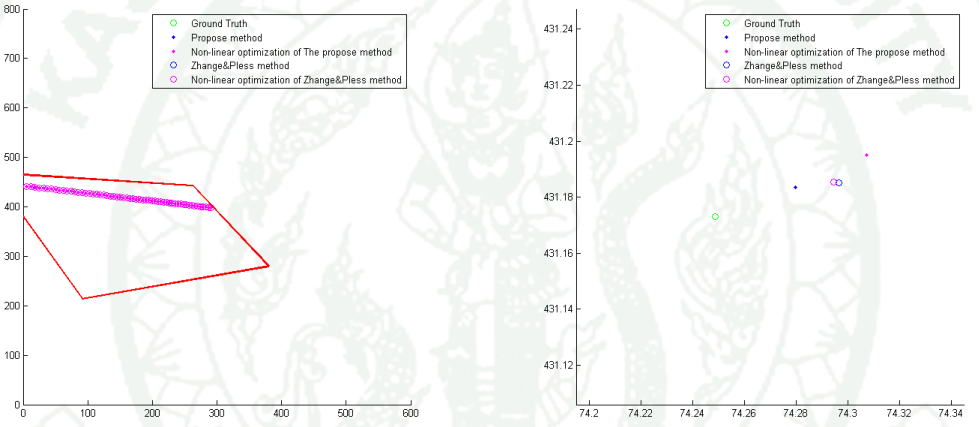
This chapter presents the experimental results. It contains two estimation result; result from simulated data and real data. The simulation is created by given the poses of the checkerboard randomly as well as the pose of the LiDAR and the camera. The checkerboard size is also given the same as the practice experiment. The experiment conducts four trails using real data. Each trail have different LiDAR-camera pose and more than 20 checkerboard poses in order to estimate the LiDAR-camera pose. Note that the illustrated results are some of all experimental results.

Simulated Data

The simulation is repeated 100 times with random process of additive Gaussian noise on LiDAR data. The estimation errors are obtained by averaging all trails. The comparisons show the displacement distance to the ground truth data, see Figure 22. The displacement for the rotation matrix is Riemanian distance, measuring the rotation matrix in manifold manner. In the other hands, the translation vector also compare with the ground truth using the displacement distance which is Euclidien distance, see Appendix A for further information of Riemanian and Euclidien Distance.



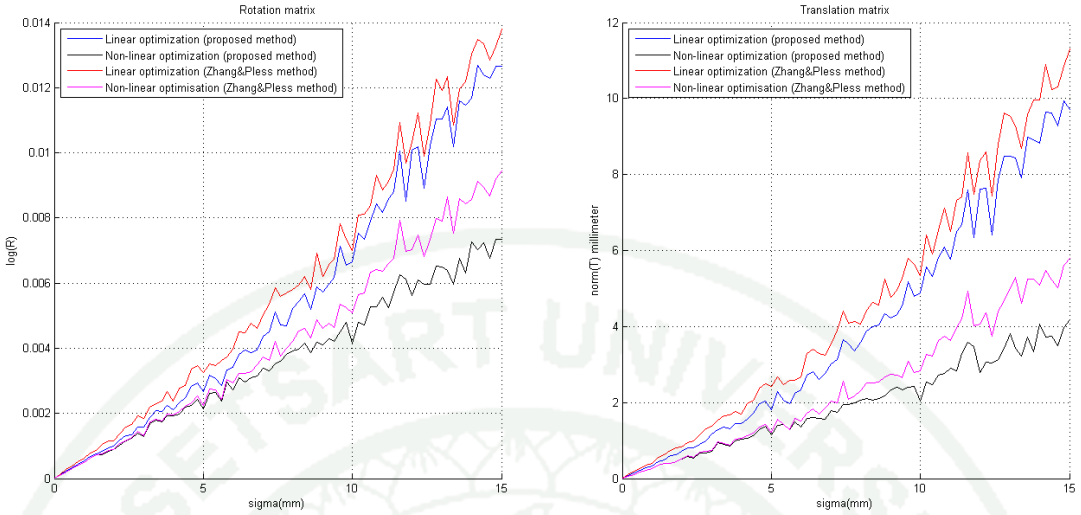
(a-1) One of simulated experiments (a-2) Zoom-in of Figure (a-1)



(b-1) One of simulated experiments (b-2) Zoom-in of Figure (b-1)

Figure 22 Samples of simulated experiments (reprojected data).

Performance with respect to noise standard deviation is presented by the displacement of rotation matrix and translation vector to the ground truth, which are shown in Figure 23-(a) and Figure 23-(b) respectively. In the experiment with respect to noise standard deviation, we used 28 poses of camera to checkerboard, and varied noise standard deviation from 0-15 millimeters. They show that the estimation errors from linear estimation grow rapidly with respect to noise standard deviations. In contrast, the errors from non-linear estimation grow linearly. The proposed algorithm outperforms Zhang and Pless method (Zhang, 2000) in both linear and non-linear estimation.

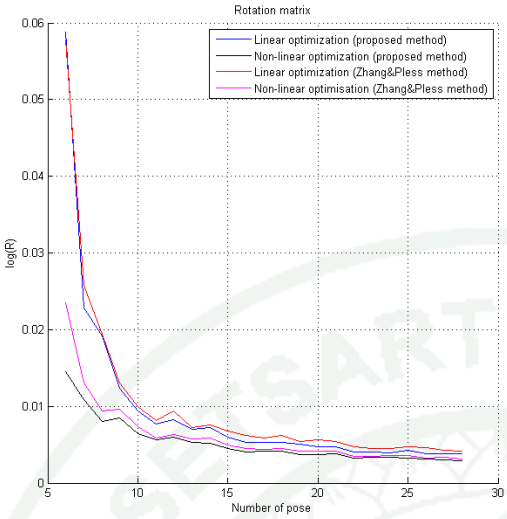


(a) Rotation matrix estimation error vs. noise standard deviation

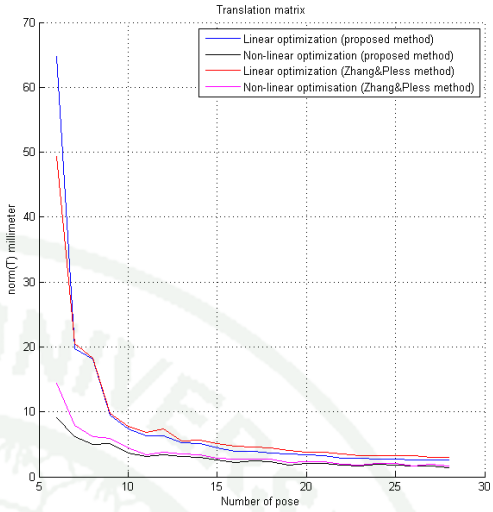
(b) Translation vector estimation error vs. noise standard deviation

Figure 23 Rotation matrix and translation vector estimated error with respect to noise standard deviation.

Performance with respect to the number of camera to checkerboard poses are presented in Figure 24, Figure 25 and Figure 26. In the experiments, noise standard deviations are 6, 15 and 30 millimeters for Figure 24, Figure 25 and Figure 26 respectively. In practice, six standard deviation is given from LiDAR user manual for less than five meters operation range. 15 and 30 millimeters standard deviation simulate more severe case of LiDAR scan to compare our estimation efficiency in different situations. The number of camera poses are varied from 6-28 poses. The estimation errors reduce drastically when the number of poses increases. In other words, the higher number of poses we have, the better estimation result it is.

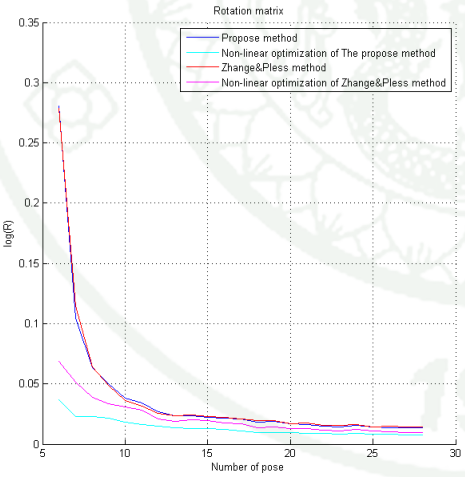


(a) Rotation matrix estimation error vs. number of poses

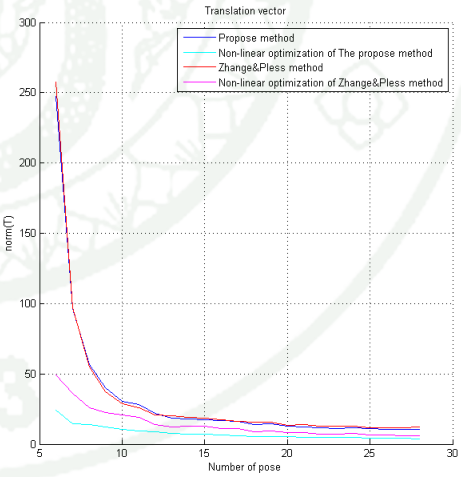


(b) Translation vector estimation error vs. number of poses

Figure 24 Rotation matrix and translation vector estimated error with respect to the number of checkerboard poses (SD: 6 millimeters).

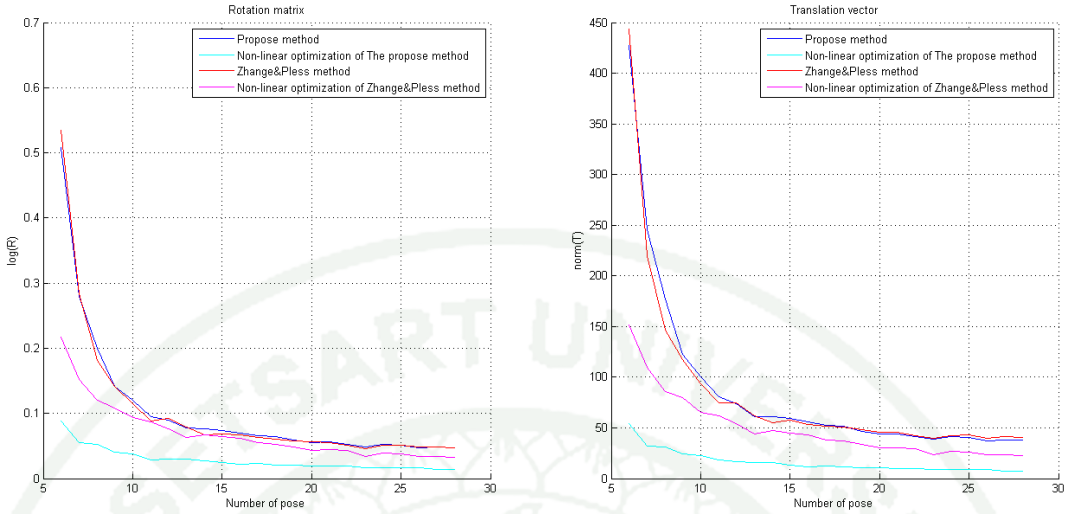


(c) Rotation matrix estimation error vs. number of poses



(d) Translation vector estimation error vs. number of poses

Figure 25 Rotation matrix and translation vector estimated error with respect to the number of checkerboard poses (SD: 15 millimeters).



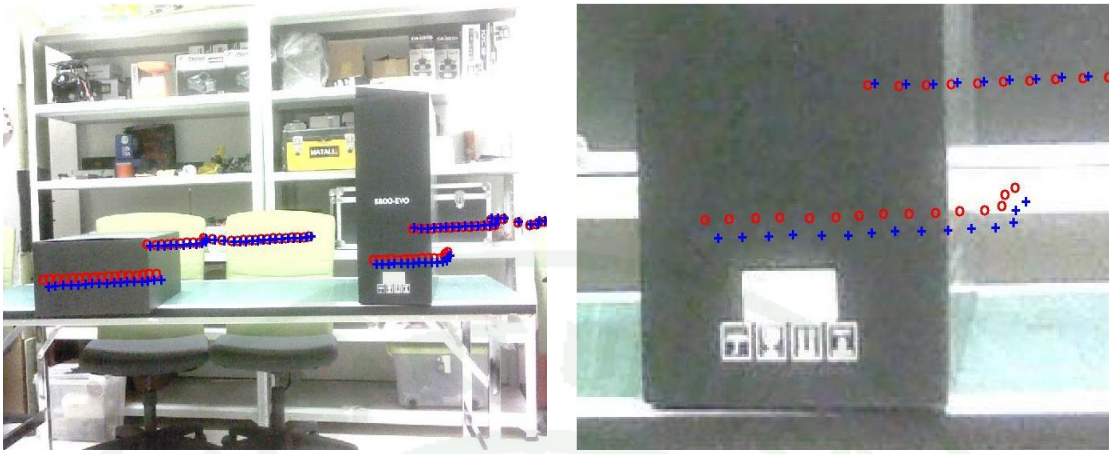
(e) Rotation matrix estimation error vs. number of poses

(f) Translation vector estimation error vs. number of poses

Figure 26 Rotation matrix and translation vector estimated error with respect to the number of checkerboard poses (SD: 30 millimeters).

Real Data

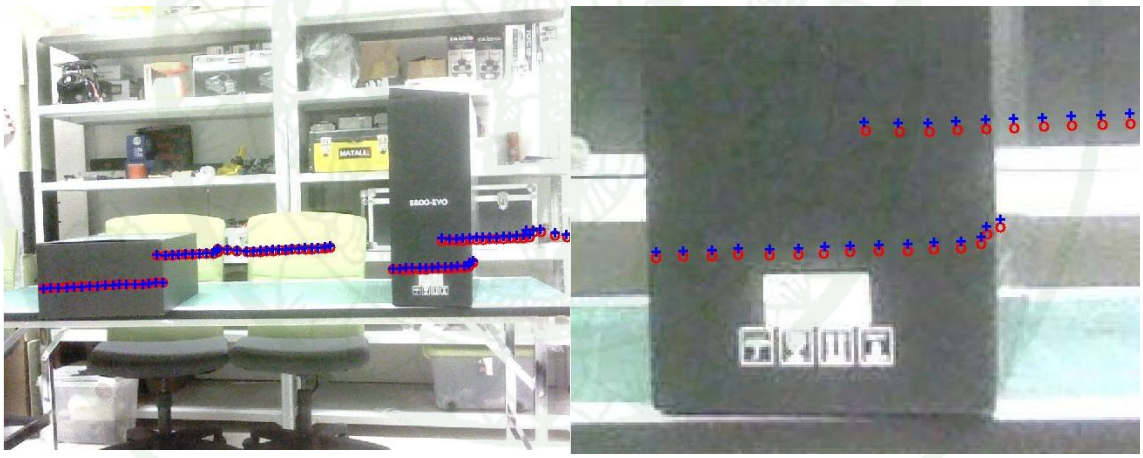
The results from real data show that our proposed method is well performed and yield reliable calibration parameters. Figure 27 and Figure 28 show projected LiDAR data on the images using the parameters from linear and non-linear estimation. They can be observed that the reprojected points using the parameters from linear estimation do not match with the corner of the black box as shown in Figure 37-(a). Then after non-linear optimization as shown in Figure 38-(b), the reprojected points match with the corner of black box more properly.



(a) Reprojected LiDAR points onto image using linear optimization

(b) Close-up of reprojected LiDAR points onto image using linear optimization

Figure 27 LiDAR data reprojection using parameters from linear solution.



(a) Reprojected LiDAR points onto image using non-linear optimization

(b) Close-up of reprojected LiDAR points onto image using non-linear optimization

Figure 28 LiDAR data reprojection using parameters from non-linear optimization.

Discussion

There are two type of experiments conducted in this thesis. Simulated experiment establish the studies the model efficiency with controlled environment. Practice experiment is applied to prove that our model can be practically use.

We compare our proposed (Tulsuk *et al*., 2014) method with that of Zhang and Pless (Zhang & Pless, 2004) using simulated and real data. In the experiment with simulated data, the comparison is performed using displacement of derived rotation matrix and translation vector to their ground truth. The estimated parameters are illustrated in terms of image with reprojected data, in the experiment using real data.

Simulated Experiment

To measure the estimation errors, the estimated parameters are then compared with the ground truth. We established a simulation of our experimental setup to analyze the estimated parameter errors of the proposed method and that of Zhang and Pless method (Zhang and Pless, 2004).

