Report Number: WUCSE-2003-59

2003-08-11

Extrinsic Auto-calibration of a Camera and Laser Range Finder

Robert Pless and Qilong Zhang

This paper describes theoretical and experimental results for the auto-calibration of sensor platform consisting of a camera and a laser range finder. Real-world use of autonomous sensor platforms often requires the recalibration of sensors without an explicit calibration object. The constraints are based upon data captured simultaneously from the camera and the laser range finder while the sensor plat-form undergoes an arbitrary motion. The rigid motions of both sensors are related, so these data constrain the relative position and orientation of the camera and laser range finder. We introduce the mathematical constraints for auto-calibration techniques based upon both discrete... Read complete abstract on page 2.

Extrinsic Auto-calibration of a Camera and Laser Range Finder

Robert Pless and Qilong Zhang

Complete Abstract:

This paper describes theoretical and experimental results for the auto-calibration of sensor platform consisting of a camera and a laser range finder. Real-world use of autonomous sensor platforms often requires the recalibration of sensors without an explicit calibration object. The constraints are based upon data captured simultaneously from the camera and the laser range finder while the sensor plat-form undergoes an arbitrary motion. The rigid motions of both sensors are related, so these data constrain the relative position and orientation of the camera and laser range finder. We introduce the mathematical constraints for auto-calibration techniques based upon both discrete and differential motions, and present simulated experimental results, and results from a implementation on a B21rT M Mobile Robot from iRobot Corporation. This framework could also encompass extrinsic calibration with GPS, inertial, infrared, and ultrasonic sensors.

2003-59

Extrinsic Auto-calibration of a Camera and Laser Range Finder

Authors: Pless, Pless; Zhang, Qilong

August 11, 2003

Abstract: This paper describes theoretical and experimental results for the auto-calibration of sensor platform consisting of a camera and a laser range finder. Real-world use of autonomous sensor platforms often requires the re-calibration of sensors without an explicit calibration object. The constraints are based upon data captured simultaneously from the camera and the laser range finder while the sensor platform undergoes an arbitrary motion. The rigid motions of both sensors are related, so these data constrain the relative position and orientation of the camera and laser range finder. We introduce the mathematical constraints for auto-calibration techniques based upon both discrete and differential motions, and present simulated experimental results, and results from a implementation on a B21r^{TM} Mobile Robot from iRobot Corporation. This framework could also encompass extrinsic calibration with GPS, inertial, infrared, and ultrasonic sensors.

Extrinsic Auto-calibration of a Camera and Laser Range Finder

Robert Pless Qilong Zhang
Department of Computer Science & Engineering
Washington University in St. Louis

Abstract

This paper describes theoretical and experimental results for the auto-calibration of sensor platform consisting of a camera and a laser range finder. Real-world use of autonomous sensor platforms often requires the re-calibration of sensors without an explicit calibration object. The constraints are based upon data captured simultaneously from the camera and the laser range finder while the sensor platform undergoes an arbitrary motion. The rigid motions of both sensors are related, so these data constrain the relative position and orientation of the camera and laser range finder. We introduce the mathematical constraints for auto-calibration techniques based upon both discrete and differential motions, and present simulated experimental results, and results from a implementation on a B21r^{TM} Mobile Robot from iRobot Corporation. This framework could also encompass extrinsic calibration with GPS, inertial, infrared, and ultrasonic sensors.

Figure 1. A B21r^{TM} Mobile Robot from iRobot Corporation can be configured to have a video camera (at the top) and a planar laser range finder which measures distances to points lying on a plane about 18 inches off the floor. The goal of this paper is to study auto-calibration methods that find the rotation Φ and the translation Δ which transform points in the laser coordinate system to points in the normalized camera coordinate system — by considering data captured simultaneously from both sensors when the robot is in motion.

1 Introduction

Fusing data captured by multiple sensors is important for many robotic tasks. For sensors such as video cameras, laser range finders, ultrasound sensor and GPS, the position and orientation of the sensor affects the geometric interpretation of its measurements. In order to effectively use the data from all these sensors, it is important to know the relative position of each from each other, or of each from a fiducial coordinate system.

The calibration of each of these geometric sensors can be decomposed into internal calibration parameters and external parameters. The external calibration parameters are the position and orientation of the sensor relative to some fiducial coordinate system. The internal parameters, such as the calibration matrix of a camera, affect how the sensor samples the scene. We concentrate only on finding the external calibration parameters because for many sensors there already exists self-calibration techniques, for cameras [2, 5, 8, 18], for optical and magnetic 6 DOF-sensors [7], and for other sensors such as an electronic compass [9]. It is both possible and often beneficial to simultaneously estimate both the intrinsic and extrinsic parameters of a sensor, but in this work we assume that the intrinsic parameters of each sensor are known.

