An Analytical Least-Squares Solution to the Line Scan LIDAR-Camera Extrinsic Calibration Problem

Chao X. Guo and Stergios I. Roumeliotis

Abstract— In this paper, we present an elegant solution to the 2D LIDAR-camera extrinsic calibration problem. Specifically, we develop a simple method for establishing correspondences between a line-scan (2D) LIDAR and a camera using a small calibration target that only contains a straight line. Moreover, we formulate the nonlinear least-squares problem for finding the unknown 6 degree-of-freedom (dof) transformation between the two sensors, and solve it analytically to determine its global minimum. Additionally, we examine the conditions under which the unknown transformation becomes unobservable, which can be used for avoiding ill-conditioned configurations. Finally, we present extensive simulation and experimental results for assessing the performance of the proposed algorithm as compared to alternative analytical approaches.

I. INTRODUCTION AND RELATED WORK

LIDAR-camera systems are widely used in various robotic applications primarily due to their complementary sensing capabilities. For instance, the camera's scale ambiguity can be determined from the LIDAR measurements, while motion estimates from the camera can be used for finding correspondences between two LIDAR scans. Additionally, when combining information from both sensors one can achieve higher motion-estimation accuracy. Fusing, however, measurements from a LIDAR and a camera requires precise knowledge of the 6 dof transformation between them. Since 2D LIDARs are more widely used due to their significantly lower cost and size, we hereafter focus on 2D LIDAR-camera calibration methods.¹

An approximate least-squares solution to the 2D LIDARcamera extrinsic calibration problem is presented in [5]. Specifically, the authors first determine the camera's pose with respect to a checkerboard using the PnP algorithm [6], and define a geometric constraint relating the LIDAR and camera measurements on the checkerboard (the cameracheckerboard distance equals the projection of the LIDAR points onto the checkerboard's normal vector). Using this geometric constraint, a linear least-squares problem is formulated by defining, as a new variable, a nonlinear quadratic term which embeds both the unknown rotation and translation. However, the proposed method is *suboptimal* since its solution does not directly satisfy the rotational matrix constraint. Therefore, the computed rotation has to be approximated by projecting the obtained solution on the special orthogonal group, SO(3).

The work described in [7], provides an analytical solution to the minimal LIDAR-camera calibration problem (i.e., using six LIDAR and camera measurement sets). In order to facilitate data association, a calibration board containing white and black bands is used. The transition line between the white and black bands is detected by both the camera (using image processing techniques) and the LIDAR (based on the differences in reflection intensities). Using these matches, geometric constraints are formed (the LIDARmeasured 3D points on a transition line belong to the plane defined by the transition line detected in the image and the camera's center) and solved analytically. This deterministic method, however, does not compute the optimal least-squares solution. Moreover, it is sensitive to noise. Thus, when more than the minimum required number of measurements are available, it must be used in conjunction with RANSAC [8] so as to improve its robustness to noise.

The work described in this paper, makes the following main contributions:

- We introduce a simple, yet powerful, calibration procedure, where the calibration target used only contains a straight line. Moreover, we do not require measuring the laser intensity which makes our method applicable to a wider range of LIDARs.
- We formulate the LIDAR-camera extrinsic calibration problem as a least-squares minimization problem and solve it analytically to find the optimal values for the 6 dof unknown transformation.
- We investigate the conditions under which the LIDARcamera transformation becomes (un)observable.
- We validate the accuracy of our approach both in simulations and experimentally, and demonstrate the performance improvement as compared to the analytical approach of [7].

The rest of the paper is structured as follows. In Section II, we describe our calibration setup and formulate the leastsquares problem for finding the LIDAR-camera calibration parameters. We present the details of our analytical solution in Section III, while Section IV describes our observability analysis. Sections V and VI present the simulation and experimental validation of the proposed algorithm, respec-

The authors are with the Department of Computer Science & Engineering, University of Minnesota, Minneapolis, MN 55455, USA {chaguo|stergios}@cs.umn.edu

This work was supported by the University of Minnesota through the Digital Technology Center (DTC), and the National Science Foundation (IIS-0835637).

¹Note that existing 3D LIDAR-camera extrinsic calibration methods cannot be used since they rely on 3D LIDAR measurements for finding the normal vector to a calibration target [1], [2], or aligning LIDAR depth discontinuities with image edges [3], which is not possible when using a 2D LIDAR. While this is not an issue for [4], the complexity of the calibration problem when using 3D LIDAR measurements requires making certain approximations (see [4] for more details) which are not necessary when using a 2D LIDAR.



Fig. 1. The geometric constraint of the *i*-th measurement between the camera's frame of reference $\{C_i\}$, the LIDAR's frame of reference $\{L_i\}$, and the calibration board.

tively. Finally, Section VII provides concluding remarks and potential directions of future research.

II. PROBLEM FORMULATION

Consider a rigidly connected line scan LIDAR and camera pair. Let $\{L\}$ and $\{C\}$ denote the LIDAR's and camera's frames of reference, respectively. Our objective is to find the LIDAR's orientation, ${}^{C}\mathbf{R}_{L}$, and position, ${}^{C}\mathbf{t}_{L}$, in the camera's frame of reference. The line scan LIDAR measures the range to objects within its scanning plane. Without loss of generality, we define this scanning plane as the x - y plane with the LIDAR's center as the origin. Thus, the LIDAR measured points can be written as ${}^{L}\mathbf{p}_{i} = \begin{bmatrix} x_{i} & y_{i} & 0 \end{bmatrix}^{T}$.