The checkerboard size and extrinsic parameters between the camera and the LiDAR are given. Sets of LiDAR points are generated by vary the pose of camera to checkerboard. The checkerboard size is 500×700 square millimeters. The LiDAR scans along the checkerboard with 0.5 degrees angular resolution without scattering laser light. The orientations of the camera with respect to checkerboard coordinate system are randomly generated such that orthonormal property of rotation matrix is satisfied while the distances from the camera to the checkerboard are limited to 2500 millimeters.

The simulation is established as Figure 29. The process including mainly three parts, i.e. given experimental poses and checkerboard size, simulating LiDAR data for the situation of scanning the calibration pattern and analysis with respect to

noise power added in LiDAR data. The outcome of the simulation is the study of noise effect to the estimation model.

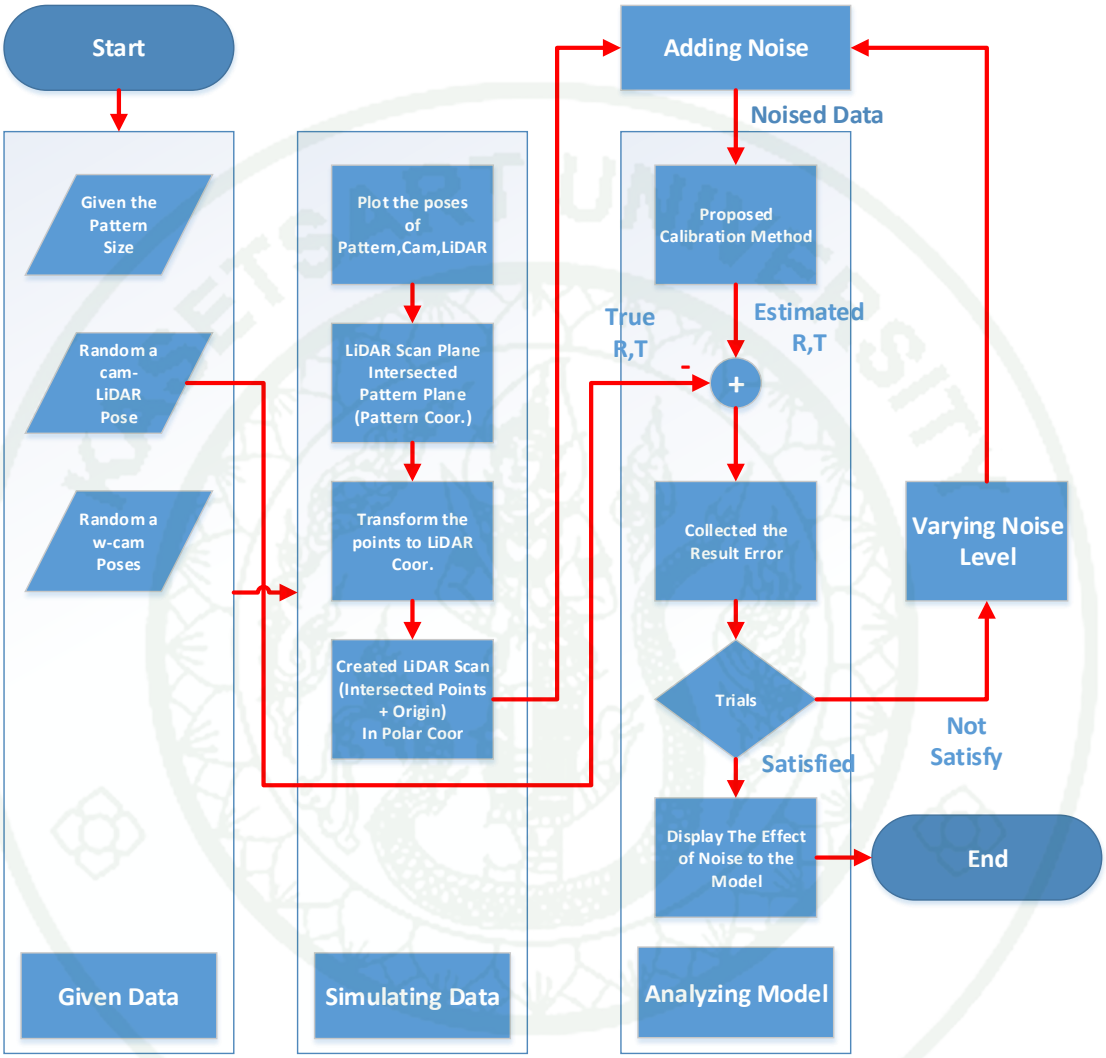


Figure 29 Flow chart of simulation experiment.

Practical Experiment

The proposed method is tested using a computer webcam camera, a modified IR webcam camera (only for observation) and SICK LMS 511(LiDAR). All devices are mounted on a fixed plate. Then they capture various poses of checkerboard such

that the calibration plane is visible to both camera and LiDAR. Its scanning angle is between -5 to 185 degrees with 0.5 degrees angular resolution. Standard deviation on measurement range is 7 millimeters for less than 10 meters operation range (SICK Sensor Intelligence, n.d.). Color camera resolution is set to 1024×768. The calibration pattern consists of 7 squares $\times$ 9 squares and the size of each square is 57×57 square millimeters. The camera and the LiDAR observed the checkerboard simultaneously. We acquired more than 20 images and LiDAR scans for estimating the pose between the camera and the LiDAR. Moreover, those images are used for the camera calibration as well in order to obtain R_W^C and $\boldsymbol{T}_{CW}^C$. The feature points, the set of points that lies on the checkerboard, are automatically picked by feature extraction process mentioned in feature detection section.

As shown in Figure 30, the estimation process contains mainly three parts which are pre-processing, calibration and display. Pre-processing prepares data for the calibration. The feature, checkerboard poses (from camera) and checkerboard intersected line (from LiDAR data), must be extracted. The checkerboard poses are observed by LiDAR as the closest lines in for each pose according to the aforementioned experimental setup in feature detection section. Line detection is applied to locate the points on the plane, i.e. hough transform and RANSAC. While the camera detects the checkerboard poses using corner detection and apply Zhang camera calibration (Zhang, 2000) to find the pose of the camera with respect to the checkerboard. Two inputs are used for our proposed LiDAR calibration model, and then estimate the pose between the camera and the LiDAR. The derived pose are verified using the projection of LiDAR data onto images.

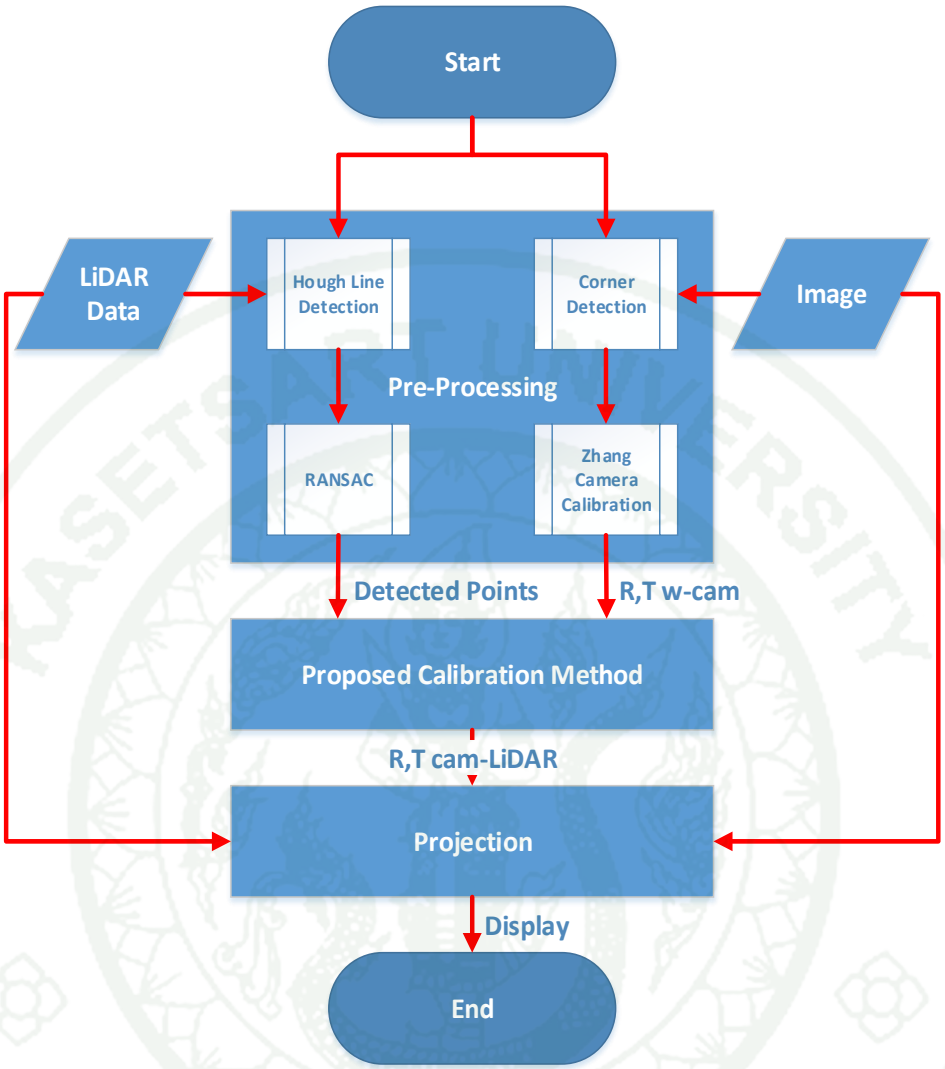


Figure 30 Flow chart of the experiment using real data.

CONCLUSION AND RECOMMENDATION

Conclusion

This thesis proposes a novel estimation model for estimating extrinsic parameters between a camera and a single line scan LiDAR. We constructed new geometric constraint that based on minimizing orthogonal distances of LiDAR points and the intersection line between the checkerboard and scanning plane.

Not only linear least-square solution but also non-linear optimization solution of our proposed method that yield more efficiency, show in the simulation. Thus, they show that our proposed (Tulsuk *et al*., 2014*)* model outperforms that of Zhang and Pless model (Zhang & Pless, 2004). We proposed the constraint without loss of information; that is the reason that ours is superior. Moreover, our plane transformation reduces uncertainties of their point transformation. Our optimization step leads to the appropriate optimum point because we included orthonormal property of rotation matrix into the procedure. Like many methods, the efficiency of algorithm depends on the number of checkerboard poses that is visible to the camera and the LiDAR.

Recommendation

The proposed method shows that it can estimate the extrinsic parameters of LiDAR reference to camera effectively. Also, it performs in the real situation property. Moreover, this method can solve the extrinsic parameters of planar-based LiDAR such as velodyne LiDAR or multi-planar LiDAR. The estimation process can be more accurate by applied probability model to handle the uncertainty in practical situation. This method helps and accommodates the task such as 3D position acquisition and reconstruction using LiDAR and camera.

LITERATURE CITED

Burger, W. and M. J. Burge. 2009. **Principles of Digital Image Processing**. Core Algorithms. Springer-Verlag London Limited.

Davis, C. H. 2003. A combined fuzzy pixel-based and object-based approach for classification of high-resolution multispectral data over urban areas. *In* **IEEE Trans. Geosci. Remote Sens.** 41 (10): 2354–2363.

Douillard, B., D. Fox and F. Ramos. 2008. **Laser and Vision Based Outdoor Object Mapping**. Available Source: http://dblp.uni-trier.de/db/conf/rss/rss2008.html\#DouillardFR08, March 15, 2014.

Fruh, C. and A. Zakhor. 2004. An automated method for large-scale, ground-based city model acquisition. *In* **International Journal of Computer Vision** *60*: 5-24.

Huang, L. and M. Barth. 2009. A novel multi-planar LIDAR and computer vision calibration procedure using 2D patterns for automated navigation, pp. 117-122 *In* **IEEE Intelligent Vehicles Symposium, 2009**.

Kovesi, P. n.d. **Computer Vision and Image Processing**. Available Source: http://www.peterkovesi.com/, April 20, 2014.

Kwak, K., D. Huber, H. Badino and T. Kanade. 2011. Extrinsic calibration of a single line scanning lidar and a camera, pp. 3283-3289. *In* **IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**.

Li, C., C. Xu and C. Gui. 2005. Level set evolution without re-initialization: A new variational formulation. *In* **Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, vol. 1, pp 430-436.

Li, G., Y. Liu, L. Dong, X. Cai and D. Zhou. 2007. An algorithm for extrinsic parameters calibration of a camera and a laser range finder using line features, pp. 3854-3859 *In* **IEEE/RSJ International Conference on Intelligent Robots and Systems.**

Moakher, M. 2002. **Means and Averaging in the Group of Rotations**.

Premebida, C., G. Monteiro, U. Nunes and P. Peixoto. 2007. A Lidar and Vision-based Approach for Pedestrian and Vehicle Detection and Tracking, pp. 1044-1049. *In* **IEEE Conference on Intelligent Transportation Systems (ITSC 2007).**

Ramos, F., T., J. Nieto and H. F. Durrant-Whyte. 2007. Recognising and Modelling Landmarks to Close Loops in Outdoor SLAM, pp. 2036-2041. *In* **IEEE International Conference on Robotics and Automation.**

SICK Sensor Intelligence. n.d. **Laser scanners LMS5xx / LMS511 / Outdoor / Mid Range, LMS511-20100 PRO**. Available Source: http://www.mysick.com, April 20, 2014.

Triggs, B., P. Mclauchlan, R. Hartley and A. Fitzgibbon. 2000. Bundle adjustment – a modern synthesis, pp. 298-375. **Vision Algorithms: Theory and Practice, LNCS**. Springer Verlag.

Tulsuk, P., P. Srestasathiern, M. Ruchanurucks, T. Phatrapornnant and H. Nagahashi. 2014. A Novel Method for Extrinsic Parameters Estimation between a Single-Line Scan LiDAR and a Camera. **THE INTELLIGENT VEHICLES SYMPOSIUM (IV'14).**

Vasconcelos, F., J. Barreto and U. Nunes. 2012. A Minimal Solution for the Extrinsic Calibration of a Camera and a Laser-Rangefinder. *In* **IEEE Transactions on Pattern Analysis and Machine Intelligence**. *34*: 2097-2107.