The contribution of this paper is the description of a theory and implementation of a method to find the relative position and orientation of a camera and a laser range finder. To our knowledge this is the first paper to discuss this auto-calibration problem. A companion paper presents calibration algorithms for the explicit calibration of a camera and laser range finder, this is used as a baseline against which the auto-calibration errors are calculated. A calibrated sys-

tem of this sort can be used to simulate a focal plane array which captures an image of a scene and overlays depth information for parts of the scene imaged by the laser (see Figure 5). This type of focal plane array has enormous advantages for many navigation problems including motion and structure estimation.

This work was inspired by two previous works. Determining the extrinsic geometric transformations between two cameras mounted on a rigidly moving object was discussed (but not implemented) in [1], and a theory and implementation for solving for the intrinsic and extrinsic parameters were calculated in [19]. In both cases, it was not necessary that the cameras share a common or overlapping field of view, both methods consider the constraints generated because the cameras motions were forced to have a motions consistent with their fixed, relative position.

It is important also to differentiate this work from two other problems that at first may appear similar. There have been several proposed methods for auto-calibration of a stereo camera pair with points that are matched between both cameras in the pair, and between images from different positions of the camera pair [3, 10, 20]. This is a fundamentally different kind of constraint, and requires that the cameras have overlapping fields of view. There has also been a great deal of work on calibration for laser scanners. Finding the geometric relationship between the laser scanner and the camera is vital to creating metric depth estimates from the camera images, and auto-calibration methods exist for this problem as well [12]. Laser scanners are the parts of active vision systems which projects a point or a stripe which is then viewed by the camera, as opposed to a laser range finder which reports distances to objects that lie in particular directions.

The next section introduces notation used to represent the position of sensors relative to one another. Section 3 derives the coherent motion constraints that relate a rigid motion in one coordinate system to the same rigid motion in another coordinate system for both differential and discrete motions. Section 4 gives methods for solving for the extrinsic calibration, first showing a method for the implausible case of two sensors which each can accurately determine their motion (in their own coordinate system), and then showing more realistic methods to calibrate a camera relative to the coordinate system of a laser range finder. We conclude by giving experimental results showing the success and failure modes of the techniques presented.

2 Background

This paper is aimed at solving constraints to relate the position of multiple sensors relative to each other. An equivalent problem — and easier to define — is to relate the position of each sensor relative to some fiducial coordinate system. The fiducial coordinate system defines an origin, (the point (0,0,0)), and the X,Y, Z axes. For the remainder of this paper, it is assumed that the sensors are always fixed relative to the fiducial coordinate system (which may be undergoing a rigid transformation).

Each sensor also has a local coordinate system. How this local coordinate system is defined is specific to the type of sensor. A common example is a pinhole camera which defines a coordinate system fixed on the pinhole, with the Z-axis pointing along the optical axis, and X and Y axes dependent upon the intrinsic calibration of the camera.

The goal of this paper can then be stated as finding the transformation from each sensor coordinate system to the fiducial coordinate system. This coordinate system transformation is also a rigid transformation. That is, a point P_i in the coordinate system of sensor i is located at a point P_f in the fiducial coordinate system:

$$P_f = \Phi_i P_i + \Delta_i \tag{1}$$

where Φ_i is a 3x3 orthonormal matrix representing the rotation and Δ_i is an offset vector corresponding to the translation. In what follows, the variables Φ and Δ *always* correspond to transformations between coordinate systems of different sensors, while variables R, T correspond to the rigid motions of the system or of individual sensors. Our goal in this paper is to develop ways to solve for these transformation parameters Φ_i and Δ_i which define the position of sensors with respect to the fiducial coordinate system.

If two sensors are known relative to fiducial system, we can map between them directly. And a point P_i in the coordinate system of sensor i is located at a point $\Phi_j^{-1}(\Phi_i P_i + \Delta_i - \Delta_j)$ in coordinate system of sensor j.

3 Coherent Motion Constraints

Autonomous systems are being built with multiple sensors. If the sensor are rigidly attached to the system, the motion of these sensor is exactly constrained with respect to the motion of the fiducial coordinates of the system. Intuitively, the relationships between the local motion at each sensor constrains their relative position. In this section we write the relationships between the motion of a fiducial system and the motion experienced in the local coordinate system of a sensor. We do this for both differential and discrete system motions.

3.1 Differential Motion

Suppose the fiducial system is undergoing an instantaneous translation $\vec{t}_f$ and an instantaneous rotation $\vec{\omega}_f$. In the local coordinate system of sensor i, this creates an instantaneous translation $\vec{t}_i$:

$$\vec{t}_i = \Phi_i^{-1}(-\vec{t}_f - \Delta_i \times \vec{\omega}_f)$$

and an instantaneous rotation $\vec{\omega}_i$ in the local coordinate system:

$$\vec{\omega}_i = \Phi_i^{-1}\vec{\omega}_f$$

This gives constraints on Φ_i, Δ_i as a relationship between the local differential motion to fiducial differential motion.