To build correspondences between the camera and LIDAR measurements, we use a white rectangular calibration board with a black line, l, in the middle (see Fig. 1). From the camera image, the line l in the calibration board can be extracted easily with edge detection followed by a line fitting algorithm. Then, the vector ${}^{C}\mathbf{n}_{i}$ perpendicular to the plane defined by the line l and the camera center, expressed in the camera's frame of reference, can be determined.

On the other hand, the line scan LIDAR can detect the edge ending points of the calibration board, ${}^{\scriptscriptstyle L}\mathbf{p}_{i1}$ and ${}^{L}\mathbf{p}_{i2}$, based on depth discontinuity. Thus, the point ${}^{L}\mathbf{p}_{i}$ on the middle line in the LIDAR's frame of reference can be computed by averaging the two points. Clearly, by transforming the point ${}^{L}\mathbf{p}_{i}$ to the camera's frame of reference ${}^{C}\mathbf{p}_{i} = {}^{C}\mathbf{R}_{L}{}^{L}\mathbf{p}_{i} + {}^{C}\mathbf{t}_{L}$, we have the following geometric constraint:

$${}^{\scriptscriptstyle C}\mathbf{n}_i^{{}^{\scriptscriptstyle T}{\scriptscriptstyle C}}\mathbf{p}_i = {}^{\scriptscriptstyle C}\mathbf{n}_i^{{}^{\scriptscriptstyle T}}({}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}{}^{\scriptscriptstyle L}\mathbf{p}_i + {}^{\scriptscriptstyle C}\mathbf{t}_{\scriptscriptstyle L}) = 0$$
(1)

Note that in contrast to [7], we do not utilize the LIDAR's reflection intensity in our calibration process.

In practice, the geometric constraint (1) does not hold exactly due to measurement noise. Instead, it equals to a nonzero value denoted by e_i . To find the best estimate of the 6 dof LIDAR-camera transformation, we formulate the following least-squares problem which minimizes the sum of the residuals from all the available m measurements obtained as the LIDAR-camera sensor pair moves in front of the

calibration board:

$${}^{C}\mathbf{t}_{L}^{*}, {}^{C}\mathbf{R}_{L}^{*} = \operatorname*{argmin}_{{}^{C}\mathbf{t}_{L}, {}^{C}\mathbf{R}_{L}} \sum_{i=1}^{m} (e_{i})^{2}$$
(2)

$$\Rightarrow^{C} \mathbf{t}_{L}^{*}, {}^{C} \mathbf{R}_{L}^{*} = \operatorname*{argmin}_{C \mathbf{t}_{L}, {}^{C} \mathbf{R}_{L}} \sum_{i=1}^{m} ({}^{C} \mathbf{n}_{i}{}^{T} ({}^{C} \mathbf{R}_{L}{}^{L} \mathbf{p}_{i} + {}^{C} \mathbf{t}_{L}))^{2} \quad (3)$$

s. t. ${}^{C} \mathbf{R}_{L}{}^{T_{C}} \mathbf{R}_{L} = \mathbf{I}, \quad det({}^{C} \mathbf{R}_{L}) = 1$

where we have added the rotational matrix constraints.

One standard method to solve this problem is to use iterative algorithms, such as Gauss-Newton [9]. However, these methods converge to a global optimum only if the initialization points lie within the basin of attraction of the global minimum. In the following, we will prove that this problem is equivalent to another optimization problem, whose optimality conditions can be solved analytically.

III. ANALYTICAL DETERMINATION OF LIDAR-CAMERA EXTRINSIC CALIBRATION

We hereafter present our analytical solution for (3). Specifically, we first convert (3) to an equivalent problem which has the same solution but fewer variables and constraints. Then, we solve its optimality conditions using algebraic geometry techniques.

A. Transformation of the Optimization Problem

Note that in (3), the translation ${}^{C}\mathbf{t}_{L}$ is not involved in the constraints. Thus, the optimality condition with respect to ${}^{C}\mathbf{t}_{L}$ only involves ${}^{C}\mathbf{t}_{L}$ and the rotation matrix ${}^{C}\mathbf{R}_{L}$. Therefore, if we can express ${}^{C}\mathbf{t}_{L}$ in terms of ${}^{C}\mathbf{R}_{L}$, ${}^{C}\mathbf{t}_{L}$ can be eliminated from the optimization problem (3), thus reducing the number of variables.

Lemma 1: Solving the optimization problem (3) is equivalent to solving the following two problems sequentially:

$${}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}^{*} = \operatorname*{argmin}_{{}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}} \sum_{i=1}^{m} \left({}^{\scriptscriptstyle C}\mathbf{n}_{i}^{{}^{\scriptscriptstyle T}{}^{\scriptscriptstyle C}}\mathbf{R}_{\scriptscriptstyle L}{}^{\scriptscriptstyle L}\mathbf{p}_{i} + \sum_{j=1}^{m} \mathbf{g}_{ij}^{{}^{\scriptscriptstyle T}{}^{\scriptscriptstyle C}}\mathbf{R}_{\scriptscriptstyle L}{}^{\scriptscriptstyle L}\mathbf{p}_{j} \right)^{2}$$

s. t. ${}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}{}^{{}^{\scriptscriptstyle T}{}^{\scriptscriptstyle C}}\mathbf{R}_{\scriptscriptstyle L} = \mathbf{I}, \quad det({}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}) = 1$
(4)