Wasielewski, S. and O. Strauss. 1995. Calibration of a multi-sensor system laser rangefinder/camera, pp. 472-477. *In* **Proceedings of Intelligent Vehicles '95 Symposium.**

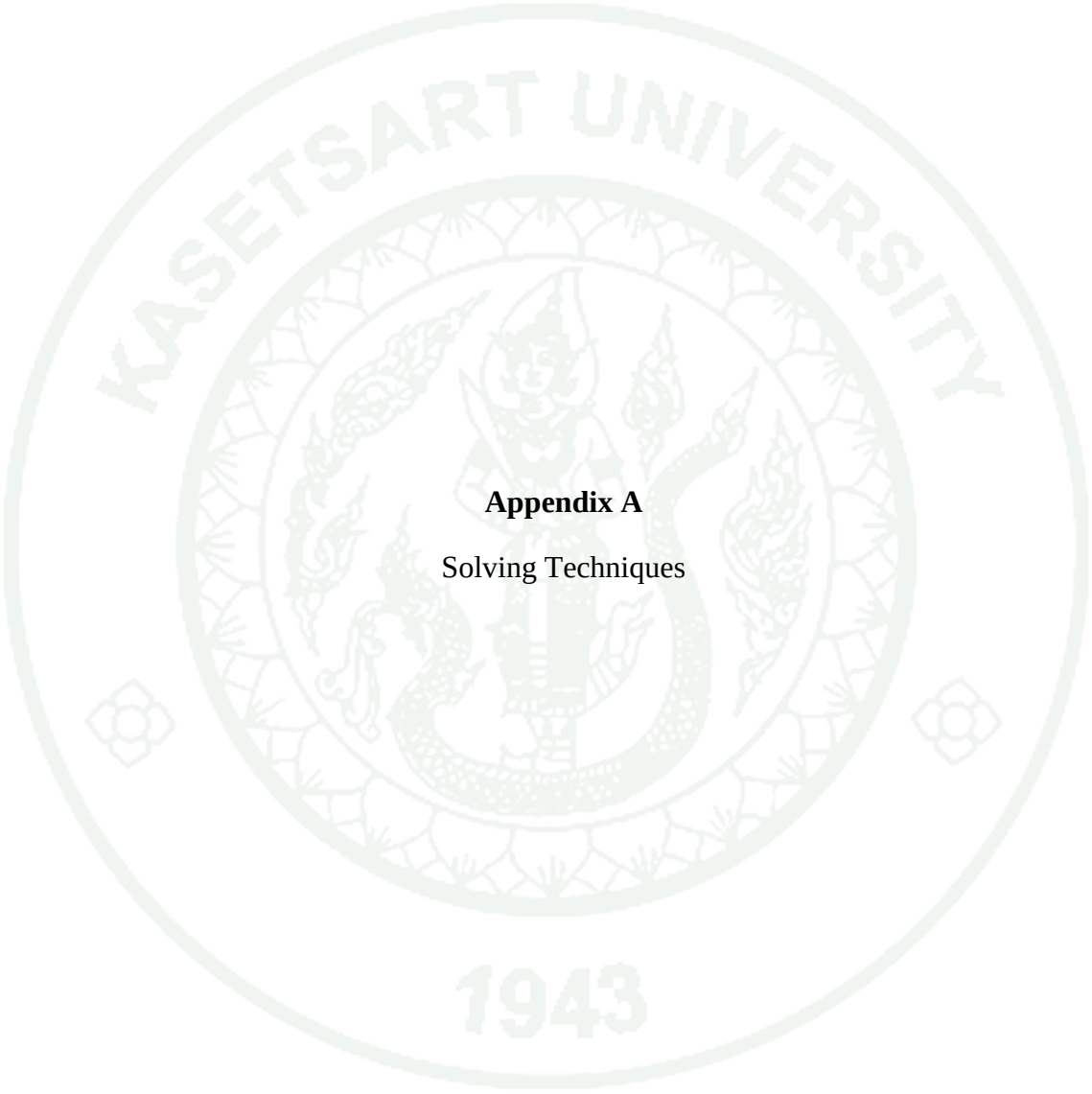
Zhang, Q. and R. Pless. 2004. Extrinsic calibration of a camera and laser range finder (improves camera calibration). *In* **Proceedings of *I*EEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)**. 3: 2301-2306.

Zhang, Z. 2000. A flexible new technique for camera calibration. *In* **IEEE Transactions on Pattern Analysis and Machine Intelligence**. 22: 1330-1334.

Zhou, L. and Z. Deng. 2012. A new algorithm for computing the projection matrix between a LIDAR and a camera based on line correspondences, pp. 436-441. *In* **International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT2012)**.

Zhou, L. and Z. Deng. 2012. Extrinsic calibration of a camera and a lidar based on decoupling the rotation from the translation, pp. 642-648. *In* **Proceedings of IEEE Intelligent Vehicles Symposium (IV)**.

APPENDICES



Appendix A

Solving Techniques

Inverse transpose of homography

An inverse transpose of a homography can be useful for plane transformation. Recalled a homography transformation can be written as a matrix:

$$\mathrm{H}_j^i = \begin{bmatrix} \mathrm{R}_j^i & -\mathrm{R}_j^i \boldsymbol{T}_{ji}^j \\ \mathbf{0}_3{}^T & 1 \end{bmatrix} \quad . \tag{61}$$

Therefore, every matrix is preserves the property: $\mathrm{H}_j^i \mathrm{H}_j^{i^{-1}} = \mathbf{I_4}$. Since the homography has 4×4 dimension, the product of the property is equal to identity matrix $\mathbf{I}_4$. Then we have the equation as follow:

$$\begin{bmatrix} \mathrm{R}_j^i & -\mathrm{R}_j^i \boldsymbol{T}_{ji}^j \\ \mathbf{0}_3{}^{\mathrm{T}} & 1 \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3{}^{\mathrm{T}} & 1 \end{bmatrix} \quad , \tag{62}$$

where A, B, C and D are members of $\mathrm{H}_j^{i^{-1}}$.They could be any matrix vector or scalar whose satisfy the equation above. By derive the equation, we can rewrite it as four linear equations:

$$\mathrm{R}_j^i A - \mathrm{R}_j^i \boldsymbol{T}_{ji}^j C = \mathbf{I}_3 \tag{A}$$
$$\mathrm{R}_j^i \mathrm{B} - \mathrm{R}_j^i \boldsymbol{T}_{ji}^j D = \mathbf{0}_3 \tag{B}$$
$$\mathbf{0}_3{}^{\mathrm{T}} A + \mathrm{C} = \mathbf{0}_3{}^{\mathrm{T}} \cdots (3) \tag{C}$$
$$\mathbf{0}_3{}^{\mathrm{T}} B + \mathrm{D} = 1 \cdots (4) \tag{D}$$

Solved those equations and obtained $A = \mathrm{R}_j^{i^{\mathrm{T}}}$, $\mathrm{B} = \boldsymbol{T}_{ji}^j$, $\mathrm{C} = \mathbf{0}_3{}^{\mathrm{T}}$and $\mathrm{D} = 1$. Replaced the answers in $\mathrm{H}_j^{i^{-1}}$:

$$\mathrm{H}_j^{i^{-1}} = \begin{bmatrix} \mathrm{R}_j^{i^{\mathrm{T}}} & \boldsymbol{T}_{ji}^j \\ \mathbf{0}_3{}^{\mathrm{T}} & 1 \end{bmatrix} \quad . \tag{63}$$

Transposed $\mathrm{H}_j^{i^{-1}}$, then, we finally obtain

$$\mathrm{H}_j^{i-\mathrm{T}} = \begin{bmatrix} \mathrm{R}_j^i & \mathbf{0}_3 \\ \boldsymbol{T}_{ji}^{j\,\mathrm{T}} & 1 \end{bmatrix} \quad . \tag{64}$$

Singular Value Decomposition and Pseudo Inverse

Singular Value Decomposition as known as SVD is used to solve a linear equation in $Ax = 0$ form, where A is known parameters according to a mathematical model, and x is unknown variables. To obtain the unknown variables, matrix A must be decomposed into $A = UDV^{\mathrm{T}}$, where D is a diagonal matrix of eigenvalues sorting form the highest to the lowest, columns of U are left singular vectors (*gene coefficient vectors*), and rows of V^{T} are the right singular vectors (*expression level vectors*). The last column of V is the answer for the model, which yield minimum error from the model. Matrix A has dimension of $m \times n$, where m is the number of data, and n is the number of parameters. The result is constrained with the condition: $\|x\| = 1$.

Pseudo inverse is used to solve a linear equation in $Ax = B$ form. Using pseudo inverse, the result is obtained as follow $x = A^{-1}B$, where $A^{-1} = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}$ is a pseudo inverse of matrix A. This method is strict to no constraint.

Rotation of Skew Matrix

Skew matrix is an operation to reform a matrix dimension mostly for matrix multiplication. Rotation matrix is a function of three variables which are ω, ϕ and κ. Therefore the matrix can be represented with skew matrix of three rotation angle as:

$$\mathrm{R}(\omega, \phi, \kappa) = \mathrm{R}_0 \exp\left(\begin{bmatrix} \omega \\ \phi \\ \kappa \end{bmatrix}_\times\right) \quad , \tag{65}$$

where

$$\begin{bmatrix}\omega\\ \phi\\ \kappa\end{bmatrix}_{\times} = \begin{bmatrix}0 & -\kappa & \phi\\ \kappa & 0 & -\omega\\ -\phi & \omega & 0\end{bmatrix} \quad , \tag{66}$$

and R_0 is an initial rotation matrix follow the properties of lie's group. The representation helps us to change the orientation form R_0 without extracting the initial rotation angle. Particularly, it simplify the calculation and preserving the properties of rotation matrix.

Kronecker Product and VEC Operator

Kronecker product is an operator for matrices multiplication, written as $\otimes$. Kronecker product of matrix $A_{(m\times n)}$ and $B_{(p\times q)}$ can be express as:

$$A_{(m\times n)} \otimes B_{(p\times q)} = \begin{bmatrix}a_{11}B & \cdots & a_{1n}B\\ \vdots & \ddots & \vdots\\ a_{m1}B & \cdots & a_{mn}B\end{bmatrix} \quad , \tag{67}$$

where a_{ij} is the member of matrix A in $i-th$ row and $j-th$ column. The dimension of the result matrix is $(mp)\times(nq)$. The operation provides a matrix of all possible of all matrices members.

VEC operator is a tool for multidimensional matrix representation. This operator transform matrix into a vector, by stacking all columns of the matrix underneath the other. Matrix $A_{(m\times n)}$ has a_i to represent $i-th$ column of A, and VEC of A can be written as:

$$vec(A) = \begin{bmatrix}a_1\\ a_2\\ \vdots\\ a_n\end{bmatrix} \tag{68}$$

In the thesis, we applied a one of VEC operator properties, which is $vec(ABC) = (C^{\mathrm{T}}A)\ vec(B)$, to resolve matrices multiplication to be more simpler for the calculation.

Levenberg-Marquardt

In least-square sense, we can write a linear equation as $y = Ax + e$, where e is error. By given $e = 0$, the equation can be resolved to $x = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}y$. That is Gauss-Newton's method. To reduce the error rapidly, Levenberg-Marquardt resolved the equation as: $x = (A^{\mathrm{T}}A + \lambda H)^{-1}A^{\mathrm{T}}y$, where H is a positive semi-definite, then H is given as identity matrix I. λ is a scalar. Therefore we applied to non-linear optimization as follow: $dv = -(f'(v)^{\mathrm{T}}f'(v) + \lambda I)^{-1}f'(v)^{\mathrm{T}}f(v)$. The pseudo-code below shows the applied Levenberg-Marquardt for the experiment.

1. $LM(f(\cdot),\ H_c^l, trial)$

Levenberg-Marquardt optimization is an iterative process, that minimize the objective function $f(\cdot)$ with respect to the parameter H_c^l. Return new H_c^l that fitted the model better than the old one. The objective function $f(\cdot)$ and its first derivative are mentioned earlier.

2. $(\mathrm{R}_c^l, \boldsymbol{T}_{cl}^c) \leftarrow H_c^l$
3. $v_{old} \leftarrow (\mathrm{R}_c^l, \boldsymbol{T}_{cl}^c)$
4. $\lambda \leftarrow 1$
5. $\boldsymbol{for}\ i \leftarrow 1 \cdots trial$
6. $\quad dv = -(f'(v)^{\mathrm{T}}f'(v) + \lambda I)^{-1}f'(v)^{\mathrm{T}}f(v)$
7. $\quad v_{new} \leftarrow v_{old} + dv$
8. $\quad \boldsymbol{if}\ f(v_{new}) > f(v_{old})\ \boldsymbol{do}$
9. $\quad\quad \lambda \leftarrow 1 \times 10$
10. $\quad \boldsymbol{if}\ f(v_{new}) > f(v_{old})\ \boldsymbol{do}$
11. $\quad\quad \lambda \leftarrow 1 \div 10$
12. $(\mathrm{R}_c^l, \boldsymbol{T}_{cl}^c) \leftarrow v_{new}$
13. $H_c^l \leftarrow (\mathrm{R}_c^l, \boldsymbol{T}_{cl}^c)$
14. $\boldsymbol{return}\ H_c^l$.

Appendix Figure A1 Pseudo code of applied Levenberg-Marquardt.

Riemanian and Euclidien Distance

To compare the estimated parameters, i.e. rotation matrix and translation vector, with the ground truth, Riemanian and Euclidien distance respectively are used.

Euclidien distance is a straightforward evaluation method to measure the displacement of position. Hence, it is applied to measure the error of estimated

translation vector compared to ground truth. Let estimated translation vector be $\boldsymbol{T_{est}}$ and the ground truth translation vector $\boldsymbol{T_{ground\ truth}}$. Then the translation vector estimation error can be calculated by:

$$d_T(\boldsymbol{T_{est}}, \boldsymbol{T_{ground\ truth}}) = \|\boldsymbol{T_{ground\ truth}} - \boldsymbol{T_{est}}\|_2 \quad . \tag{69}$$

Riemanian distance is used to measure the distance between two rotation matrices. Riemanian distance between estimated rotation matrix $\mathrm{R}_{\boldsymbol{est}}$ and the ground truth rotation matrix $\mathrm{R}_{\boldsymbol{ground\ truth}}$ can be calculated by:

$$d_{\mathrm{R}}(\mathrm{R}_{\boldsymbol{est}}, \mathrm{R}_{\boldsymbol{ground\ truth}}) = \frac{1}{\sqrt{2}} \|Log(\mathrm{R}_{\boldsymbol{est}}{}^{\mathrm{T}} \mathrm{R}_{\boldsymbol{ground\ truth}})\|_{\mathrm{F}} \quad , \tag{70}$$

where $\|\cdot\|_{\mathrm{F}}$ is Frobenius norm. The logarithm of orthonormal matrix is:

$$Log(\mathrm{R}) = \begin{cases} 0 & if\ \theta = 0 \\ \dfrac{\theta}{2sin\theta} & if\ \theta \neq 0 \end{cases} , \tag{71}$$

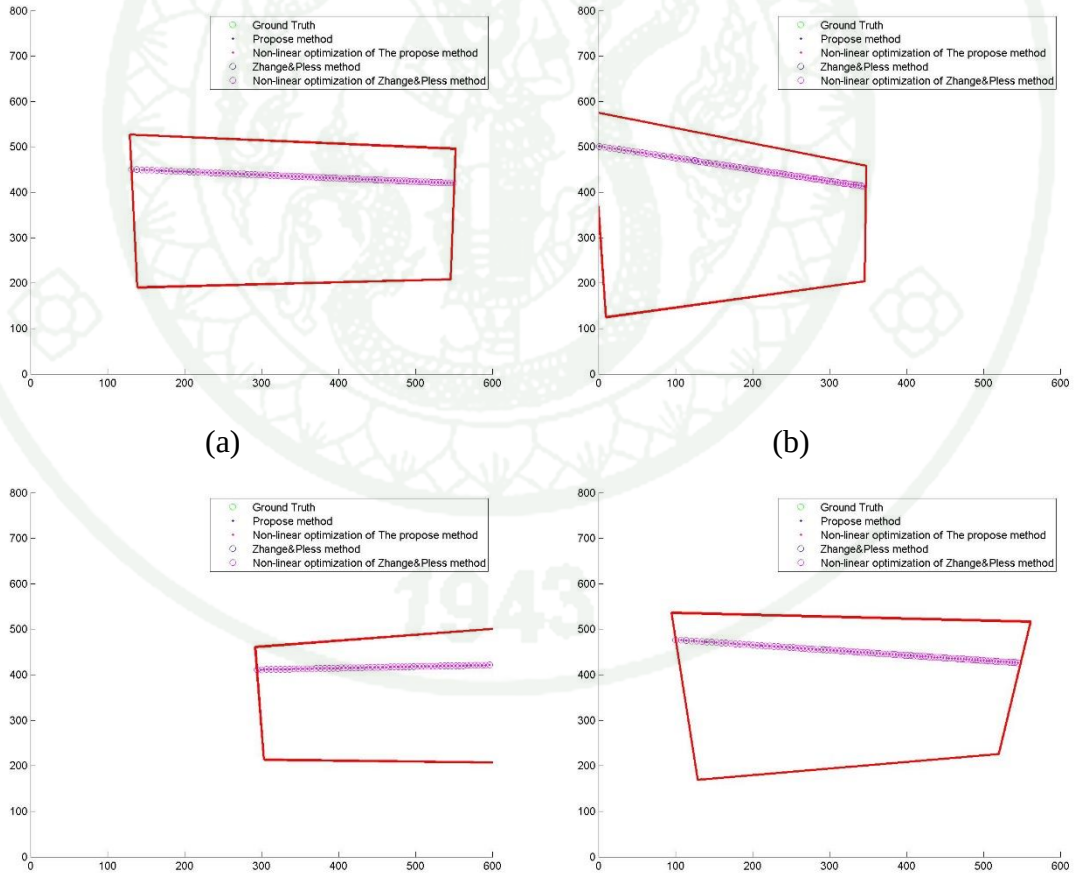
where θ must satisfy the conditions $trace(\mathrm{R}) = 1 + 2cos\theta$ and $|\theta| < \pi$ (this formula breaks down when $\theta = \pm\pi$ according to (Moakher, 2002).