3.2 Discrete Motion

Suppose the fiducial system is undergoes a discrete translation T_f and an instantaneous rotation R_f. This motion transforms a point P_f in the fiducial coordinate system as follows:

$$P'_f = R_f P_f + T_f \tag{2}$$

Relative to the coordinate system of sensor i, the motion of a point P_i can be written (by substituting into Equation 1):

$$\Phi_i P'_i + \Delta_i = R_f(\Phi_i P_i + \Delta_i) + T_f, \tag{3}$$

which can be transformed to give the position P'_i as a function of the fiducial motion and the position and orientation of sensor i:

$$\begin{aligned}
&\Phi_i P'_i + \Delta_i = R_f(\Phi_i P_i + \Delta_i) + T_f \\
&\Phi_i P'_i = R_f(\Phi_i P_i + \Delta_i) + T_f - \Delta_i \\
&P'_i = \Phi_i^{-1}\left(R_f(\Phi_i P_i + \Delta_i) + T_f - \Delta_i\right) \\
&P'_i = \Phi_i^{-1} R_f \Phi_i P_i + \Phi_i^{-1} R_f \Delta_i + \Phi_i^{-1}(T_f - \Delta_i)
\end{aligned} \tag{4}$$

Relative to the local coordinate system, each point is transformed as a rigid motion:

$$P'_i = R_i P_i + T_i. \tag{5}$$

Setting the right hand sides of Equations 4,5 to be equal, we can write:

$$\begin{aligned}
R_i &= \Phi_i^{-1} R_f \Phi_i \\
T_i &= \Phi_i^{-1} R_f \Delta_i + \Phi_i^{-1}(T_f - \Delta_i).
\end{aligned} \tag{6}$$

This gives a constraint on Φ, Δ as relationship between the discrete motion in the local coordinate system and the discrete motion in the fiducial coordinate system.

4 Estimation of Extrinsic Calibration

The previous section gives constraints relating the motion of each sensor to that of the fiducial system. Now we consider solving for Φ, and Δ using these constraints. As a warm up, we first consider sensors that are capable of estimating all parameters of motion in their local coordinate system. Then the coherent motion constraint allows the solution for the relative orientation and position of each sensor. We describe this process for two sensors for the discrete motion case, the differential case is slightly easier. For convenience, we choose as the fiducial coordinate system the coordinate system attached to one of the two sensors. Then the orientation and position of another sensor relative to the fiducial coordinate system are denoted by Φ and Δ respectively.

Suppose that this system undergoes several different motions. In each motion, both of the sensors can independently measure their rotation and translation exactly. Then for the jth motion, the rotation and translation of the fiducial system are R_f^j and T_f^j respectively, while those of the other one are R^j and T^j respectively. Then we can write out coherent motion constraints for the jth motion from Equation 6:

$$\begin{aligned}
R^j &= \Phi^{-1} R_f^j \Phi \\
T^j &= \Phi^{-1} R_f^j \Delta + \Phi^{-1}(T_f^j - \Delta)
\end{aligned} \tag{7}$$

A 3D rotation can be specified by an axis of rotation with a unit length vector $\vec{r}$, and an angle of rotation θ. Usually these two value are jointly specified as a rotation vector $\vec{\omega} = \vec{r}\theta$. So in this paper, we also use $\vec{\omega}$ to denote the rotation represented by the rotation matrix R. The conversion from vector $\vec{\omega}$ to matrix R is given by $Rodrigues\ formula$:

$$R = I + [\vec{r}]_\times \sin(\theta) + [\vec{r}]_\times^2 (1 - \cos(\theta))$$

In terms of rotation vectors, the rotation constraint of Equation (7) can be written as $\vec{\omega}_f^j = \Phi\vec{\omega}^j$, which can derive a constraint on the rotation axis as $\vec{r}_\Phi \cdot (\vec{\omega}_f^j - \vec{\omega}^j) = 0$, where $\vec{r}_\Phi$ denotes the unit vector representing the axis of rotation represented by Φ. Each motion j gives one constraint of this form, so we can solve for $\vec{r}_\Phi$ by minimizing,

$$\sum_j (\vec{r}_\Phi \cdot (\vec{\omega}_f^j - \vec{\omega}^j))^2 \tag{8}$$

Once the axis of rotation is known, the angle of rotation θ_Φ is can be estimated by minimizing:

$$\sum_j \|\vec{\omega_f}^j - \Phi\vec{\omega}^j\|^2, \tag{9}$$

where $\Phi = I + [\vec{r}_\Phi]_\times \sin(\theta_\Phi) + [\vec{r}_\Phi]_\times^2 (1 - \cos(\theta_\Phi))$, which is now a function only of the unknown rotation angle θ_Φ.