$${}^{\scriptscriptstyle C}\mathbf{t}_{\scriptscriptstyle L}^* = -\left(\sum_{i=1}^{m} {}^{\scriptscriptstyle C}\mathbf{n}_{i}{}^{\scriptscriptstyle C}\mathbf{n}_{i}{}^{\scriptscriptstyle T}\right)^{-1} \left(\sum_{i=1}^{m} {}^{\scriptscriptstyle C}\mathbf{n}_{i}{}^{\scriptscriptstyle C}\mathbf{n}_{i}{}^{\scriptscriptstyle T_{\scriptscriptstyle C}}\mathbf{R}_{\scriptscriptstyle L}^{*\,{\scriptscriptstyle L}}\mathbf{p}_{i}\right)$$
(5)

where $\mathbf{g}_{ij} \triangleq {}^{C}\mathbf{n}_{i}{}^{T} \left(\sum_{i=1}^{m} {}^{C}\mathbf{n}_{i}{}^{C}\mathbf{n}_{i}{}^{T}\right)^{-1} {}^{C}\mathbf{n}_{j}{}^{C}\mathbf{n}_{j}{}^{T}$. *Proof:* Let J denote the cost function of (3). Its optimality

condition with respect to ${}^{C}\mathbf{t}_{L}$ is given by:

$$\frac{\partial J}{\partial^{C} \mathbf{t}_{L}} = \sum_{i=1}^{m} 2 \left[{}^{C} \mathbf{n}_{i}{}^{T} ({}^{C} \mathbf{R}_{L}{}^{L} \mathbf{p}_{i} + {}^{C} \mathbf{t}_{L}) \right] {}^{C} \mathbf{n}_{i} = \mathbf{0}$$

$$\Rightarrow^{C} \mathbf{t}_{L} = -\underbrace{\left(\sum_{i=1}^{m} {}^{C} \mathbf{n}_{i}{}^{C} \mathbf{n}_{i}{}^{T} \right)}_{\mathbf{Q}}^{-1} \left(\sum_{i=1}^{m} {}^{C} \mathbf{n}_{i}{}^{C} \mathbf{n}_{i}{}^{TC} \mathbf{R}_{L}{}^{L} \mathbf{p}_{i} \right)$$
(6)

(0)

Substituting (6) in the cost function of (3), we have:

$$J = \sum_{i=1}^{m} \left[{}^{C} \mathbf{n}_{i} {}^{T_{C}} \mathbf{R}_{L} {}^{L} \mathbf{p}_{i} - {}^{C} \mathbf{n}_{i} {}^{T} \mathbf{Q}^{-1} \left(\sum_{j=1}^{m} {}^{C} \mathbf{n}_{j} {}^{C} \mathbf{n}_{j} {}^{T_{C}} \mathbf{R}_{L} {}^{L} \mathbf{p}_{j} \right) \right]^{2}$$
$$= \sum_{i=1}^{m} \left[{}^{C} \mathbf{n}_{i} {}^{T_{C}} \mathbf{R}_{L} {}^{L} \mathbf{p}_{i} - \left(\sum_{j=1}^{m} {}^{C} \mathbf{n}_{i} {}^{T} \mathbf{Q}^{-1_{C}} \mathbf{n}_{j} {}^{C} \mathbf{n}_{j} {}^{T_{C}} \mathbf{R}_{L} {}^{L} \mathbf{p}_{j} \right) \right]^{2}$$
$$= \sum_{i=1}^{m} \left({}^{C} \mathbf{n}_{i} {}^{T_{C}} \mathbf{R}_{L} {}^{L} \mathbf{p}_{i} + \sum_{j=1}^{m} \mathbf{g}_{ij} {}^{T_{C}} \mathbf{R}_{L} {}^{L} \mathbf{p}_{j} \right)^{2}$$
(7)

Minimizing (7) with respect to ${}^{C}\mathbf{R}_{L}$ yields ${}^{C}\mathbf{R}_{L}^{*}$, and substituting ${}^{C}\mathbf{R}_{L}^{*}$ into (6) results in (5). This completes the proof.

To further simplify problem (4), we choose to use the quaternion representation for the rotation matrix ${}^{C}\mathbf{R}_{L}$, instead of the Cayley-Gibbs-Rodriguez parameterization [10] as in [4], which has a singular configuration when the rotation angle is π . For more details, we refer the interested reader to [10]. In particular, problem (4) is simplified by application of the following Lemma:

Lemma 2: Using the quaternion representation ${}_{L}^{C}\mathbf{q}$ for the rotation matrix ${}^{C}\mathbf{R}_{L}$, problem (4) is equivalent to:

$${}^{C}_{L} \mathbf{q}^{*} = \operatorname*{argmin}_{L} \sum_{i=1}^{m} \left({}^{C}_{L} \mathbf{q}^{T} \mathbf{M}_{i_{L}} {}^{C}_{L} \mathbf{q} \right)^{2}$$
(8)
s. t.
$${}^{C}_{L} \mathbf{q}^{T}_{L} {}^{C}_{L} \mathbf{q} = 1$$

where

$$\mathbf{M}_{i} \triangleq \mathcal{L}({}^{\scriptscriptstyle C}\bar{\mathbf{n}}_{i})^{\scriptscriptstyle T} \mathcal{R}({}^{\scriptscriptstyle L}\bar{\mathbf{p}}_{i}) + \sum_{j=1}^{m} \mathcal{L}(\bar{\mathbf{g}}_{ij})^{\scriptscriptstyle T} \mathcal{R}({}^{\scriptscriptstyle L}\bar{\mathbf{p}}_{j}) \qquad (9)$$

In the above expression, $\bar{\mathbf{x}}$ denotes the quaternion form of a vector \mathbf{x} , while $\mathcal{L}(\cdot)$ and $\mathcal{R}(\cdot)$ are left and right quaternion multiplication matrices (see Appendix I).

