

Constraints for Heterogeneous Sensor Auto-Calibration

Qilong Zhang and Robert Pless
Department of Computer Science and Engineering
Washington University in St. Louis
St. Louis, MO. 63130

Abstract

This paper describes a framework for calibrating motion sensitive sensors attached to an autonomous vehicle. Based on camera auto-calibration techniques, we derive constraints relating sensor measurements to the relative position and orientation of different sensors. For the case of a camera and laser range finder, we present an auto-calibration algorithm for discrete motions. Auto-calibration tools are vital for real world use of vision algorithms, and the framework presented here is important to merge image sensors with GPS, inertial, infrared and ultrasonic sensors.

1 Introduction

Fusing data captured by multiple sensors is important for many robotic tasks. For sensors such as video cameras, laser range finders, and inertial sensors, 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 pose of each from each other, or of each from a fiducial coordinate system. Fig. 1 shows a typical sensor package in an autonomous robot with a variety of different sensors.

The calibration of each of these geometric sensors can be decomposed into intrinsic and extrinsic parameters. The extrinsic calibration parameters are the position and orientation of the sensor relative to some fiducial coordinate system. The intrinsic parameters, such as the calibration matrix of a camera, affect how the sensor samples the world. We concentrate only on finding the extrinsic calibration parameters because for many sensors there already exists self-calibration techniques, for cameras [5, 9, 13, 24, 3, 8], for optical and magnetic 6 DOF sensors [12], and for other sensors such as an electronic compass [15]. 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 a framework for auto-



Figure 1: A B21rTM Mobile Robot from iRobot Corporation is an often used experimental robot platform. Autonomous systems may compute motion estimates with a large variety of different sensors. The goal of this paper is to study auto-calibration methods that find the rotation Φ and the translation Δ which relate the coordinate systems of different sensors, by considering data captured simultaneously from both sensors when the robot undergoes arbitrary motions in an unknown environment.

calibration of motion sensitive sensors on rigidly moving platforms. We illustrate this algorithm on a mobile robot with a camera and laser range finder. To our knowledge this is the first paper to discuss about auto-calibration of a camera to other sensors. It is important that the vision community actively participates in driving research in multi-sensor fusion, as sensor systems become more heterogeneous.

This work was inspired by two previous works. Determining the geometric transformations between two cameras mounted on a rigidly moving object was discussed in [2], and a theory and implementation for solving for the intrinsic and extrinsic parameters were calculated in [25, 6]. 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 be consistent with their fixed, relative positions.

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 [7, 16, 26]. This is a fundamentally different kind of constraint, and requires that the cameras have overlapping fields of view. There have also been a great deal of works on calibration for laser scanners. Finding the geometric relationship between a laser scanner and a camera is vital to creating metric depth estimates from the camera images, and auto-calibration methods exist for this problem as well [17]. Laser scanners are the parts of active vision systems which project points or stripes which are then viewed by the cameras, as opposed to laser range finders which report distances to objects that lie in particular directions.

The remainder of this paper is organized as follows. The next section introduces notation used to represent the pose 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 cases. 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 of the techniques presented.

2 Background

This paper is aimed at solving constraints on the poses of multiple sensors relative to each other. An equivalent problem – and easier to define – is to relate the pose of each sensor relative to some fiducial coordinate system. For the remainder of this paper, it is assumed that the sensors are always fixed relative to the fiducial coordinate system (which is undergoing a rigid transformation).

The goal of this paper can then be stated as finding the rigid transformation from each sensor coordinate system to the fiducial coordinate system. Suppose a point p_i in the coordinate system of sensor i is located at a point p_f in the fiducial coordinate system, then the rigid transformation between the fiducial system and sensor i is represented by

$$p_f = \Phi_i p_i + \Delta_i \quad (1)$$

where Φ_i is a 3x3 orthonormal matrix representing the sensor orientation, and Δ_i an offset vector corresponding to the sensor position, with respect to the fiducial system. In what follows, the variable pair (Φ, Δ) *always* corresponds to transformations between coordinate systems of different

sensors. (R, t) and $(\vec{\omega}, \vec{t})$ represent discrete and differential motion, respectively.

The next section introduces the coherent motion constraints which relate sensor motion to the fiducial motion and the pose of the sensor with respect to the fiducial coordinate system.

3 Coherent Motion Constraints

If sensors are rigidly attached to the system, the motions of the sensors are exactly constrained with respect to the motion of the fiducial coordinates of the system. Intuitively, the relationships between the local motion at each sensor constrain their relative pose. In this section we write the relationships between the fiducial motion 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 :

$$\vec{t}_i = \Phi_i^{-1}(-\vec{t}_f - \Delta_i \times \vec{\omega}_f) \quad (2)$$

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

$$\vec{\omega}_i = \Phi_i^{-1} \vec{\omega}_f \quad (3)$$

This gives constraints on Φ_i, Δ_i as a relationship between the local differential motion to the fiducial differential motion. The rest of this paper concentrates on the discrete case, and complete details for the differential case are in [23].