After the orientation of the sensor Φ is determined, we can continue to solve its relative position Δ. It can be also estimated by solving a least-square problem, which minimizes the discrepancy from the coherent motion constraint.

$$\sum_j (\Phi T^j - R_f^j \Delta - T_f^j + \Delta)^2 \tag{10}$$

This presentation, while unrealistic in its assumptions, sets up the following sections. Each section seeks to estimate Φ, and Δ, by minimizing a function that depends on the coherent motion constraint.

4.1 Using the Epi-Polar constraint

It is unreasonable to assume that each sensor accurately measures its own ego-motion (if this were the case, there would be less of a reason to use multiple sensors). For example, motion estimation from image data is known to have ambiguities confusing the estimation of rotation and translation. A more reasonable assumption is that a sensor can compute an error measure relating its measurements to a local motion estimate. An example is the epi-polar constraint, a relationship between camera motion and correspondence between image points.

The outline of the approach is the following. Initially, assume that the fiducial motion is known exactly. Then for a choice of Φ, and Δ, calculate the motion experienced in the camera coordinate system. This motion (like all rigid motions of a camera) defines an epi-polar constraint. We find corresponding points between two images taken by the camera, and calculate how well this epi-polar constraint fits the set of correspondences. This becomes an error for that choice of Φ and Δ.

More formally, let image I be the image before camera motion j, and I′ be the image after camera motion j. Suppose there are given m corresponding points between I and I' noted by $\mathbf{m}_k$, $\mathbf{m}'_k$ ($k = 1, .., m$), they satisfy the epi-polar constraint:

$$\mathbf{m'}^j_k F^j \mathbf{m}^j_k = 0 \; (k = 1, .., m) \tag{11}$$

where F^j is fundamental matrix for motion j, relating images I and I'.

Suppose camera motion j is described by camera rotation R^j and translation T^j, then the essential matrix E^j relating image I and I' is given by:

$$E^j = [T^j]_{\mathrm{x}} R^j$$

The fundamental matrix F^j can be decomposed as $F^j = K^{-\mathrm{T}} E^j K^{-1}$, where K is the intrinsic matrix for the camera. After some simple substitutions, we have the following constraint:

$$F^j = K^{-\mathrm{T}} [T^j]_{\mathrm{x}} R^j K^{-1} \tag{12}$$

We can write an expression for the fundamental matrix in terms of the motion in the fiducial coordinate system, by substituting Equation 6 into Equation 12:

$$F^j = K^{-\mathrm{T}} [R^j_f \Delta - \Delta + T^j_f]_{\mathrm{x}} R^j_f \Phi K^{-1} \tag{13}$$

This defines the fundamental matrix as a function of Φ and Δ, and we can write an error function for Φ and Δ directly in terms of the corresponding points. Then Φ and Δ can be estimated by solving a least-square problem, which minimizes the residual from the epi-polar constraints (for all points j in all frames k.) (Equation 11),

$$\sum_j \sum_k (\mathbf{m'}^j_k F^j \mathbf{m}^j_k)^2 \tag{14}$$

where F^j is expressed in terms of $\Phi, \Delta, R^j_f, T^j_f$ in Equation 13 [1]. This algebraic error, however, does not have direct interpretation in the measurement space, (i.e., it is not a distance in the image plane). So we rewrite the residual function in Equation 14 using the discrepancy in the epi-polar geometry,

$$\sum_j \sum_k (d^2(\mathbf{m'}^j_k, F^j \mathbf{m}^j_k) + d^2(\mathbf{m}^j_k, F^{j\mathbf{T}} \mathbf{m'}^j_k)) \tag{15}$$

Where $d^2(\cdot, \cdot)$is the squared distance between a point and a line. Since Euclidean distance in the metric space are used, the new criterion is more reasonable.

In our experiments we have used MATLAB's nonlinear optimization tools, either using Gauss-Newton method [4] with line-search for finding an unconstrained minimum of a sum of squares of nonlinear functions, or specifying the Levenberg-Marquardt method [13, 14, 15]. They demand the initial estimate of the orientation and position of the camera to be provided. If we have prior knowledge of cameras and the fiducial system, we can directly use it as the initial estimation.

4.1.1 Implementation

We implement the previous algorithm on a B21r^{TM} Mobile Robot from iRobot Corporation. A Sony DFW-VL500 camera is mounted on top of the robot, and the laser ranger finder calculates distances to points on a plane parallel to the floor. The center of the laser range finder is set as the fiducial coordinate system.