Proof: Using quaternion parameterization, the cost function of (4) can be written as:

$$J = \sum_{i=1}^{m} \left({}^{C} \bar{\mathbf{n}}_{i}^{T} ({}^{C}_{L} \mathbf{q} \otimes {}^{L} \bar{\mathbf{p}}_{i} \otimes {}^{C}_{L} \mathbf{q}^{-1}) + \sum_{j=1}^{m} \bar{\mathbf{g}}_{ij}^{T} ({}^{C}_{L} \mathbf{q} \otimes {}^{L} \bar{\mathbf{p}}_{j} \otimes {}^{C}_{L} \mathbf{q}^{-1}) \right)^{2}$$

$$= \sum_{i=1}^{m} \left({}^{C} \bar{\mathbf{n}}_{i}^{T} \mathcal{R} ({}^{C}_{L} \mathbf{q})^{T} \mathcal{L} ({}^{C}_{L} \mathbf{q})^{L} \bar{\mathbf{p}}_{i} + \sum_{j=1}^{m} \bar{\mathbf{g}}_{ij}^{T} \mathcal{R} ({}^{C}_{L} \mathbf{q})^{T} \mathcal{L} ({}^{C}_{L} \mathbf{q})^{L} \bar{\mathbf{p}}_{j} \right)^{2}$$

$$= \sum_{i=1}^{m} \left({}^{C} \mathbf{q}^{T} \mathcal{L} ({}^{C} \bar{\mathbf{n}}_{i})^{T} \mathcal{R} ({}^{L} \bar{\mathbf{p}}_{i}) {}^{C}_{L} \mathbf{q} + \sum_{j=1}^{m} {}^{C}_{L} \mathbf{q}^{T} \mathcal{L} (\bar{\mathbf{g}}_{ij})^{T} \mathcal{R} ({}^{L} \bar{\mathbf{p}}_{j}) {}^{C}_{L} \mathbf{q} \right)^{2}$$

$$= \sum_{i=1}^{m} \left[{}^{C}_{L} \mathbf{q}^{T} \left(\mathcal{L} ({}^{C} \bar{\mathbf{n}}_{i})^{T} \mathcal{R} ({}^{L} \bar{\mathbf{p}}_{i}) + \sum_{j=1}^{m} \mathcal{L} (\bar{\mathbf{g}}_{ij})^{T} \mathcal{R} ({}^{L} \bar{\mathbf{p}}_{j}) \right) {}^{C}_{L} \mathbf{q} \right]^{2}$$
(10)

By including the quaternion unit-norm constraint, we complete the proof. \blacksquare

It is important to note that (8) has only one quadratic constraint in four variables, which is easier to solve, as compared to the original optimization problem (3) that has six quadratic constraints and one cubic constraint in twelve variables. Using the Lagrange multiplier theorem, the Karush-Kuhn-Tucker (KKT) conditions [9] of (8) result in the following equations:

$$\begin{cases} \sum_{i=1}^{m} {\binom{c}{L} \mathbf{q}^{T} \mathbf{M}_{iL}^{C} \mathbf{q}} \left(\mathbf{M}_{i} + \mathbf{M}_{i}^{T} \right)_{L}^{C} \mathbf{q} + \lambda_{L}^{C} \mathbf{q} = \mathbf{0} \\ {}_{L}^{C} \mathbf{q}^{T} {}_{L}^{C} \mathbf{q} - 1 = 0 \end{cases}$$
(11)

where λ is the Lagrange multiplier. Note that (11) consists of four cubic polynomials and one quadratic polynomial in five variables. The solution of (11) can be computed using the eigenvalue decomposition of the so-called multiplication matrix, which we explain in the next section.

B. Analytical Solution

We now describe our analytical approach to directly solve the polynomial equations (11) using an algebraic-geometry technique that involves the multiplication matrix. The multiplication matrix is the generalization of the companion matrix adopted from univariate to multivariate polynomial systems [11]. The roots of a multivariate polynomial system can be computed from the eigenvector of the associated multiplication matrix. In the following, we briefly explain the procedure of constructing the multiplication matrix. The interested reader is referred to [12] for a thorough presentation of this method.

Any polynomial equation of order d_i can be written as $f_i = \mathbf{c}_i^T \mathbf{x}_{d_i}$, where \mathbf{x}_{d_i} is the vector of all the monomials up to order d_i and \mathbf{c}_i is the vector of coefficients. For a polynomial system with n equations $f_i = \mathbf{c}_i^T \mathbf{x}_{d_i}, i = 1, ..., n$, by stacking all the coefficient vectors \mathbf{c}_i into a matrix \mathbf{C} , the polynomial system can be expressed as $\mathbf{C}\mathbf{x}_{d_{max}} = \mathbf{0}$ where d_{max} is the highest order among all the d_i .