Appendix B

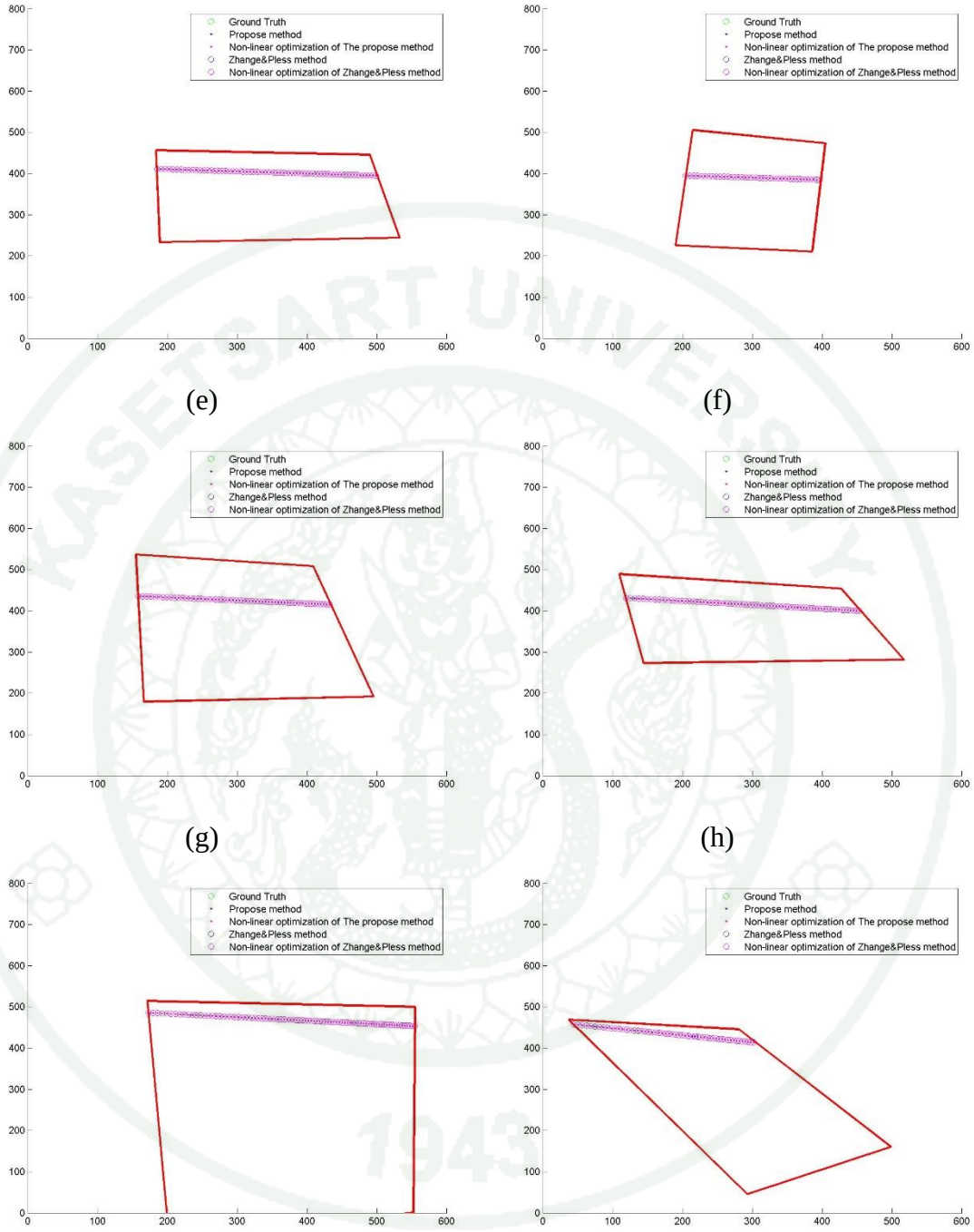
Experimental Results

Simulation Results

LiDAR-camera estimated pose is shown in Appendix Figure B1 compare with the method of Zhang and Pless. The following figure shows twenty simulated situations of the experimental setup shown on images. The reprojected data compares with ground truth (green points). Four kinds of reprojected data are displayed; reprojected points using estimated parameters of our proposed method (blue points), reprojected points using estimated parameters of Zhang and Pless method (pink points), reprojected points using optimized parameters of our proposed method (blue circles) and reprojected points using optimized parameters of Zhang and Pless method (pink circles).



Appendix Figure B1 Simulation results.

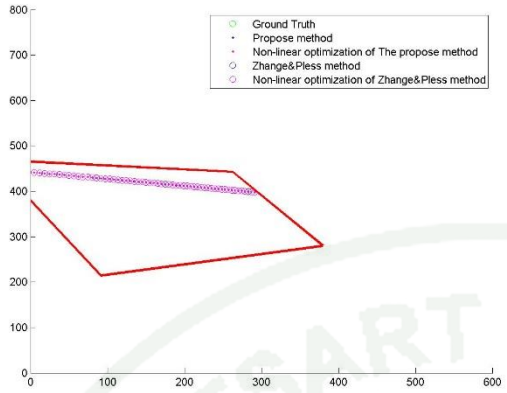


(e) (f)

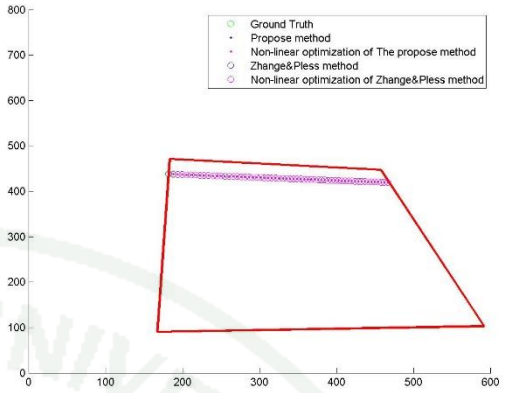
(g) (h)

(i) (j)

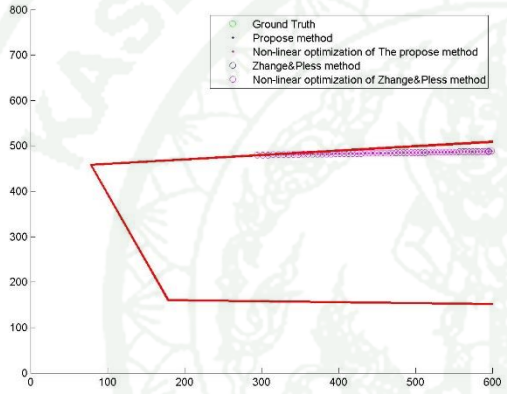
Appendix Figure B1 (Continued)



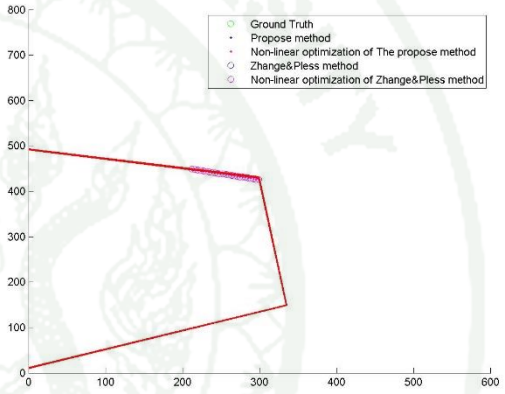
(k)



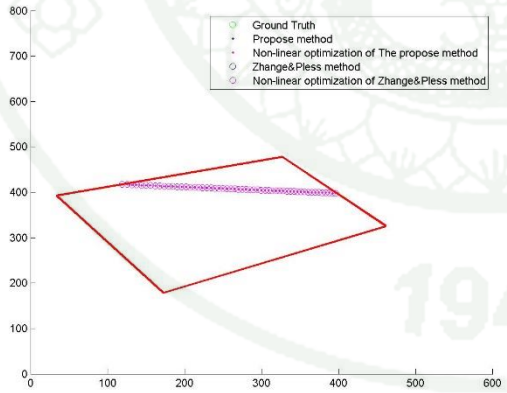
(l)



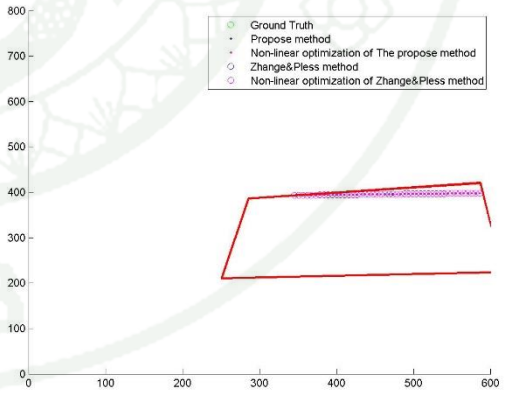
(m)



(n)

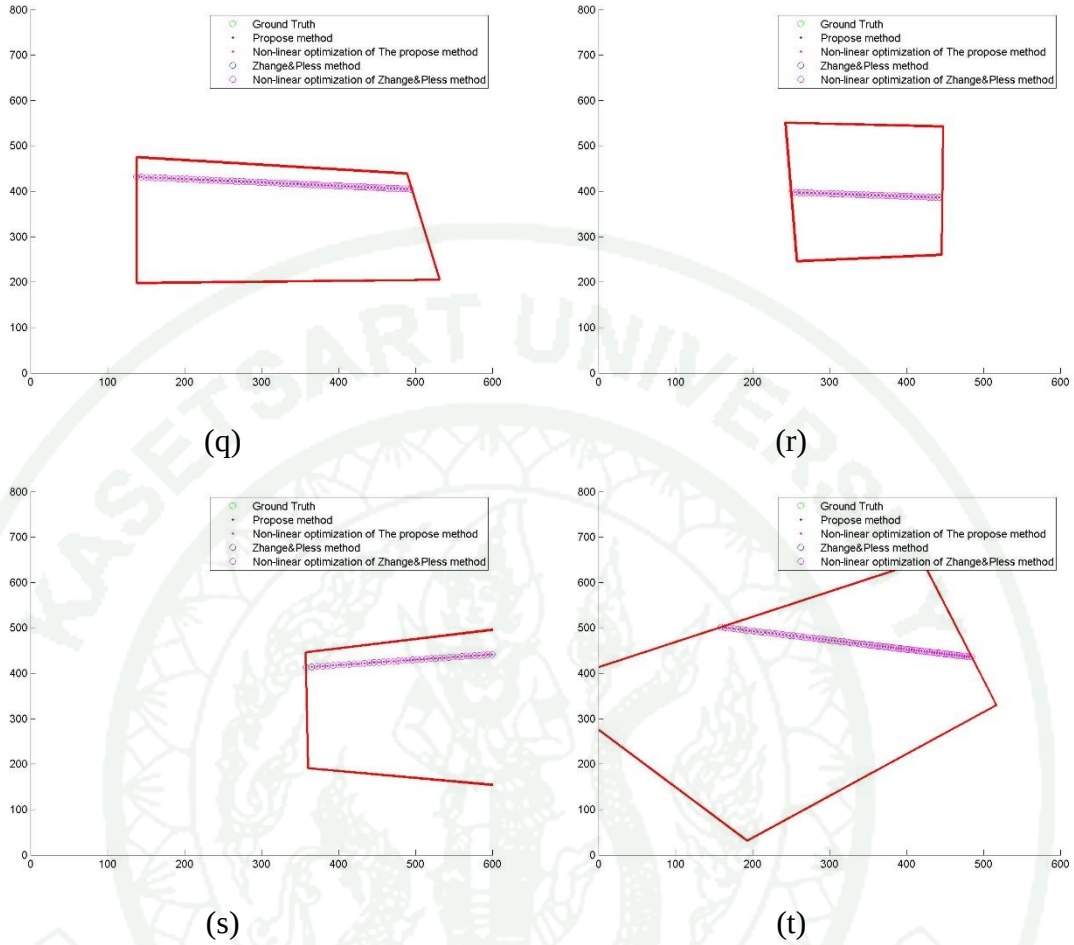


(o)



(p)

Appendix Figure B1 (Continued)



Appendix Figure B1 (Continued)

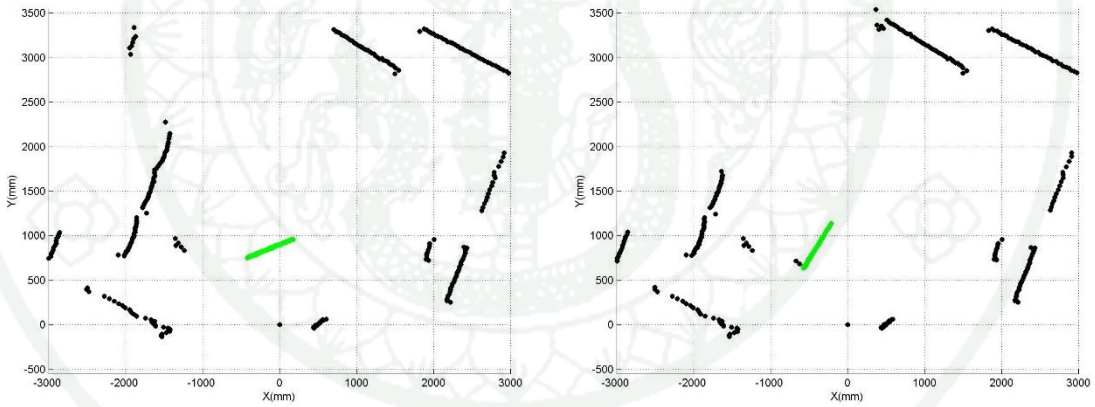
Real Data Results

There are four results using real data to estimate the pose between the LiDAR and the camera. Three of them using conventional color webcam and the other infrared webcam. Although the infrared webcam is intentionally used to detect the LiDAR scan plane on the checkerboard, we also used it for calibration as well. The result using infrared webcam yielded unsatisfied outcome because some of checkerboard corners are lost; the checkerboard pose is inaccurately estimated. Each trail shows only ten of pre-processing results and the calibrated results, i.e. detected checkerboards in LiDAR data and camera calibration results, and the comparison between the calibration result of our proposed method and of Zhang and Pless. The

order of subfigures are matched in each trail. Detected checkerboards in LiDAR data set is colored green. Camera calibration shows the result of corner detection (blue circle) and resolved positions (red cross) using camera calibration (Zhang, 2000). The result of LiDAR calibration compares the reprojected data using the optimized parameter that estimate from our proposed method (red circle) (Tulsuk *et al*., 2014) and Zhang and Pless's (blue cross).