Estimating the Fiducial Motion from Laser Points. Initially, we implement an algorithm (a simplified version of [16]) to compute an estimate of the fiducial motion from the laser range finder data alone. In order to estimate R_f and T_f, we define an error measure by computing generalized Hausdorff Distance between the two sets of laser points generated before and after the fiducial motion. Given two sets of points $P = \{p_1, p_2, \ldots, p_{180}\}$ and $P' = \{p'_1, p'_2, \ldots, p'_{180}\}$, where P is the laser reading before fiducial motion, and P' is the laser reading after fiducial motion. The generalized Hausdorff measure is defined as

$$H_k(P, P') = \max(h_k(P, P'), h_k(P', P))$$

where

$$h_k(P, P') = \underset{p \in P}{kth} \min_{p' \in P'} \|p - p'\|$$

where kth denotes the k-th ranked value. In this paper, we are interested in using this generalized Hausdorff distance

[1] Note that although F^j changes based upon both the fiducial motion and the camera position, the estimate of Φ, Δ is constant over all images, so many system motions can simultaneously constrain the solution.

to measure how P' matches with P after a rigid transformation. In terms of fiducial motion R_f and T_f, the error approximation can be defined as:

$$\sum_{k=1}^{K} (H_k(R_f P + T_f, P') + H_k(R_f^{-1}(P' - T_f), P)) \quad (16)$$

where K is the parameter determining how many laser points are taken into account (We use a value of K = 160, which allows up to 20 range points to be outliers). The estimated fiducial motion parameters R_f and T_f are those which by minimizing above error function.

4.1.2 An Iterative Algorithm

As we can see, it is not possible to measure the fiducial motion parameters R_f and T_f precisely, which can introduce much noise during the estimation of Φ and Δ. A robust approach is to iteratively refine the fiducial motion parameters by considering Equation 16 during the estimation of Φ and Δ. Assume the camera's orientation Φ and position Δ are available, we can instead optimize our estimate of the fiducial motion, and write the residual function for R_f and T_f in terms of the corresponding points between camera captured images I and I':

$$\rho(R_f, T_f) = \sum_k (d^2(\mathbf{m'}_k, F\mathbf{m}_k) + d^2(\mathbf{m}_k, F^{\mathbf{T}}\mathbf{m'}_k))$$

We also denote the error from Equation 16 by a function of R_f, T_f as $\sigma(R_f, T_f)$. Then we can estimate the fiducial motion parameters R_f and T_f by solving a nonlinear minimization problem as:

$$\underset{R_f, T_f}{minimize} \, (\rho(R_f, T_f) + \alpha\sigma(R_f, T_f)) \quad (17)$$

where α is the relative confidence of the different error measures.

Now we present an iterative scheme for computing relative position and orientation of the camera.

1. for each motion, fiducial motion parameters R_f and T_f are estimated by minimizing Equation 17. From the camera images, extract corresponding points. This was implemented with a stereo correspondence algorithm based on Singular Value Decomposition [17] and optimized using RANSAC [6].

2. Estimate camera orientation Φ and position Δ by minimizing the residual function from the epi-polar constraints in Equation 16.

3. Based on current estimated camera orientation and position, re-estimate fiducial motion parameters R_f and T_f for each motion by minimizing Equation 17, with the current estimates of R_f and T_f as initial guess.

4. Repeat Step 2, 3 until convergence (usually two or three iterations).

This iterative algorithm is used in the experiments shown in Section 5.

4.2 Differential Motion Contraints

In this section we consider auto-calibration of a system undergoing differential motion. We assume that the field of view of laser ranger finder and camera overlap. The intuitive approach is the following. The laser range finder gives the 3D coordinates of a number of points in the environment. Given Φ and Δ and the correct fiducial motion, these 3D points determined by the laser range finder can be correctly projected onto the image, *and* the optic flow can be predicted at these points. Calculating the spatio-temporal image derivatives at these points, we can use the optic flow constraint equation as an error function, then solve for the Φ and Δ that minimize this error function.

In fact, it is theoretically not necessary to know the correct fiducial motion. Since the planar laser range finder only finds points that lie on a plane, the motion of all the points viewed by the laser range finder is constrained to be a planar projective transformation. Measuring deviation from this constraint gives an error measure that does not require knowledge of the fiducial motion.