A polynomial system defines an ideal I, which is spanned by its Gröbner basis, $\mathbf{G} \triangleq \langle g_1, \ldots, g_t \rangle$, with two properties: (i) The remainder of any polynomial divided by \mathbf{G} is unique; (ii) Any polynomial whose remainder divided by \mathbf{G} equals zero, is a member of this ideal I. Based on the first property, for any polynomial $\phi(\mathbf{x})$ we have $\phi(\mathbf{x}) = r(\mathbf{x}) + \sum_{i=1}^{t} g_i h_i(\mathbf{x})$, where $h_i(\mathbf{x})$ is a polynomial of \mathbf{x} and it is called the quotient polynomial. Furthermore, the remainder $r(\mathbf{x})$ can be expressed as a linear combination of a group of monomials, the normal set \mathbf{x}_B , which can also be determined from the Gröbner basis. Therefore, multiplying any polynomial $\phi(\mathbf{x})$ with the normal set \mathbf{x}_B yields:

$$\phi(\mathbf{x}) \cdot \mathbf{x}_B = \mathbf{M}_{\phi} \mathbf{x}_B + \begin{bmatrix} h_{11} \cdots h_{1t} \\ \vdots & \vdots \\ h_{s1} \cdots h_{st} \end{bmatrix} \begin{bmatrix} g_1 \\ \vdots \\ g_t \end{bmatrix}$$
(12)

where \mathbf{M}_{ϕ} is the so-called multiplication matrix associated with the polynomial $\phi(\mathbf{x})$ determined by \mathbf{x}_B , s is the cardinality of the normal set and h_{ij} are polynomials in \mathbf{x} . By evaluating (12) at the polynomial's roots, we have $\phi(\mathbf{x}) \cdot \mathbf{x}_B = \mathbf{M}_{\phi} \mathbf{x}_B$, since $g_i = 0, i = 1, \dots, t$. Therefore, if we define $\phi(\mathbf{x})$ as one of the unknown variables \mathbf{x}_i , it becomes one of the eigenvalues of the multiplication matrix \mathbf{M}_{ϕ} . Furthermore, \mathbf{x}_B may contain a number of monomials of all the unknown variables \mathbf{x}_i , in which case the roots \mathbf{x}_i may be directly obtained from the eigenvector of the multiplication matrix.

Thus far, we have shown that given the Gröbner basis, the polynomial system's roots can be computed by the eigenvalue decomposition of the multiplication matrix. However, the Gröbner basis is not always available for polynomial systems. For a polynomial system with integer coefficients, the Gröbner basis can be computed using Buchberger's algorithm [12]. However this is not the case for polynomial systems with floating-point coefficients, because of the round-off error in iterative computations. Since, in practice, the LIDAR and camera measurements are not guaranteed to be integers, we employ the method proposed in [13] to compute the multiplication matrix \mathbf{M}_{ϕ} .

Notice that $\phi(\mathbf{x}) \cdot \mathbf{x}_B$ is the linear combination of both the monomials included in \mathbf{x}_B and others not included in \mathbf{x}_B , which we can denote as \mathbf{x}_R and rewrite (12) when evaluated at the solution of the system as:

$$\phi(\mathbf{x}) \cdot \mathbf{x}_B = \mathbf{M}'_{\phi} \begin{bmatrix} \mathbf{x}_R \\ \mathbf{x}_B \end{bmatrix}$$
(13)

where \mathbf{M}'_{ϕ} is the so-called unreduced multiplication matrix. If \mathbf{x}_R can be expressed as a linear combination of monomials in \mathbf{x}_B , $\mathbf{x}_R = \mathbf{H}\mathbf{x}_B$, then (13) can be written as:

$$\phi(\mathbf{x}) \cdot \mathbf{x}_B = \mathbf{M}'_{\phi} \begin{bmatrix} \mathbf{H} \\ \mathbf{I} \end{bmatrix} \mathbf{x}_B = \mathbf{M}_{\phi} \mathbf{x}_B$$
(14)

and \mathbf{x}_B becomes the eigenvector of the matrix \mathbf{M}_{ϕ} .

To do so, we collect all the polynomials up to order l from the product of f_i with all the other monomials, and stack them in a matrix to form the extended system:

$$\mathbf{C}_{e}\mathbf{x}_{l} = \begin{bmatrix} \mathbf{C}_{E} & \mathbf{C}_{R} & \mathbf{C}_{B} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{E} \\ \mathbf{x}_{R} \\ \mathbf{x}_{B} \end{bmatrix} = \mathbf{0}$$
(15)

where \mathbf{x}_E are the monomials not included in \mathbf{x}_R and \mathbf{x}_B . Defining the left null space of \mathbf{C}_E as \mathbf{N}^T , and multiplying it to the left side of (15), yields:

$$\begin{bmatrix} \mathbf{N}^T \mathbf{C}_R & \mathbf{N}^T \mathbf{C}_B \end{bmatrix} \begin{bmatrix} \mathbf{x}_R \\ \mathbf{x}_B \end{bmatrix} = \mathbf{0}$$
(16)

To express \mathbf{x}_R with respect to \mathbf{x}_B , we only need to perform QR decomposition of $\mathbf{N}^T \mathbf{C}_R = \mathbf{Q}\mathbf{R} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_1 \mathbf{R}_1$. If we select *l* large enough, \mathbf{R}_1 will be a full-rank matrix [14], in which case $\mathbf{x}_R = -\mathbf{R}_1^{-1}\mathbf{Q}_1^T\mathbf{N}^T\mathbf{C}_B\mathbf{x}_B \triangleq \mathbf{H}\mathbf{x}_B$. Then, \mathbf{x}_B in (14) is obtained by the eigenvalue decomposition of \mathbf{M}_{ϕ} .