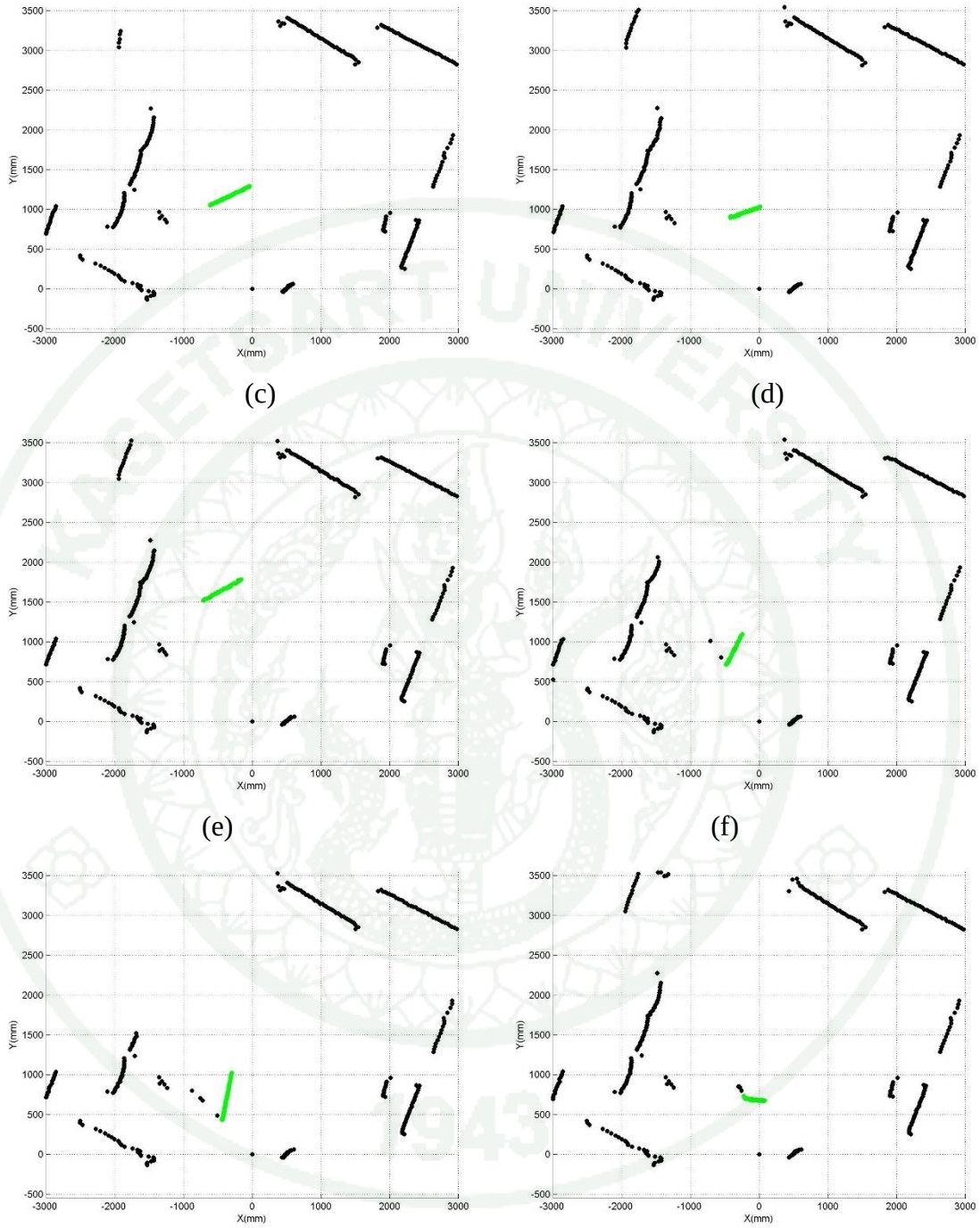
First Trail

Detected checkerboards in first LiDAR data set are shown in Appendix Figure B2, presented in green points. The illustrated figure is the result after performed hough transform and RANSAC.

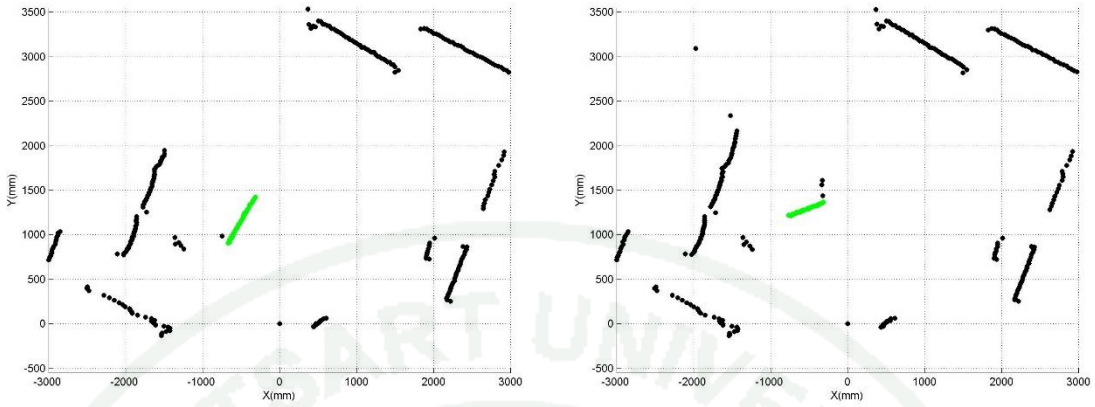


(a) (b)

Appendix Figure B2 First trail: Detected checkerboard in LiDAR data.



Appendix Figure B2 (Continued)



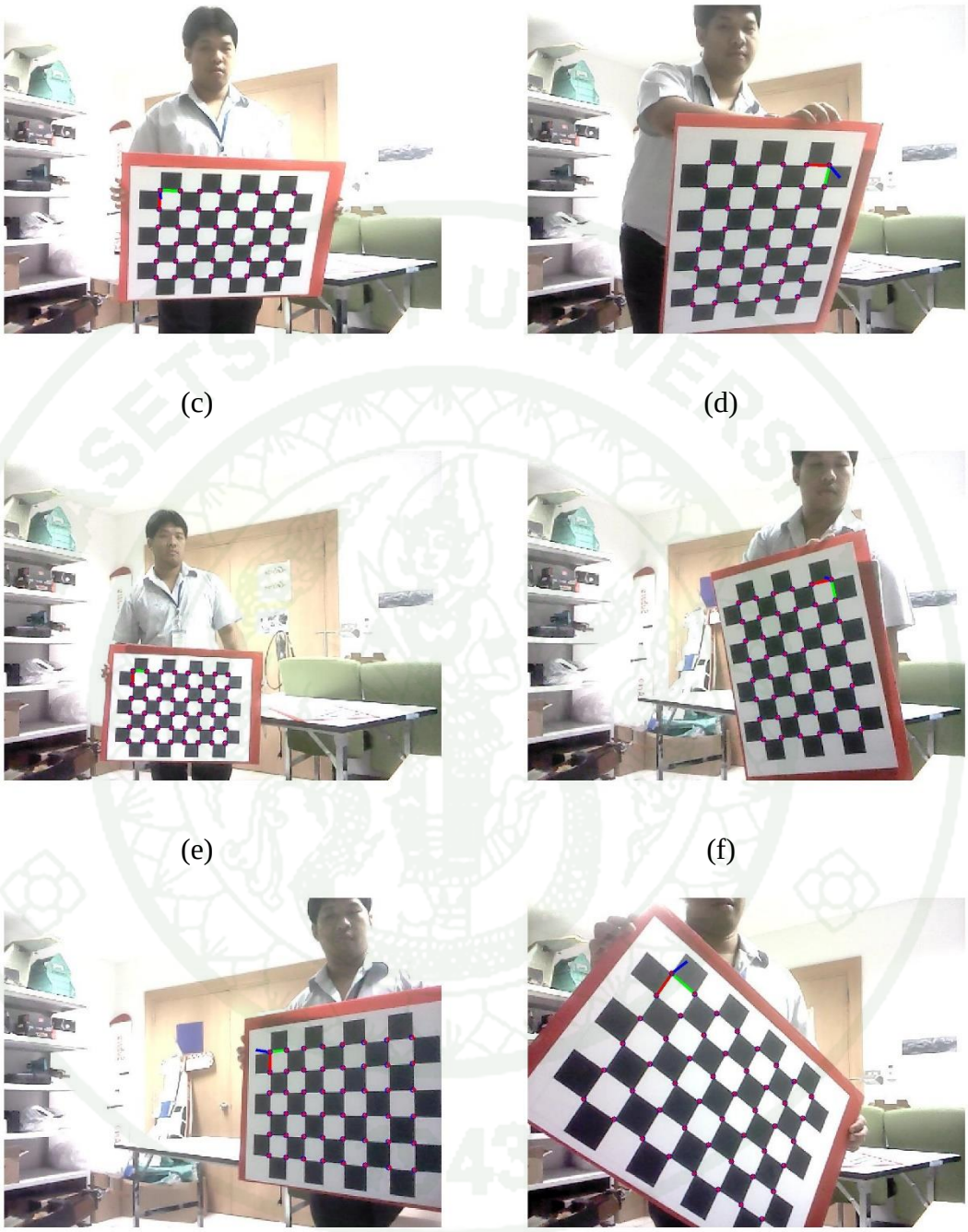
(i) (j)

Appendix Figure B2 (Continued)

Zhang camera calibration result of first data set shows in Appendix Figure B3 The result showing as the reprojection of every corner of the checkerboard onto the related image.

(a) (b)

Appendix Figure B3 First trail: Camera calibration result.

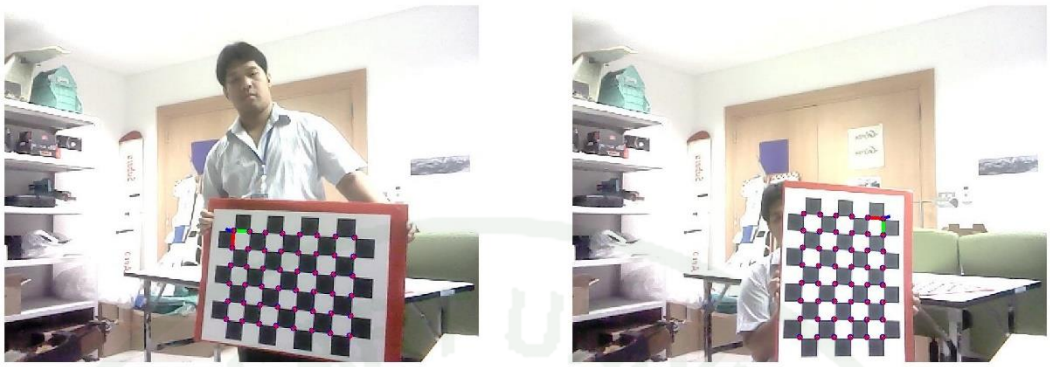


(c) (d)

(e) (f)

(g) (h)

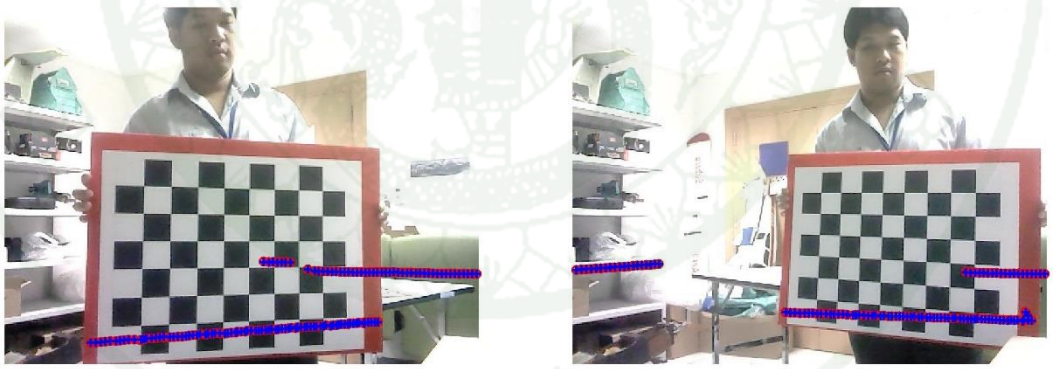
Appendix Figure B3 (Continued)



(i) (j)

Appendix Figure B3 (Continued)

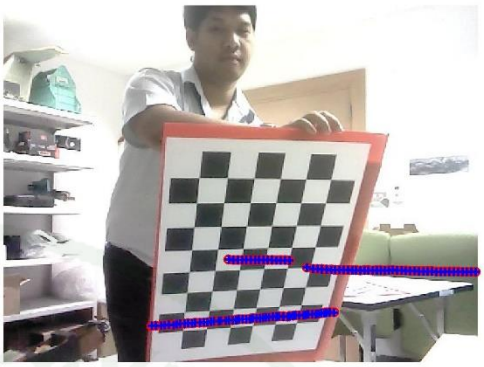
LiDAR-camera estimated pose of first data set is shown as the reprojection of LiDAR scan onto the related image comparing with Zhang and Pless method, see Appendix Figure B4



(a) (b)

Appendix Figure B4 First trail: LiDAR calibration result.

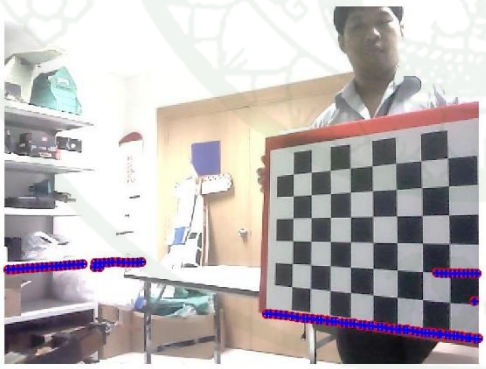
(c)



(d)

(e)

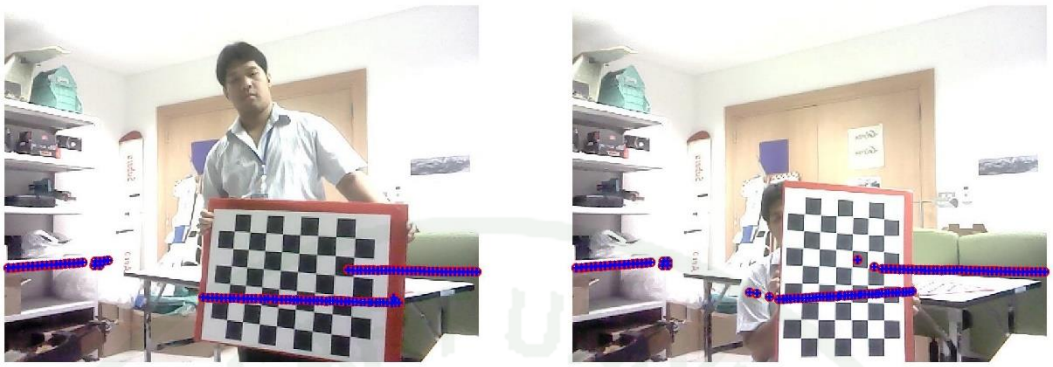
(f)



(g)

(h)

Appendix Figure B4 (Continued)

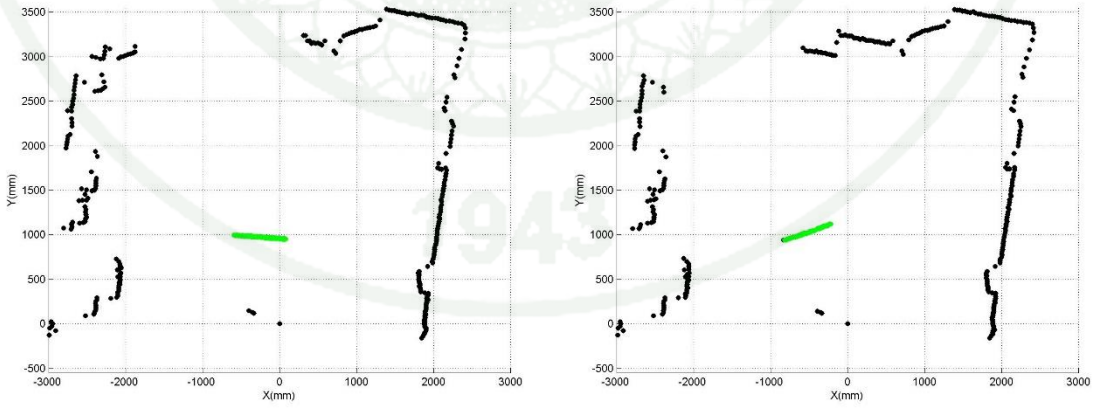


(i) (j)

Appendix Figure B4 (Continued)