To be more formal, here are basic notations describing the ego-motion of the camera and the corresponding optic flow of points viewed by the planar laser range finder. Let (X, Y, Z) denote the coordinates of a scene point respect to the camera, and let (x, y) denote the corresponding coordinates in the normalized image plane (i.e., the focal length is 1 so the image plane is the plane $Z = 1$). The perspective projection of a scene point $P = (X, Y, Z)^{\mathrm{T}}$ to an image point $p = (x, y)^{\mathrm{T}}$ is described by:

$$p = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \end{bmatrix} \quad (18)$$

The ego-motion of camera[2] is determined by a rotation $\vec{\omega} = (\omega_X, \omega_Y, \omega_Z)^{\mathrm{T}}$ and a translation $T = (T_X, T_Y, T_Z)^{\mathrm{T}}$. Due to the camera motion, the scene point $P = (X, Y, Z)^{\mathrm{T}}$ has a velocity $\dot{P} = (\dot{X}, \dot{Y}, \dot{Z})^{\mathrm{T}}$ with respect to camera coordinates system, described by:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = -\vec{\omega} \times \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - T \quad (19)$$

The instantaneous motion of the camera creates a 2D optic flow (u, v) of an image point (x, y) in the image plane,

[2]Note, we are about changed the definition of ω from the derivations in the discrete section. ω is now the angular velocity vector, and not a vector of Rodrigues parameters that represents a discrete rotation matrix.

which can be expressed by:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -(\frac{T_X}{Z} + \omega_Y) + x\frac{T_Z}{Z} + y\omega_Z - x^2\omega_Y + xy\omega_X \\ -(\frac{T_Y}{Z} - \omega_X) + y\frac{T_Z}{Z} - x\omega_Z + y^2\omega_X - xy\omega_Y \end{bmatrix} \tag{20}$$

Assume that the laser plane is describe by $1 = AZ + BX + CY$ with respect to the camera coordinates. By perspective projection, this yields: $\frac{1}{Z} = A + Bx + Cy$ where: (x, y) are normalized image coordinates. Therefore, Equation 20 can be rewritten as (derivation taken from [11]):

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a + bx + cy + gx^2 + hxy \\ d + ex + fy + gxy + hy^2 \end{bmatrix} \tag{21}$$

where:

$$\begin{aligned} a &= -AT_X - \omega_X & b &= AT_Z - BT_X \\ c &= \omega_X - CT_X & d &= -AT_Y + \omega_X \\ e &= -\omega_Z - BT_Y & f &= AT_Z - CT_Y \\ g &= -\omega_Y + BT_Z & h &= \omega_X + CT_Z \end{aligned}$$

Equation 21 describes the 2D motion of the laser plane in the image plane, expressed as a linear function in eight parameters (a, b, c, d, e, f, g). All scene points viewed by the laser should have an optic flow that conforms to this system. If the spatio-temporal derivatives of the image intensity are computed, and the scene in view is Lambertian, then as an error measure, we can use the optic-flow constraint equation:

$$uI_x + vI_y + I_t = 0 \tag{22}$$

where: (u, v) is the displacement of the laser point on the image place, I_x, I_y, I_t are the derivatives of the image intensity with respect to the direction of X, Y and T respectively.

An estimate of Φ and Δ define the locations where a scene point is projected onto the image. If, additionally, the fiducial motion of the system is known, the values of $a, b, \ldots, h$ can be computed explicitly to give an exact estimate of u, v for each laser point projected on the image. The deviation from the optical flow constraint is summed over all of these points to give a direct error measure for Φ, Δ.

$$Err(\Phi, \Delta) = \sum (uI_x + vI_y + I_t)^2 \tag{23}$$

Surprisingly, it is possible to define an error measure even if the fiducial motion is not known. For any fiducial motion, the optic flow at all points on the laser plan must fit the model given by Equation 21. Symbolically substituting that optic flow into the optic flow constraint equation (Equation 22), we get one equation for each projected laser point, which is linear in the unknowns $a, b, \ldots, h$:

$$\begin{aligned} & I_x(x,y)(a + bx + cy + gx^2 + hxy) \\ + \; & I_y(x,y)(d + x + fy + gxy + hy^2) \\ + \; & I_t = 0 \end{aligned} \tag{24}$$

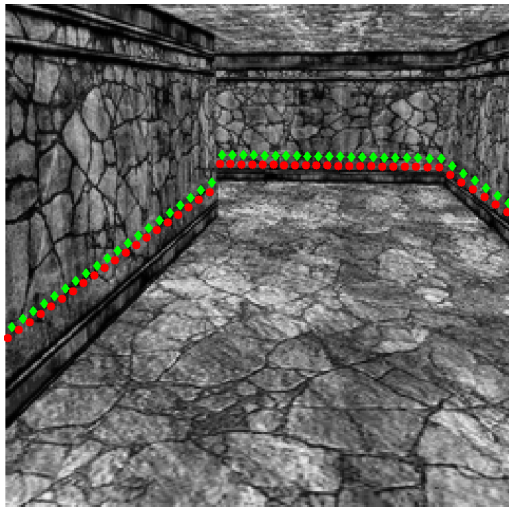

Figure 2. Running the auto-calibration for discrete motions with simulated data. The green diamonds are reprojections of the actual laser points, the red circles are reprojections for the estimated Φ and Δ.