In particular, for solving our LIDAR-camera calibration problem, \mathbf{R}_1 becomes full rank when l = 11. Specifically, the size of \mathbf{x}_E , \mathbf{x}_R and \mathbf{x}_B reaches 6045, 51 and 80 respectively, and the matrix \mathbf{C}_e is expanded to have 11011 rows. The dominant computation to solve the problem is determining the nullspace of \mathbf{C}_E , which has cost $\mathcal{O}(11011 * 6045^2)$. Note that the number of available measurements does not affect the order or number of unknown variables in (11), and thus it has barely any impact on the computational cost. Finally, we point out that the used normal set \mathbf{x}_B is not computed from the Gröbner basis, but by a numerical method (details of implementation are given in [13] and [14]). Once ${}_L^C \mathbf{q}$ is determined, the translation ${}^C \mathbf{t}_L$ can be computed from (5).

IV. OBSERVABILITY ANALYSIS

In this section, we analyse the line scan LIDAR-camera calibration system's observability properties and present the

conditions under which the LIDAR-camera transformation can be estimated. To do so, in the following, we identify the cases when the system has infinite number of solutions because of the LIDAR measurements (*Case 1*) or the camera measurements (*Cases 2-3*).

Case 1: Suppose all the LIDAR measurements ${}^{L}\mathbf{p}_{i}, i = 1, \ldots, m$, are parallel. Then, for any matrix **R** representing rotations around axis ${}^{L}\mathbf{p}_{i}$, we have ${}^{L}\mathbf{p}_{i} = \mathbf{R}^{L}\mathbf{p}_{i}$ and thus the geometric constraint (1) can be written as:

$${}^{\scriptscriptstyle C}\mathbf{n}_i^{\scriptscriptstyle T}({}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}{}^{\scriptscriptstyle L}\mathbf{p}_i + {}^{\scriptscriptstyle C}\mathbf{t}_{\scriptscriptstyle L}) = {}^{\scriptscriptstyle C}\mathbf{n}_i^{\scriptscriptstyle T}({}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}\mathbf{R}^{\scriptscriptstyle L}\mathbf{p}_i + \mathbf{n}) = 0$$

which means the rotation matrix ${}^{C}\mathbf{R}_{L}$ can be perturbed by any rotation matrix around ${}^{L}\mathbf{p}_{i}$ and the constraint will still hold (i.e., we have infinite solutions).

Case 2: Suppose the normal vectors ${}^{C}\mathbf{n}_{i}$, i = 1, ..., m, determined from the camera measurements are all parallel. Then, for any matrix **R** representing rotations around axis ${}^{C}\mathbf{n}_{i}$, ${}^{C}\mathbf{n}_{i} = \mathbf{R}^{C}\mathbf{n}_{i}$. Additionally, for any vector \mathbf{n}^{\perp} perpendicular to ${}^{C}\mathbf{n}_{i}$, ${}^{C}\mathbf{n}_{i}^{T}\mathbf{n}^{\perp} = 0$. Thus, (1) can be written as:

which means that ${}^{C}\mathbf{R}_{L}$ and ${}^{C}\mathbf{t}_{L}$ can be perturbed by any rotation matrix \mathbf{R} around axis ${}^{C}\mathbf{n}_{i}$, and ${}^{C}\mathbf{t}_{L}$ can also be perturbed by any vector perpendicular to the normal vector ${}^{C}\mathbf{n}_{i}$ and the constraint will still hold.

Case 3: Suppose the normal vectors determined from the camera measurements all lie in one plane spanned by two non-parallel ${}^{c}\mathbf{n}_{1}$ and ${}^{c}\mathbf{n}_{2}$. Define a vector \mathbf{n}^{\perp} perpendicular to both ${}^{c}\mathbf{n}_{1}$ and ${}^{c}\mathbf{n}_{2}$, i.e., $\mathbf{n}^{\perp} = {}^{c}\mathbf{n}_{1} \times {}^{c}\mathbf{n}_{2}$, then ${}^{c}\mathbf{n}_{1}^{T}\mathbf{n}^{\perp} = {}^{c}\mathbf{n}_{2}^{T}\mathbf{n}^{\perp} = 0$ and (1) can be written as:

$${}^{\scriptscriptstyle C}\mathbf{n}_i^{\scriptscriptstyle T}({}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}{}^{\scriptscriptstyle L}\mathbf{p}_i + {}^{\scriptscriptstyle C}\mathbf{t}_{\scriptscriptstyle L}) = {}^{\scriptscriptstyle C}\mathbf{n}_i^{\scriptscriptstyle T}({}^{\scriptscriptstyle C}\mathbf{R}_{\scriptscriptstyle L}{}^{\scriptscriptstyle L}\mathbf{p}_i + ({}^{\scriptscriptstyle C}\mathbf{t}_{\scriptscriptstyle L} + \mathbf{n}^{\perp})) = 0$$

which means that ${}^{C}\mathbf{t}_{L}$ can be perturbed by any vector perpendicular to the plane and the constraint will still hold.