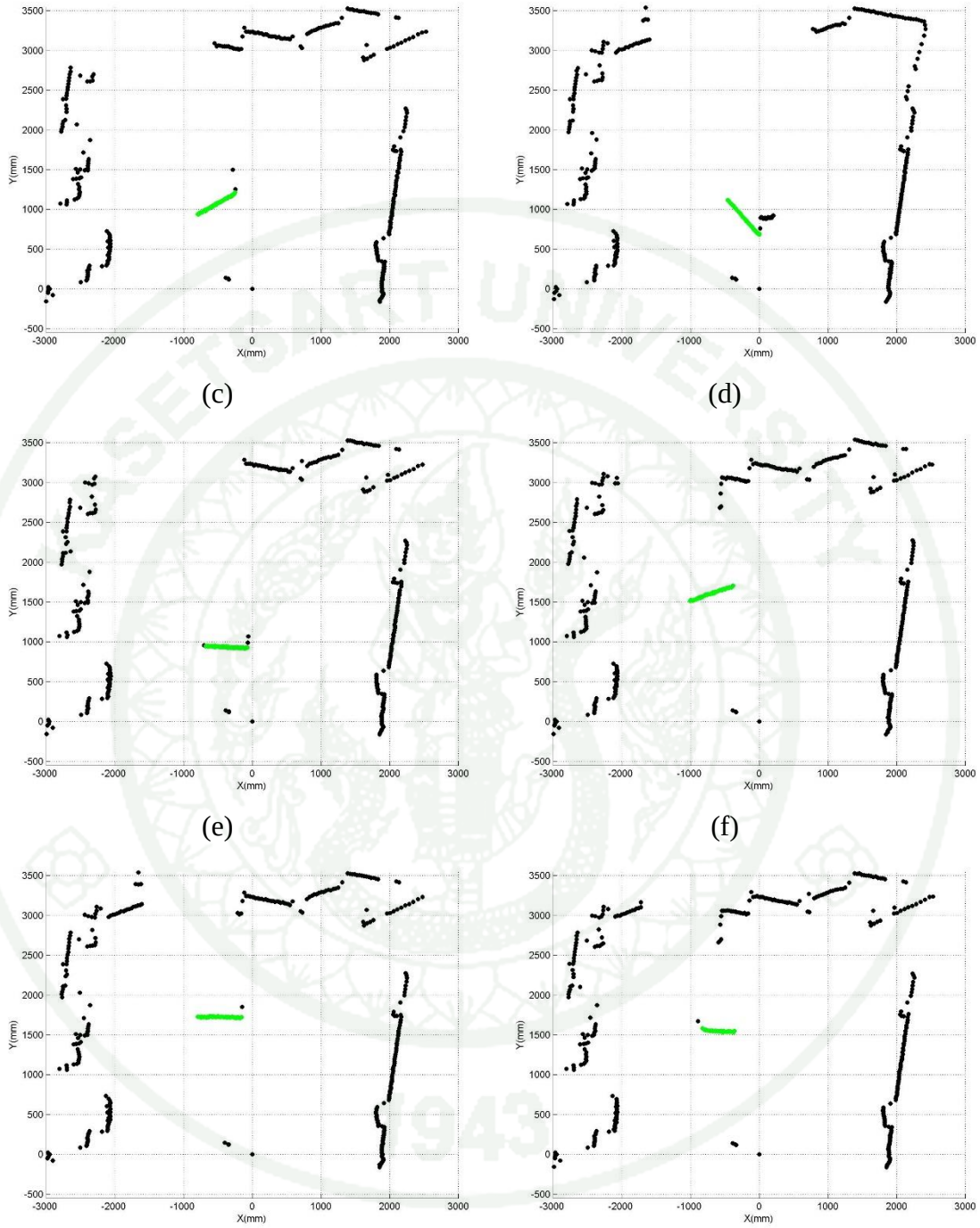
Second Trail

Detected checkerboards in second LiDAR data set are shown in Appendix Figure B5 presented in green points. The illustrated figure is the result after performed hough transform and RANSAC.



(a) (b)

Appendix Figure B5 Second trail: Detected checkerboard in LiDAR data.

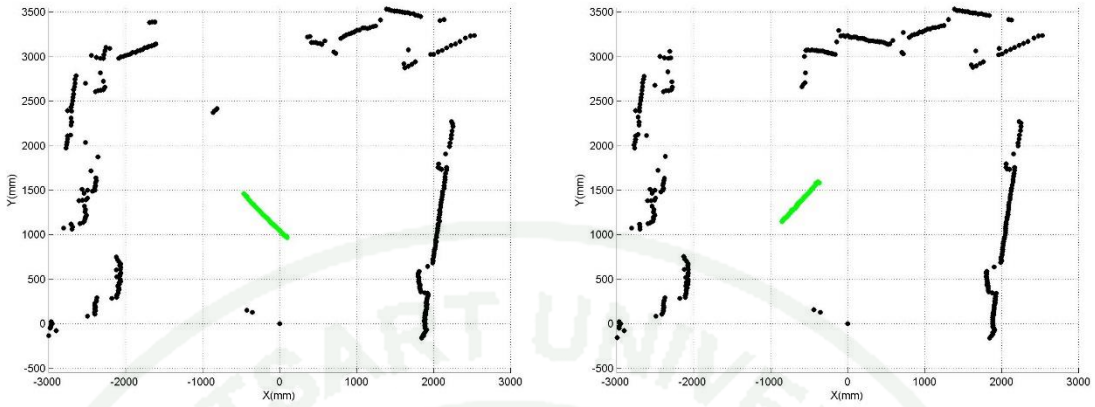


(c) (d)

(e) (f)

(g) (h)

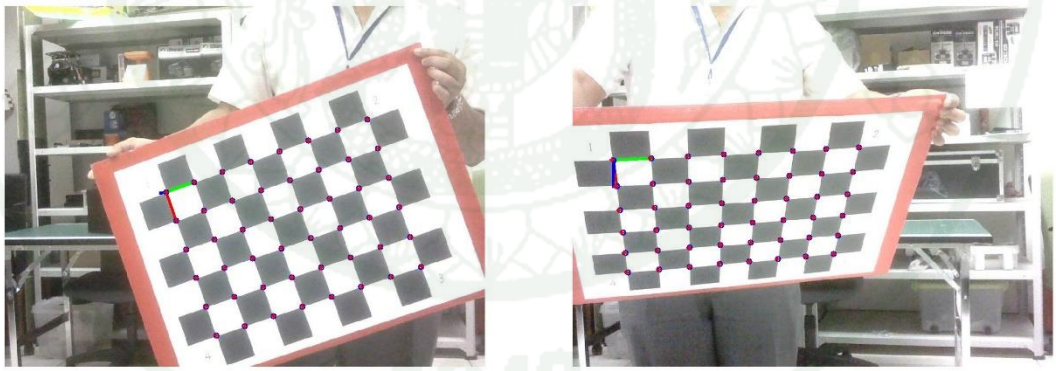
Appendix Figure B5 (Continued)



(i) (j)

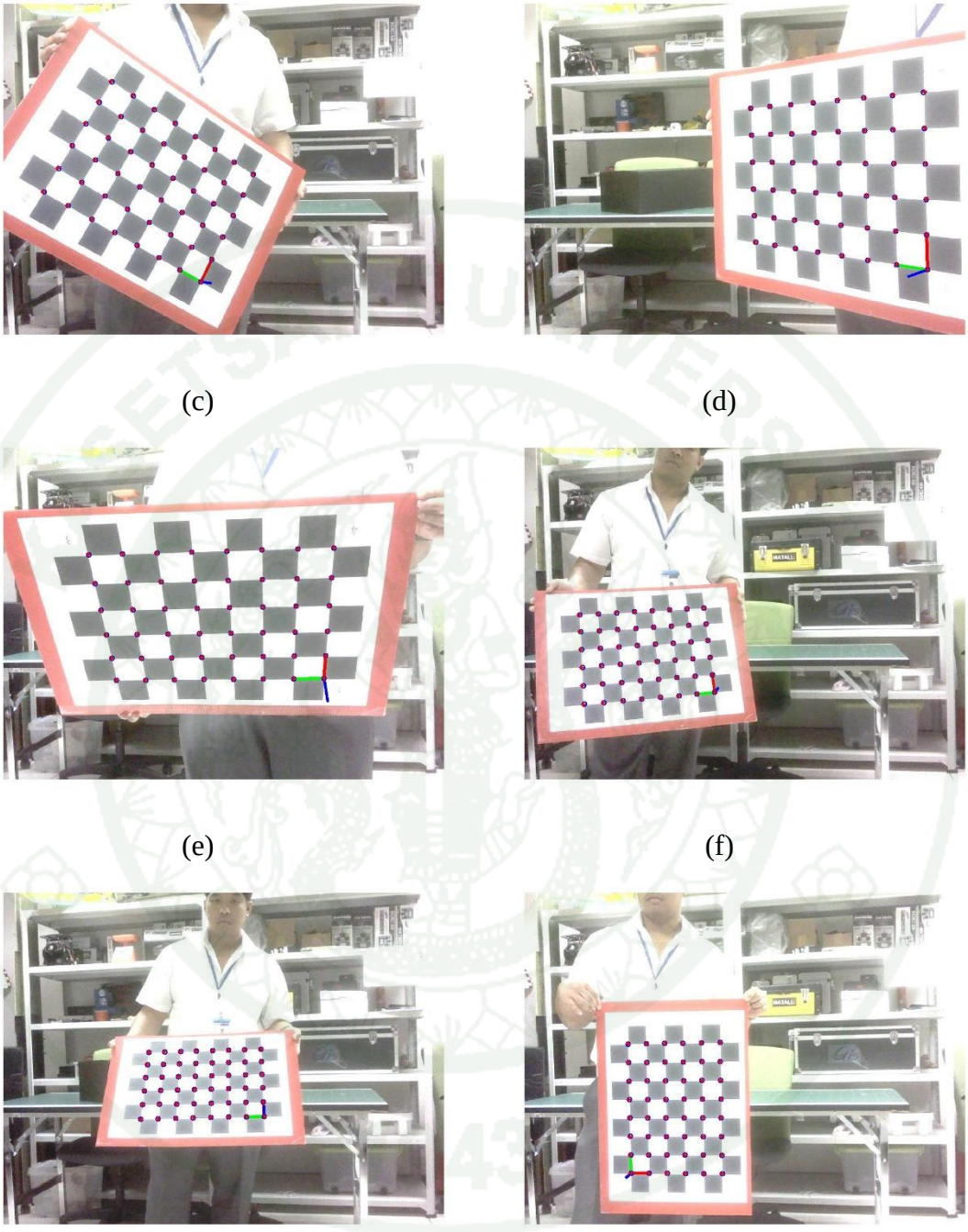
Appendix Figure B5 (Continued)

Zhang camera calibration result of second data set shows in Appendix Figure B6. The result showing as the reprojection of every corner of the checkerboard onto the related image.



(a) (b)

Appendix Figure B6 Second trail: Camera calibration result.

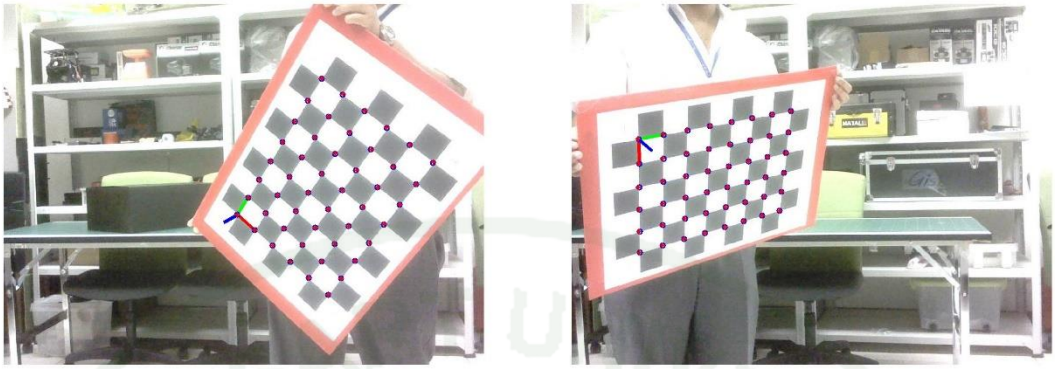


(c) (d)

(e) (f)

(g) (h)

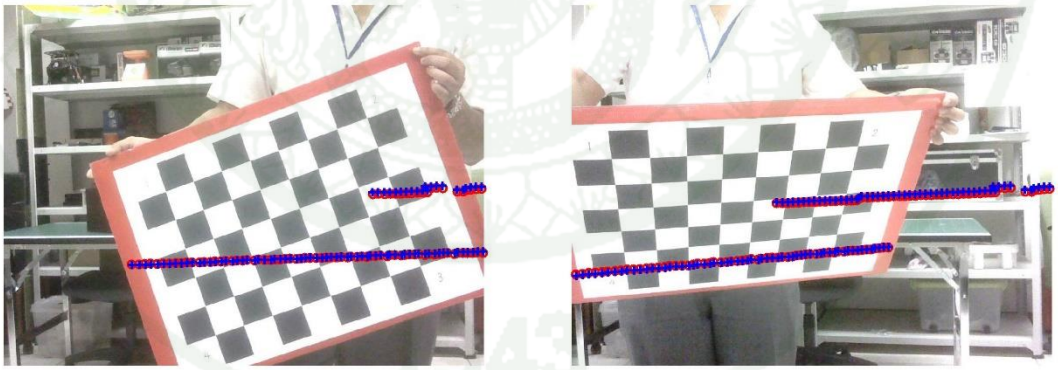
Appendix Figure B6 (Continued)



(i) (j)

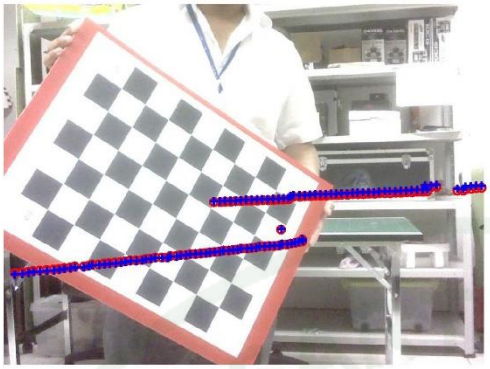
Appendix Figure B6 (Continued)

LiDAR-camera estimated pose of second data set is shown as the reprojection of LiDAR scan onto the related image comparing with Zhang and Pless method, see Appendix Figure B7.

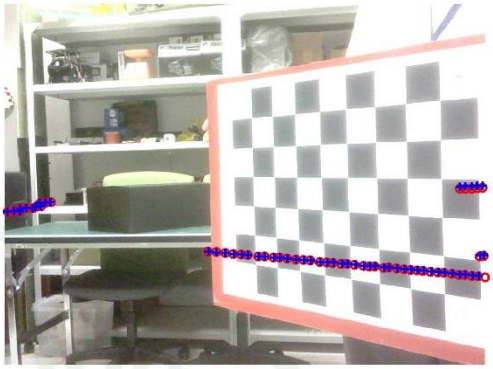


(a) (b)

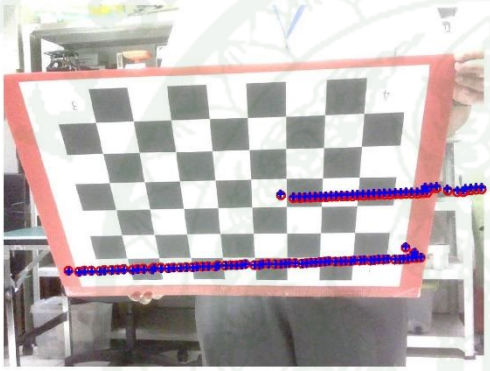
Appendix Figure B7 Second trail: LiDAR calibration result.