The residual of the solution to this linear system defines an error function for Φ and Δ. In essence this defines exactly the same constraint as [11], but instead of solving for egomotion from a single video sequence, we use the error function to find a Φ and Δ which are consistent with image derivative measurements on the plane.

In our implementation all of the above error function are minimized using MATLAB's nonlinear optimization tools. We find that *when* the system converges at all, it does so after fewer than 40 Levenberg-Marquardt iterations. The experimental section delineates the convergence properties of the system and accuracy of that solution for several different scenes.

5 Experiments

The section presents results from experiments with the auto-calibration algorithm defined in Section 4.1.2 for discrete system motions, and the algorithm which minimizes Equation 23 for differential system motions.

Discrete Motion Simulations The initial test of the auto-calibration for discrete system motion uses simulated data created in a graphics environment. Simulated laser data and two images of the scene were generated for a known fiducial motion. Corresponding points in these two images were calculated, and the iterative algorithm from Section 4.1.2 was run. For this simulated data, the iterative algorithm calculates a Φ and Δ which gives an epi-polar error with an average error for each point from its corresponding epi-polar line of 0.68 pixels. Because this data is simulated,

Figure 3. Results of running the auto-calibration for discrete motions. The red circles circles are the re-projected images of the points measured by the laser range finder using the Φ and Δ. Green squares show reprojection using correct (hand calculated) Φ and Δ. (Top) After one iteration of algorithm described in Section 4.1.2. (Bottom) After three iterations of alternating refinement of fiducial motion and camera position.

we can also compare the computed Φ and Δ to the real solution, and find an error in the position of 1 centimeter[3], and a rotational error of (slightly less than) 1°. Results are depicted in Figure 2.

Discrete Motion Actual Experiments The second experiment uses real data from the robot and camera system shown in Figure 1. The results of the iterative estimation algorithm are illustrated in Figure 3. After one iteration (minimizing the epi-polar error for the initial estimate of fiducial motion), the mean distance from each laser point to its epi-polar line was 1.5 pixels. After 3 iterations (alternating between updating Φ, Δ and R_f, T_f), the mean epi-polar distance error drops to 0.66 pixels, and the orientation estimate of the camera is improved but retains a slight error

[3]The translation estimate is limited to the plane of the floor because fiducial motions were limited to that plane. That is, we get an error of 1 cm. in estimating the X and Z component of the position, and no estimate at all of the Y component of the position. The figures (for the discrete algorithms) are drawn with the correct Y component "hard-coded".

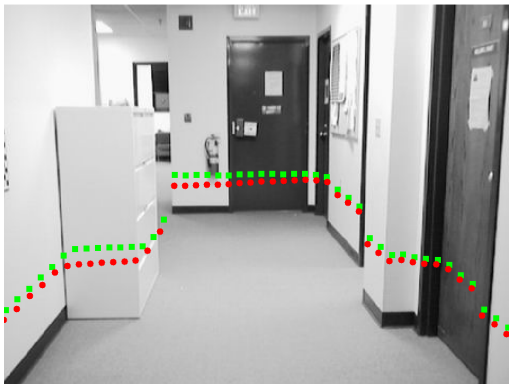

Figure 4. A similar experiment two Figure 3, but with a more complicated scene. Shown here is the result after 3 iterative refinement stages

around its horizontal axis. The same experiment was run on data from a more complicated scene (with great depth variation). Figure 3 shows results for this case after 3 iterations, which converged with a mean epi-polar distance error of 0.84 pixels.

Differential Motion Actual Experiments. Final auto-calibration results from the algorithm in Section 4.2 are shown in Figure 5. This algorithm is highly sensitive to the initial condition and often does not converge to be near the correct solution. Empirically we have observed that for different situations, there is a limit to the tolerated error in the initial estimate of the Φ. This critical error limit depends on the complexity of the scene (more depth variation in the scene gives stronger constraints) and the availability of texture (as required for most image derivative methods).

Situation	Critical rotation error
Complex Scene, Good Texture	10°
Complex Scene, Little Texture	7°
Simple Scene, Good Texture	5°
Simple Scene, Little Texture	NA

Conclusions: Auto-calibration is an important tool for many real-world applications. The consistent rigid motion constraint is a general tool that can lead to auto-calibration algorithms for many different kinds of sensors. The results here are an encouraging first look at the possibilities for a system with a camera and a laser range finder. This work could be usefully extended to test these algorithms on a broader set environments, especially if there is a way to give a useful parameterization (or other formalization) of "typical outdoor environments". The consistent rigid motion constraint can also be used as a framework for the auto-calibration of different varieties of sensors including an electronic compass, gyroscope, and inertial sensors.