Case 4: Suppose the three normal vectors determined from the camera measurements span the whole 3D space, and the LIDAR measurements are not all parallel. Then, as shown in [4] the polynomial system has up to eight solutions which can be computed in closed-form. Mathematically, the observability analysis described here is close to that in [4], but with very different physical interpretations due to the different geometric configurations.

Based on the above analysis, and since there is a low probability that all the LIDAR measurements will be parallel, in practice it is the configuration of the normal vectors from the camera measurements that plays a key role in determining the problem's observability properties. To obtain a well-conditioned measurement set, it is required to rotate the calibration board in front of the LIDAR-camera platform so that ${}^{C}\mathbf{n}_{i}$'s span all three directions.

V. SIMULATION RESULTS

To validate our proposed algorithm, extensive simulations have been conducted, in each of which 1000 Monte-Carlo trials are performed. The rotation angle between the line



Fig. 2. The histogram for the estimated rotation and translation error for noise-free measurements.

scan LIDAR and the camera is generated randomly from a uniform distribution $\mathcal{U}[0, 2\pi]$, and the rotation axis is generated as a random vector with normal distribution. The LIDAR measured points ${}^{L}\mathbf{p}_{i}$ are uniformly distributed within the range 50 - 150 cm. The corresponding camera captured vectors are selected randomly from the null space of the LIDAR measured points expressed in the camera's frame of reference ${}^{C}\mathbf{R}_{L}{}^{L}\mathbf{p}_{i} + {}^{C}\mathbf{t}_{L}$. For each Monte Carlo trial, 10 LIDAR and camera measurement pairs are generated.

In order to test the numerical stability of our algorithm, we first consider the case of noise-free measurements, because our method may obtain inaccurate or even incorrect estimates due to the numerical error in solving the polynomial system. The histograms of the Root Mean Square Error (RMSE) of the estimated rotation and translation are shown in Fig. 2. To find the rotation error, first we compute the error quaternion $\delta \mathbf{q} \approx {}_{L}^{C} \mathbf{q} \otimes {}_{L}^{C} \mathbf{\hat{q}}^{-1}$, where ${}_{L}^{C} \mathbf{\hat{q}}$ is our estimated quaternion and ${}_{L}^{C} \mathbf{q}$ is the true quantity. Then, we use the approximation $\delta \mathbf{q} \approx [\frac{1}{2} \delta \theta^{T} \ 1]^{T}$ to obtain $\delta \theta$ and compute its norm. Using as criterion $log_{10} || \delta \theta || > -2$, we determined the failure rate to be 0.5%.

Moreover, we also compared our proposed method to the minimal problem solver presented in [7], whose implementation is available at [15]. The minimal solver only requires 6 LIDAR-camera measurements, and thus we have many choices in selecting a subset from the 10 available LIDARcamera measurements. To do so, we employ RANSAC [8] and select the estimate that minimizes the cost function (3) as the resulting estimate from the minimal solver. To generate noise in the camera-measured normal vector, we perturb the true normal vector around a randomly-generated axis by an angle drawn from normal distribution. The RMSEs of the translation and rotation estimates obtained using our leastsquares algorithm and the minimal solver versus different noise levels in the LIDAR and camera measurements are plotted in Fig. 3. As evident, our algorithm significantly outperforms the minimal solver for all cases considered.



(b)

Fig. 3. The RMSE of the estimated LIDAR-camera transformation versus: (a) the standard deviation (std) of the noise in the LIDAR measurements; (b) the standard deviation (std) of the angle noise in the camera-measured normal vectors.

VI. EXPERIMENTAL RESULTS

To demonstrate the validity of our algorithm in practice, we tested it using real data. In our experiment, a HOKUYO UBG-04LX-F01 line scan LIDAR and a Chameleon CMLN-13S2M camera are rigidly mounted on the same platform. The LIDAR-camera pair and the calibration board used for calibration are shown in Fig. 4. The camera is intrinsically calibrated using the method of [16]. The accuracy of the LIDAR is ± 1 cm in the range 6 - 100 cm, has 1% error for ranges larger than 100 cm, while its angular resolution is 0.36 degree. The calibration board moves between 50 - 150cm in front of the LIDAR-camera pair, and 10 measurements from the LIDAR and camera are used in the calibration.

The computed transformation between the LIDAR and the camera is shown in Table I, where ρ denotes the Euler angles for rotation. Note that for evaluating the accuracy of the minimal solver, we estimate the LIDAR and camera transformation by randomly selecting 30 sets of measurements and



Fig. 4. (a) Line scan LIDAR-camera platform (b) Calibration board

TABLE I CALIBRATION RESULT

Least-Squares Solution			Minimal Problem Solution		
${}^{C}_{L}\mathbf{q}$	ρ (deg)	$^{C}\mathbf{t}_{L}$ (cm)	$^{C}_{L}\mathbf{q}$	ρ (deg)	$^{C}\mathbf{t}_{L}$ (cm)
0.0029	3.13	7.20	0.0095	-0.02	8.91
-0.0118	0.02	4.61	0.0055	-0.01	3.74
0.1004	-0.20	1.85	0.0917	-0.18	2.69
0.9949			0.9957		

keeping the one with the least cost (3). The cost function value is 3.16 when evaluated with our least-squares solution, and 4.62 with the minimal solver. As evident, our method attains a smaller least-squares error, and hence provides a more accurate least-squares solution.