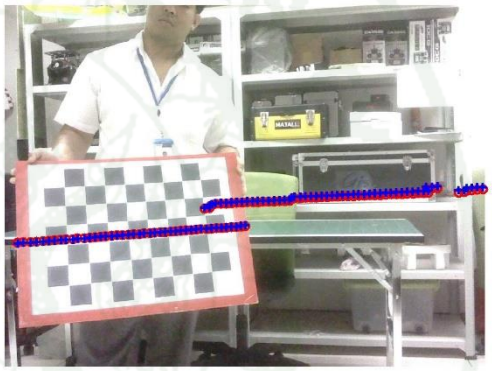
(c)



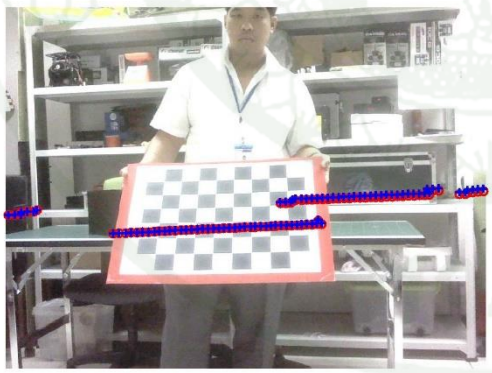
(d)



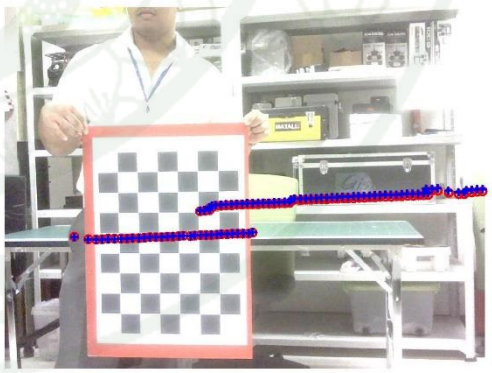
(e)



(f)

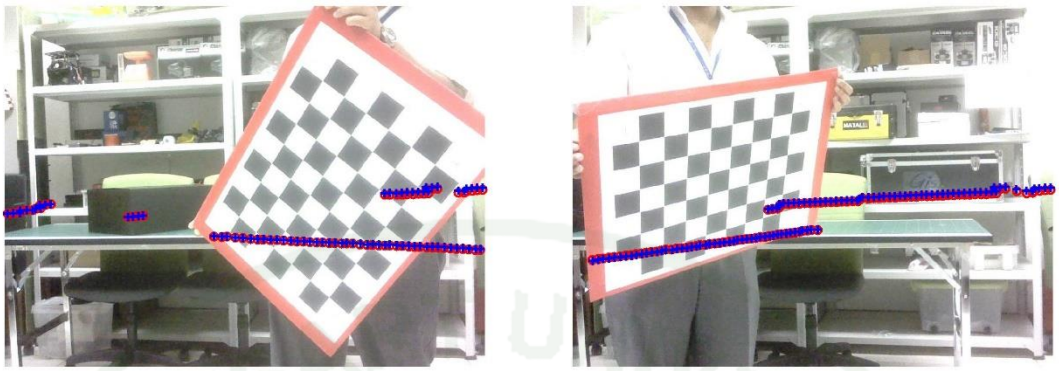


(g)



(h)

Appendix Figure B7 (Continued)

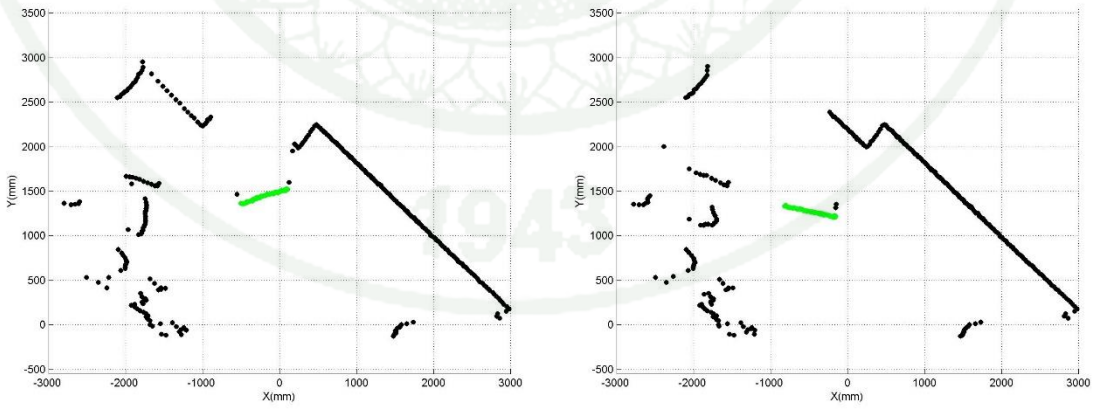


(i) (j)

Appendix Figure B7 (Continued)

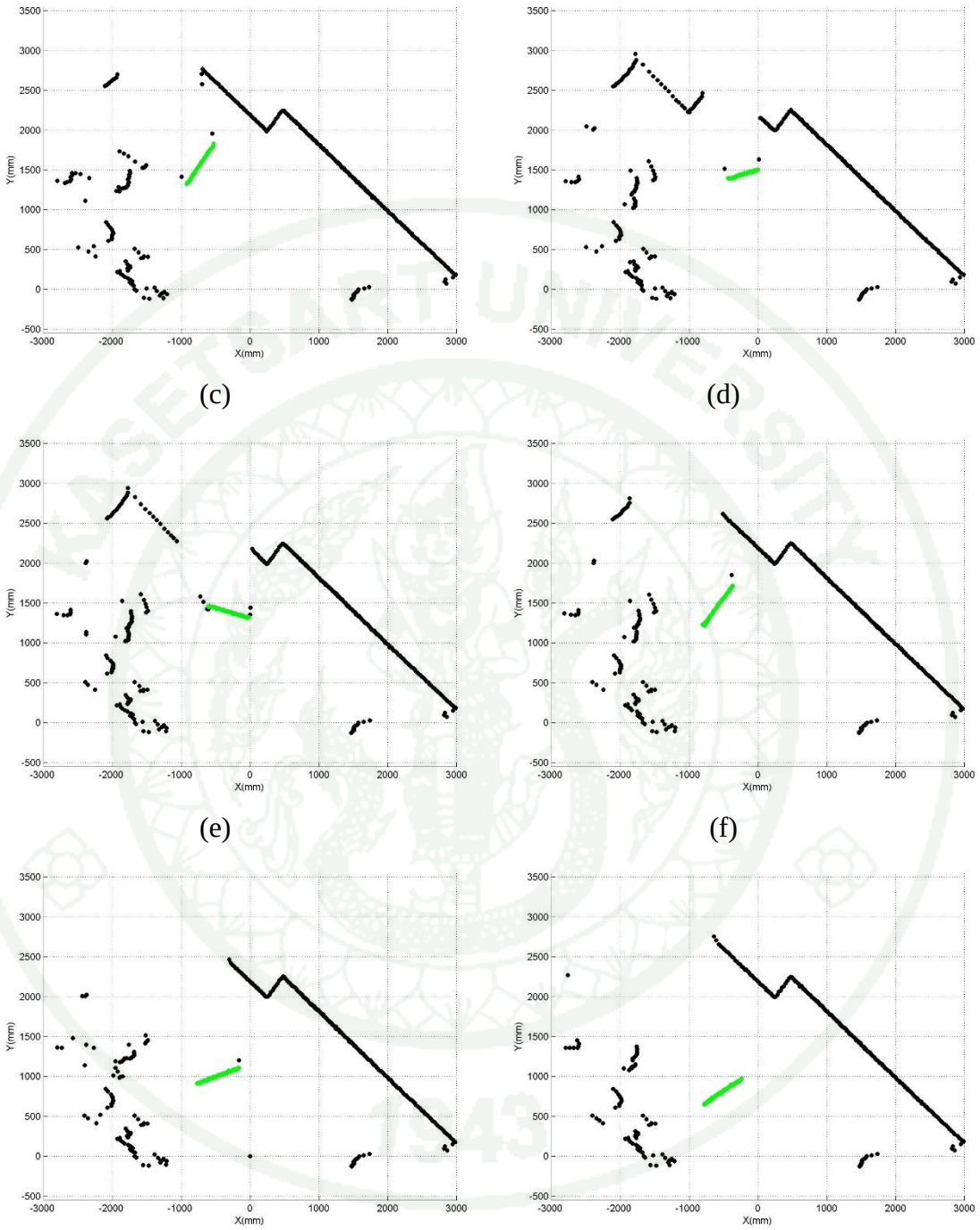
Third Trail

Detected checkerboards in third LiDAR data set are shown in Appendix Figure B8, presented in green points. The illustrated figure is the result after performed hough transform and RANSAC.



(a) (b)

Appendix Figure B8 Third trail: Detected checkerboard in LiDAR data.

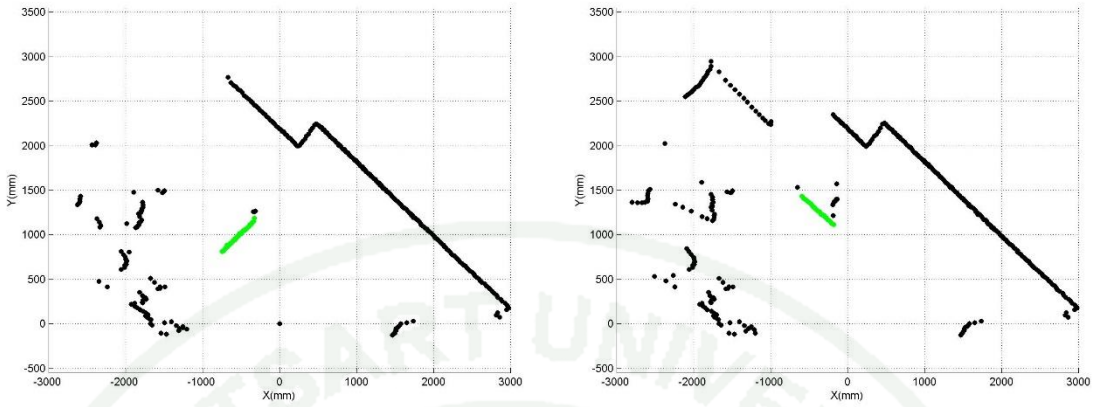


(c) (d)

(e) (f)

(g) (h)

Appendix Figure B8 (Continued)



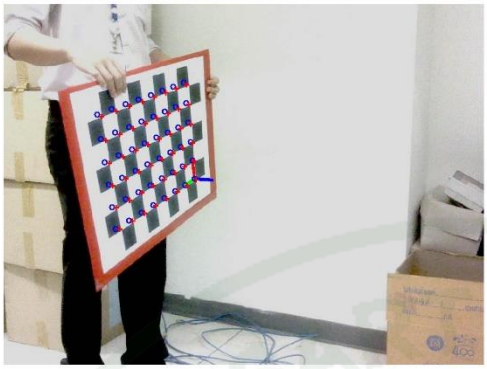
(i) (j)

Appendix Figure B8 (Continued)

Zhang camera calibration result of third data set shows in Appendix Figure B9. The result showing as the reprojection of every corner of the checkerboard onto the related image.

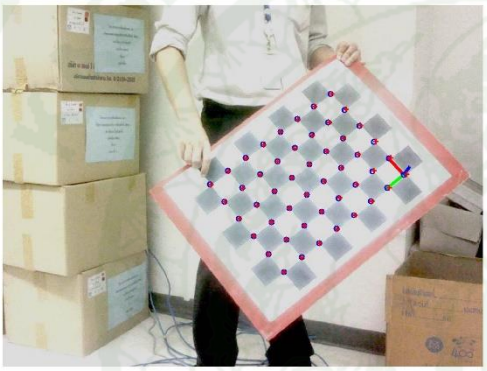
(a) (b)

Appendix Figure B9 Third trail: Camera calibration result.



(c)

(d)



(e)

(f)

(g)

(h)

Appendix Figure B9 (Continued)

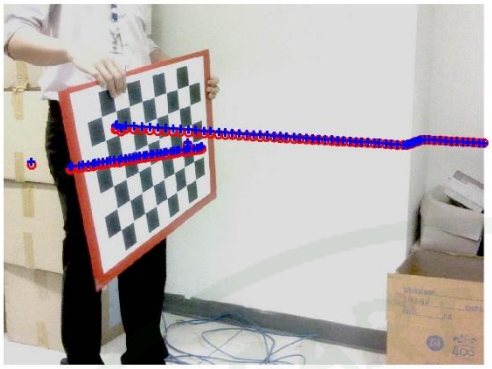
(i) (j)

Appendix Figure B9 (Continued)

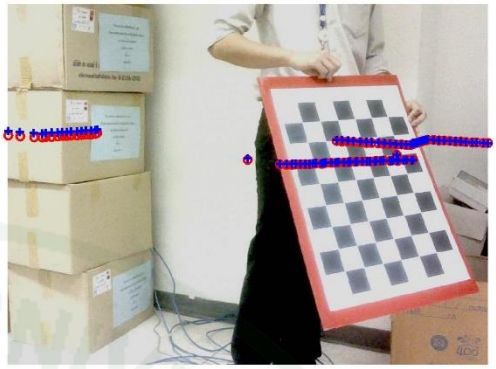
LiDAR-camera estimated pose of third data set is shown as the reprojection of LiDAR scan onto the related image comparing with Zhang and Pless method, see Appendix Figure B10.

(a) (b)

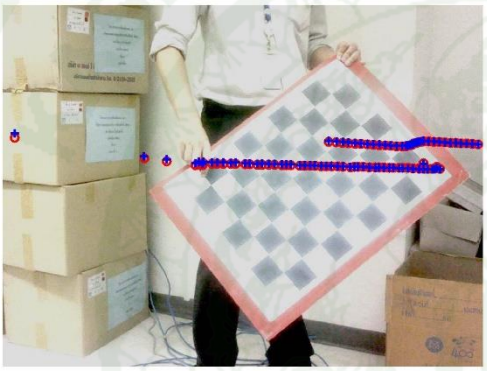
Appendix Figure B10 Third trail: LiDAR calibration result.



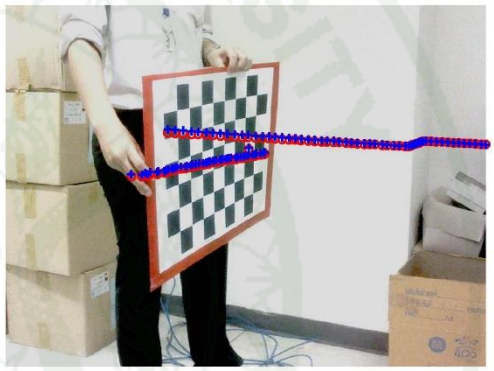
(c)



(d)

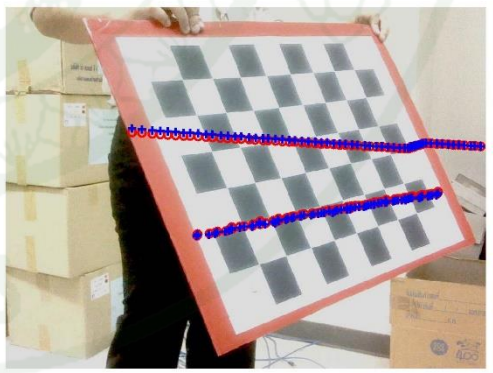


(e)



(f)

(g)



(h)

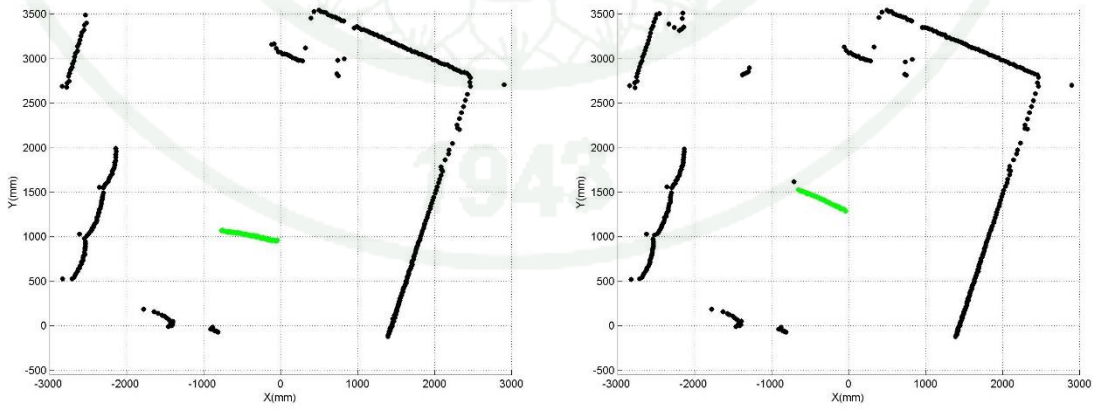
Appendix Figure B10 (Continued)

(i) (j)

Appendix Figure B10 (Continued)