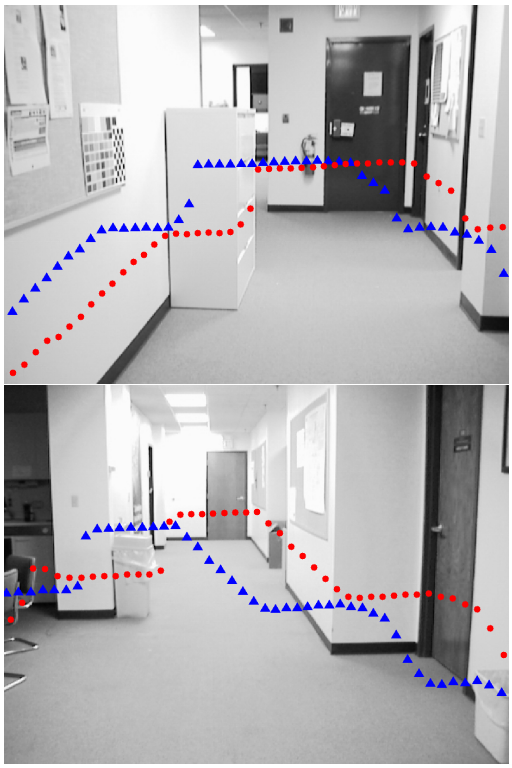

Figure 5. Two results of running the auto-calibration for differential motions. The red dots are the re-projected points for the estimated Φ, Δ. The blue triangles are re-projected according the (incorrect) Φ, Δ used as a starting condition for the optimization algorithm of Section 4.2.

References

[1] P Baker, R Pless, C Fermuller, and Y Aloimonos. New eyes for shape and motion estimation. In *Biologically Motivated Computer Vision (BMCV2000)*, 2000.

[2] T Brodský, C Fermüller, and Y Aloimonos. Self-calibration from image derivatives. In *Proc. International Conference on Computer Vision*, pages 83–89, 1998.

[3] D Demirdjian, A Zisserman, and R Horaud. Stereo autocalibration from one plane. In *Proceedings of the 6th European Conference on Computer Vision, Dublin, Irlande*, volume II, pages 625–639. Sprin-ger-Ver-lag, Juin/Juillet 2000.

[4] J E Dennis. Nonlinear least squares. In D Jacobs, editor, *State of the Art in Numerical Analysis*, pages 269–312. Academic Press, 1977.

[5] O D Faugeras, Q T Luong, and S J Maybank. Camera self-calibration: Theory and experiments. In *Proc. European Conference on Computer Vision*, pages 321–334, Santa Margherita Ligure, Italy, 1992.

[6] 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*, 24(6):381–395, 1981.

[7] S Gottschalk and J F Hughes. Autocalibration for virtual environments tracking hardware. In *SIGGRAPH*, pages 65–71. ACM Press, 1993.

[8] R I Hartley. An algorithm for self calibration from several views. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 908–912, Seattle, WA, 1994.

[9] B Hoff and R Azuma. Autocalibration of an electronic compass in an outdoor augmented reality system. In *Proceedings of IEEE and ACM International Symposium on Augmented Reality*, pages 159–164, Munich, Germany, October 2000.

[10] R Horaud, G Csurka, and D Demirdjian. Stereo calibration from rigid motions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1446–1452, 2000.

[11] Michal Irani, Benny Rousso, and Shmuel Peleg. Recovery of Ego-Motion Using Image Stabilization. In *ICVPR*, pages 454–460, March 94.

[12] O Jokinen. Self-calibration of a light striping system by matching multi-ple 3-d profile maps. In *Proceedings of the 2nd International Conference on 3-D Digital Imaging and Modeling*, pages 180–190. IEEE, 1999.

[13] K Levenberg. A method for the solution of certain problems in least squares. *Quarterly Applied Math. 2, pp. 164-168, 1944.*, 2:164–168, 1944.

[14] D Marquardt. An algorithm for least squares estimation of nonlinear parameters. *SIAM Journal of Applied Math*, 11:431–441, 1963.

[15] J J Mor—. The levenberg-marquardt algorithm:implementation and theory. In G A Watson, editor, *Lecture Notes in Mathematics*, volume 630, pages 105–116. Springer Verlag, 1977.

[16] C F Olson. Probabilistic self-localization for mobile robots. *IEEE Transactions on Robotics and Automation*, 16(1):55–65, Feb 2000.

[17] M Pilu. A direct method for stereo correspondence based on singular value decomposition. In *CVPR97*, pages 261–266, 1997.

[18] M Pollefeys, R Koch, and L Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. International Conference on Computer Vision*, pages 90–95, Bombay, January 1998.

[19] L Wolf and A Zomet. Sequence-to-sequence self calibration. In *ECCV (2)*, pages 370–382, 2002.

[20] A Zisserman, P Beardsley, and I Reid. Metric calibration of a stereo rig, 1995.