3.2 Discrete Motion

Suppose the fiducial system undergoes a discrete motion described by a rotation R_f and translation t_f . Then we can write down the relation between the discrete motion (R_i, t_i) of sensor i and that of the fiducial system :

$$\begin{aligned} R_i &= \Phi_i^{-1} R_f \Phi_i \\ t_i &= \Phi_i^{-1} R_f \Delta_i + \Phi_i^{-1} (t_f - \Delta_i). \end{aligned} \quad (4)$$

This gives a constraint on Φ, Δ as relationship between the discrete motion in the local coordinate system and that 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. For intuition, 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 pose of each sensor. For simplicity, we describe this process for the case of two sensors, and choose as the fiducial system the coordinate system attached to one of the two sensors. Then the pose of another sensor relative to the fiducial coordinate system are denoted by (Φ, Δ) .

The system must undergo several different motions. In each motion, both of the sensors can independently measure their rotation and translation exactly. Then for the j^{th} 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 j^{th} motion from Eq. (4):

$$\begin{aligned} R^j &= \Phi^{-1} R_f^j \Phi \\ t^j &= \Phi^{-1} R_f^j \Delta + \Phi^{-1} (t_f^j - \Delta) \end{aligned} \quad (5)$$

Let ω denote the rotation vector corresponding to the rotation matrix R . In terms of rotation vectors, the rotation constraint of Eq. (5) can be written as $\omega_f^j = \Phi \omega^j$. Each motion gives one constraint of this form, and given several different motions we can solve for Φ with standard linear least squares.

This computed matrix Φ doesn't always meet the properties of a rotation matrix. We can compute a rotation matrix $\hat{\Phi}$ to approximate the computed matrix Φ by minimizing Frobenius norm of the difference $\hat{\Phi} - \Phi$, subject to $\hat{\Phi} \hat{\Phi}^\top = I$. The details about this matrix computation can be referred in [11]. This concludes the direct estimation of the sensor orientation with respect to the fiducial system.

After the sensor orientation Φ is determined, we can continue to solve the sensor position Δ . It is done with linear least squares minimizing the following cost function:

$$\sum_j \|\Phi t^j - t_f^j - R_f^j \Delta + \Delta\|^2 \quad (6)$$

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

4.1 Using the Epipolar 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). A more reasonable assumption is that a sensor can compute an error measure relating its measurements to a motion estimate.

The outline of the approach for the case of a camera 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 epipolar constraint. We find corresponding feature points between two images taken by the camera, and calculate how well this epipolar constraint fits the set of correspondences. This becomes an error for that choice of Φ and Δ .

More formally, let I and I' denote the images before and after camera motion j , respectively. Suppose there are given m corresponding points between I and I' noted by m_k, m'_k ($k = 1, \dots, m$), they satisfy the epipolar constraint:

$$(\tilde{m}'_k)^{\top} F^j \tilde{m}_k = 0 \quad (k = 1, \dots, m) \quad (7)$$

where \tilde{m} is the homogenous representation of m , and F^j fundamental matrix for motion j , relating images I and I' . We can write an expression for the fundamental matrix in terms of the motion in the fiducial coordinate system :

$$F^j = K^{\top} \left[R_f^j \Delta - \Delta + t_f^j \right]_{\times} R_f^j \Phi K^{-1} \quad (8)$$

where K is the intrinsic matrix for the camera. 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 nonlinear least-square problem, which minimizes the residual from the epipolar constraints (for all points k in all frames j .) obtained by Eq. (7),

$$\sum_j \sum_k ((\tilde{m}'_k)^{\top} F^j \tilde{m}_k)^2 \quad (9)$$

where F^j is expressed in terms of Φ , Δ , R_f^j , t_f^j in Eq. (8)¹. This algebraic error, however, does not have direct interpretation in the measurement space, and the solutions that minimize algebraic distance may not be those expected intuitively. So we write an alternative error function for Eq. (9) using the discrepancy in the epipolar geometry,

$$\sum_j \sum_k \left[d^2(m'_k, F^j \tilde{m}_k) + d^2(m_k, F^{j\top} \tilde{m}'_k) \right] \quad (10)$$

Where $d^2(\cdot, \cdot)$ is the squared image distance between a point and a line. Since Euclidean distance in the metric space is used, the new criterion is more geometrically and statistically meaningful [14].

In our experiments we have used MATLAB's nonlinear optimization toolkit, with the Levenberg-Marquardt method [19]. They demand the initial estimate of the orientation and position of the camera to be provided. If we have prior knowledge of the camera and the fiducial system, we can directly use it as the initial estimation.

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

4.2 Implementation

We implement the previous algorithm on a B21rTM Mobile Robot from iRobot Corporation. A Sony DFW-VL500 camera is mounted on top of the robot, and a SICK-PLS laser range finder reports 2D range measurements by scanning 180 degrees of the environment parallel to the ground plane, with an angle resolution of one measurement per degree and a range measuring accuracy of $\pm 5\text{cm}$. The center of the laser range finder is set as the fiducial coordinate system. We assume that the robot undergoes planar motion.