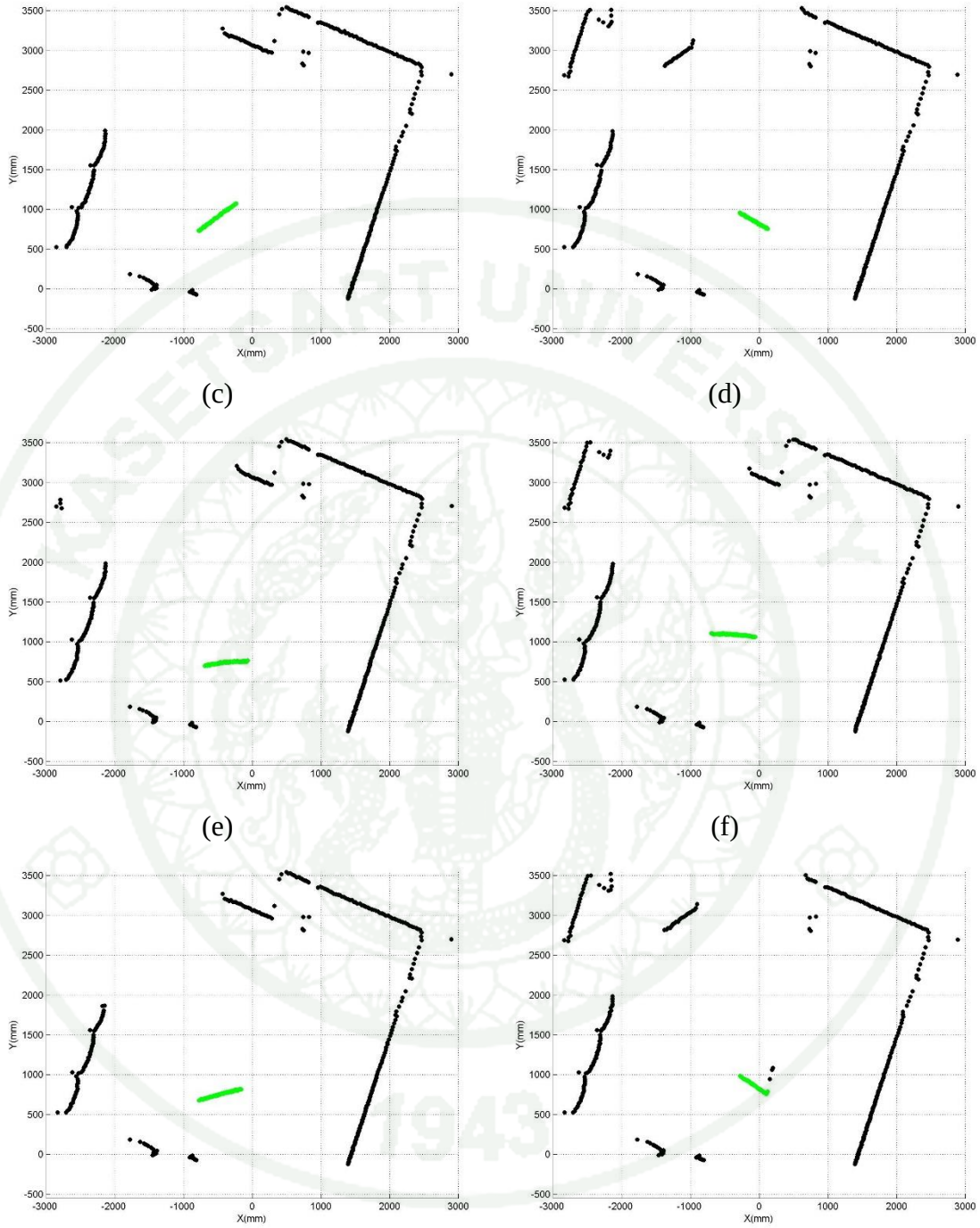
Fourth Trail

This trail is the result using infrared camera instead of color cameraDetected checkerboards in fourth LiDAR data set are shown in Appendix Figure B11, presented in green points. The illustrated figure is the result after performed hough transform and RANSAC.



(a) (b)

Appendix Figure B11 Fourth trail: Detected checkerboard in LiDAR data.

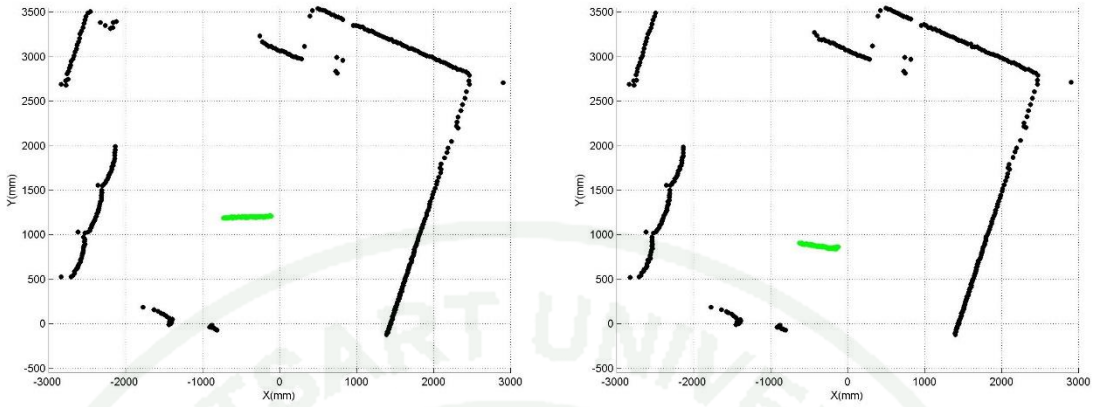


(c) (d)

(e) (f)

(g) (h)

Appendix Figure B11 (Continued)



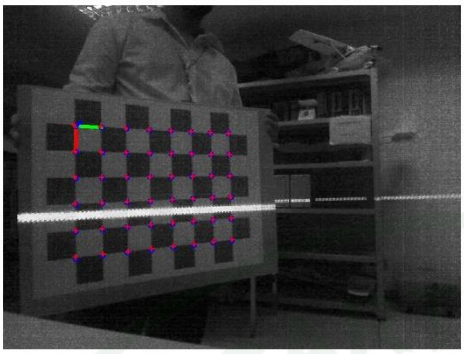
(i) (j)

Appendix Figure B11 (Continued)

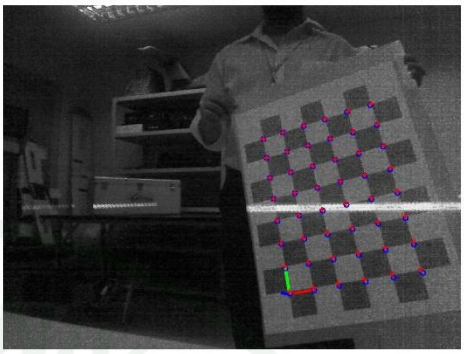
Zhang camera calibration result of fourth data set shows in Appendix Figure B12. The result showing as the reprojection of every corner of the checkerboard onto the related image.

(a) (b)

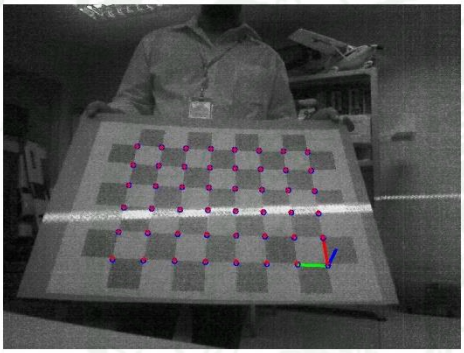
Appendix Figure B12 Fourth trail: Camera calibration result.



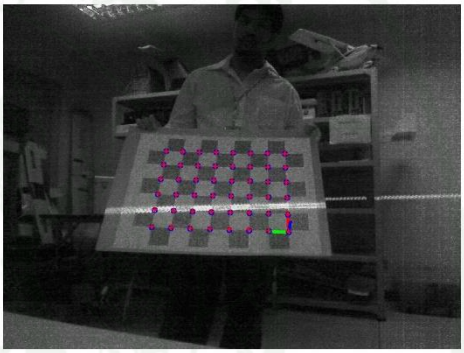
(c)



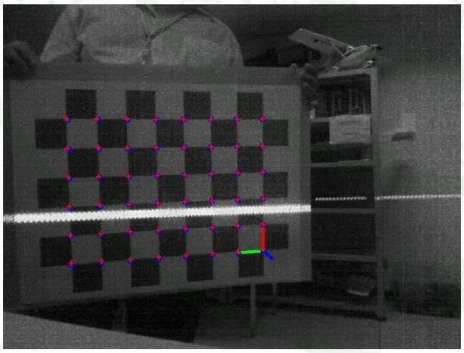
(d)



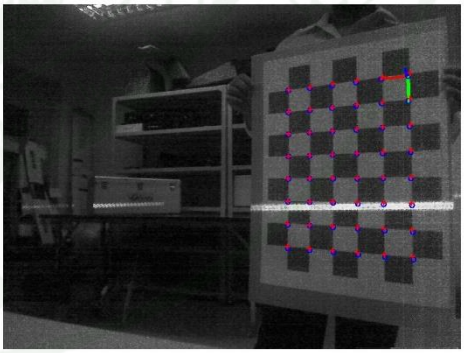
(e)



(f)

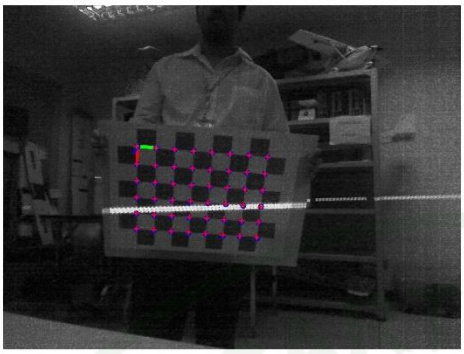
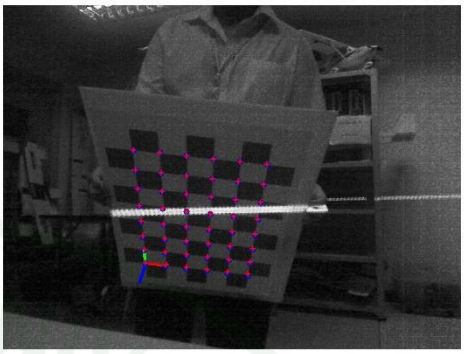


(g)



(h)

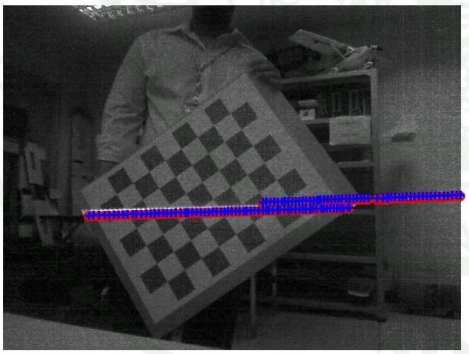
Appendix Figure B12 (Continued)

(i) (j)

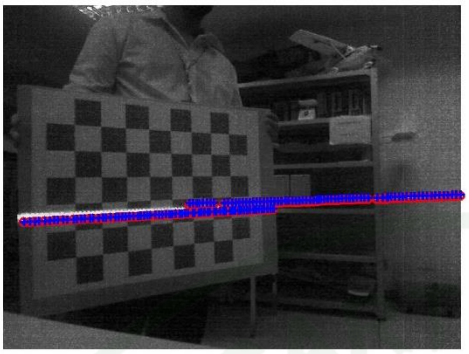
Appendix Figure B12 (Continued)

LiDAR-camera estimated pose of the fourth data set is shown as the reprojection of LiDAR scan onto the related image comparing with Zhang and Pless method, see Appendix Figure B13.

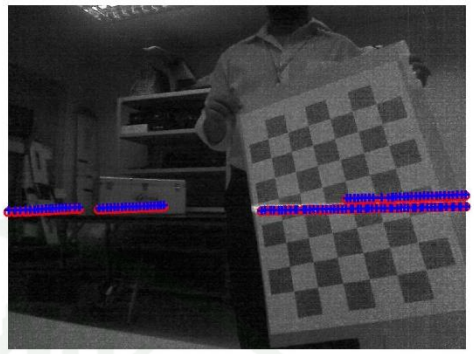



(a) (b)

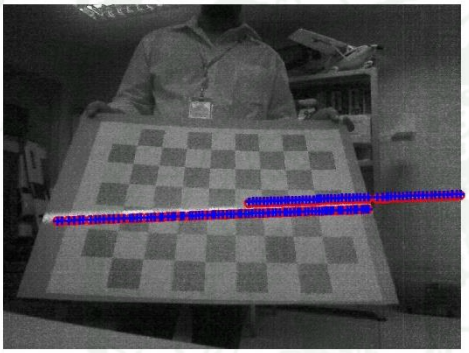
Appendix Figure B13 Fourth trail: LiDAR calibration result.



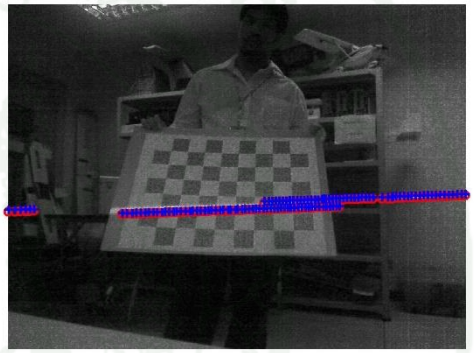
(c)



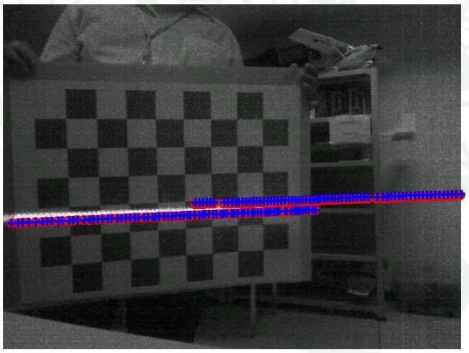
(d)



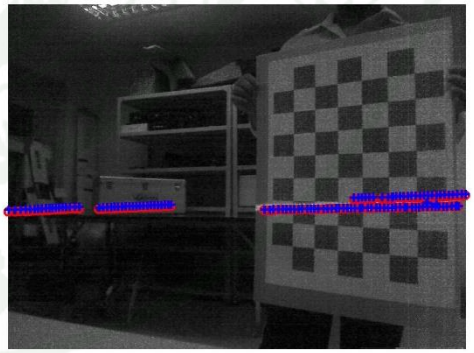
(e)



(f)

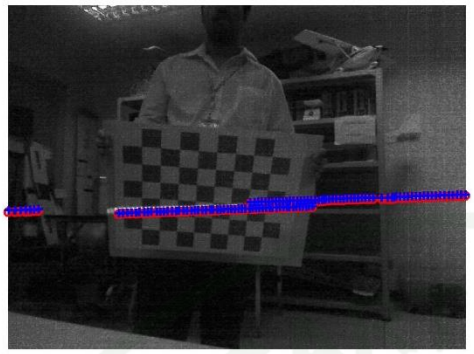


(g)

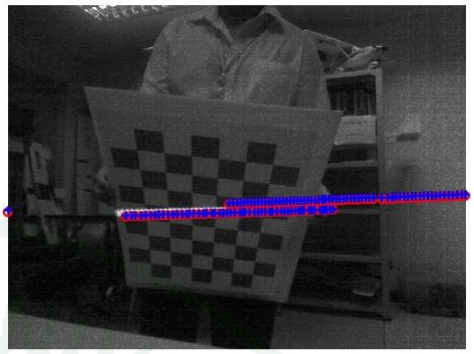


(h)

Appendix Figure B13 (Continued)



(i)



(j)

Appendix Figure B13 (Continued)

CIRRICULUM VITAE

NAME : Mr. Pakapoj Tulsuk

BIRTH DATE : January 14, 1990

BIRTH PLACE : Bangkok, Thailand

EDUCATION :

YEAR	**INSTITUTE**	**DEGREE/DIPLOMA**
2011	Kasetsart University	B.Eng. (Computer Engineering)
2013	Kasetsart University	M.Eng. (Information and Communication Technology for Embedded Systems)

POSITION/TITLE : -

WORK PLACE : -

SCHOLARSHIP/AWARDS : TAIST ICTES Master Degree Scholarship

PUBLICATIONS : ICICTES2014 ,and IV'14