VII. CONCLUSION AND FUTURE WORK

In this paper, we have presented an analytical least-squares solution for computing the line scan LIDAR-camera extrinsic calibration parameters. In particular, we have formulated this problem as a nonlinear least-squares minimization and shown that using an appropriate change of variables, its optimality conditions form a system of multivariate polynomial equations. Moreover, we have solved this system analytically, using techniques from algebraic geometry, and found its global minimum. Finally, we have identified under which conditions the unknown transformation can be recovered. As part of our future work, we plan to investigate the line scan LIDAR-camera extrinsic calibration problem in unknown environments.

APPENDIX I

Following the quaternion parameterization described in [17], $\mathbf{p}_2 = \mathbf{R}\mathbf{p}_1$ can be written as

$$\bar{\mathbf{p}}_2 = \mathbf{q} \otimes \bar{\mathbf{p}}_1 \otimes \mathbf{q}^{-1} \tag{17}$$

where \otimes represents quaternion multiplication, \mathbf{q}^{-1} is the quaternion's inverse defined as $\mathbf{q}^{-1} = \begin{bmatrix} -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix}^T$, and $\mathbf{\bar{p}}_i = \begin{bmatrix} \mathbf{p}_i^T & 0 \end{bmatrix}^T$, i = 1, 2, is the quaternion form of \mathbf{p}_i , which is not necessary of unit norm, but can be manipulated with quaternion operations.

For any quaternions q_1 and q_2 , their product, $q_1 \otimes q_2$, is defined as:

$$\mathbf{q}_1 \otimes \mathbf{q}_2 \triangleq \mathcal{L}(\mathbf{q}_1) \mathbf{q}_2 = \mathcal{R}(\mathbf{q}_2) \mathbf{q}_1 \tag{18}$$

where

$$\mathcal{L}(\mathbf{q}) \triangleq \begin{bmatrix} q_4 & -q_3 & q_2 & q_1 \\ q_3 & q_4 & -q_1 & q_2 \\ -q_2 & q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix} \qquad \mathcal{R}(\mathbf{q}) \triangleq \begin{bmatrix} q_4 & q_3 & -q_2 & q_1 \\ -q_3 & q_4 & q_1 & q_2 \\ q_2 & -q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix}$$

$$\mathcal{L}(\mathbf{q}^{-1}) = \mathcal{L}(\mathbf{q})^T$$
 $\mathcal{R}(\mathbf{q}^{-1}) = \mathcal{R}(\mathbf{q})^T$

REFERENCES

- R. Unnikrishnan and M. Hebert, "Fast extrinsic calibration of a laser rangefinder to a camera," Robotics Institute, Carnegie Mellon University, Tech. Rep., July 2005.
- [2] A. Geiger, F. Moosmann, O. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *Proc. of the IEEE International Conference on Robotics and Automation*, Saint Paul, Minnesota, May 14–18 2012, pp. 3936–3943.
- [3] J. Levinson and S. Thrun, "Automatic calibration of cameras and lasers in arbitrary scenes," in *Proc. of the International Symposium* on *Experimental Robotics*, Quebec City, Canada, June 17–21 2012, pp. 1–6.
- [4] F. M. Mirzaei, D. G. Kottas, and S. I. Roumeliotis, "3d lidar-camera intrinsic and extrinsic calibration: Observability analysis and analytical least squares-based initialization," *International Journal of Robotics Research*, vol. 31, no. 4, pp. 452–467, 2012.
- [5] Q. Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, Sept. 28 – Oct. 2 2004, pp. 2301 – 2306.
- [6] R. M. Haralick, C. N. Lee, K. Ottenberg, and M. Nolle, "Review and analysis of solutions of the three point perspective pose estimation problem," *International Journal of Computer Vision*, vol. 13, no. 3, pp. 331–356, Dec. 1994.
- [7] O. Naroditsky, A. Patterson, and K. Daniilidis, "Automatic alignment of a camera with a line scan lidar system," in *Proc. of the IEEE International Conference on Robotics and Automation*, Shanghai, China, May 9–13 2011, pp. 3429 – 3434.
- [8] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, June 1981.
- [9] D. P. Bertsekas, Nonlinear Programming. Athena Scientific, 1999.
- [10] M. D. Shuster, "A survey of attitude representations," Astronautical Sciences, vol. 41, no. 4, pp. 439–517, 1993.
- [11] W. Auzinger and H. J. Stetter, "An elimination algorithm for the computation of all zeros of a system of multivariate polynomial equations," in *Proc. of the International Conference on Numer. Math*, vol. 86, Singapore, 1988, pp. 11–30.
- [12] D. A. Cox, J. B. Little, and D. O'Shea, *Using Algebraic Geometry*. Springer, 2005.
- [13] M. Byrod and K. Josephson, "A column-pivoting based strategy for monomial ordering in numerical grobner basis calculations," in *Proc.* of the European Conference of Computer Vision, Marseile, France, Oct. 12–18 2008, pp. 130–143.
- [14] G. Reid and L. Zhi, "Solving polynomial systems via symbolicnumeric reduction to geometric involutive form," *Journal of Symbolic Computation*, vol. 44, no. 3, pp. 280–291, May 2009.
- [15] [Online]. Available: http://www.seas.upenn.edu/~narodits/
- [16] J. Y. Bouguet. (2006) Camera calibration toolbox for matlab. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calibdoc/
- [17] N. Trawny and S. I. Roumeliotis, "Indirect kalman filter for 3d attitude estimation," University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep., March 2005. [Online]. Available: http://www-users.cs.umn.edu/~trawny/publications.htm