4.2.1 Estimating the Fiducial Motion from Laser Range Data

Initially, we implement an algorithm (a simplified version of [20]) to compute the fiducial motion from the laser range 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 $\mathcal{P} = \{p_1, p_2, \dots, p_{180}\}$ and $\mathcal{P}' = \{p'_1, p'_2, \dots, p'_{180}\}$, where \mathcal{P} is the laser reading before fiducial motion, and \mathcal{P}' is the laser reading after. The generalized Hausdorff measure is defined as

$$H_k(\mathcal{P}, \mathcal{P}') = \max(h_k(\mathcal{P}, \mathcal{P}'), h_k(\mathcal{P}', \mathcal{P})) \quad (11)$$

where

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

where *kth* denotes the *k*-th ranked value. In this paper, we are interested in using this generalized Hausdorff distance to measure how \mathcal{P}' matches with \mathcal{P} after a rigid transformation. In terms of fiducial motion R_f and t_f , the error approximation can be defined as:

$$\sum_{k=1}^K [H_k(R_f \mathcal{P} + t_f, \mathcal{P}') + H_k(R_f^{-1}(\mathcal{P}' - t_f), \mathcal{P})] \quad (12)$$

where K is the parameter determining how many laser points are taken into account (In practice, the two laser readings are assumed to overlap sufficiently, and we then use a value of $K = 160$, which allows up to 20 range points for each laser reading to be outliers). The estimated fiducial motion parameters R_f and t_f are those which by minimizing above error function. More sophisticated methods for mobile robot motion estimation by matching laser range points can be found in [18, 21].

4.2.2 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 Eq. (12) 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 by writing an error function for R_f and t_f in terms of the corresponding points between images I and I':

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

We also denote the error from Eq. (12) by a function of R_f, t_f as $\psi(R_f, t_f)$. Then we can refine the fiducial motion parameters R_f and t_f by solving a nonlinear minimization problem as:

$$\min_{R_f, t_f} [(\rho(R_f, t_f) + \alpha\psi(R_f, t_f))] \quad (13)$$

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

Now we present the complete scheme for computing relative pose of the camera.

1. For each motion, fiducial motion parameters (R_f, t_f) are estimated by minimizing Eq. (12), and the outliers are removed. Corresponding points are extracted automatically from the camera images. This was implemented with a stereo correspondence algorithm based on Singular Value Decomposition [22] and optimized using RANSAC [10].
2. Estimate camera pose (Φ, Δ) by minimizing the error function from the epipolar constraints in Eq. (10).
3. Based on current estimated camera pose, refine fiducial motion parameters for each motion by minimizing Eq. (13), with the current estimates of R_f and t_f as initial guesses.
4. Repeat Step 2, 3 until convergence (usually two or three iterations).

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

5 Experiments

This section presents experimental results for the auto-calibration algorithm defined in Section 4.2.2 for a camera and laser range finder.

Discrete Motion Simulations The initial test of the auto-calibration for discrete system motion uses simulated data created in a graphics environment. Simulated laser range data and images of the scene were generated for known fiducial motions. Laser range points were corrupted

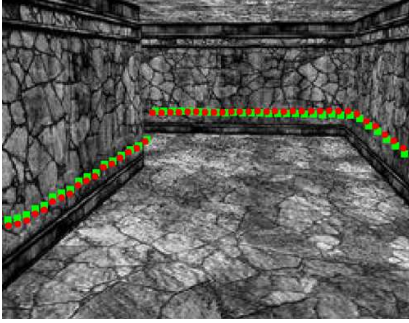


Figure 2: Running the auto-calibration for discrete motions with simulated data. The red circles are reprojections of the actual laser points, while the green diamonds are reprojections for the estimated Φ and Δ .

with zero-mean Gaussian random noise (see [1] for justification) with standard deviation $5cm$, and the estimation error for the fiducial motion was within 0.86° in rotation and $2cm$ in translation. Corresponding points in the image pairs were calculated so that the average distance error between the points and their corresponding epipolar lines was up to 0.5 pixel. The iterative algorithm from Section 4.2.2 was run to estimate Φ and Δ . In the experiment in which the virtual camera was set up with an orientation $\omega_\Phi = [-0.262, -0.175, 0.0]^\top$ and a position $\Delta = [10, -50, 5]^\top cm.$, we get following results (depicted in Fig. 2) for a simulated data set with 7 discrete motions:

- With the computed camera pose (Φ, Δ) , the average epipolar error for each point from its corresponding epipolar line: 0.62 pixels
- Comparing the computed Φ, Δ with the ground truth, we have rotational error 1.10° and positional error $4.8cm.$ ²

Discrete Motion Actual Experiments The proposed auto-calibration algorithm was also tested on real data from the robot system shown in Fig. 1. We calibrated the camera intrinsic parameters using [4], and assumed the camera had no significant lens distortion or the images were warped to eliminate it. The camera was placed on top of the robot with a pitch angle of about 10 degrees down towards and a height of about 1.1 meters with respect to the laser range finder. The camera resolution was set as 640×480 . Corresponding points were extracted and the average distance between the points and their epipolar lines was around 0.5 pixel. For the hallway scene with 15 discrete motions, illustrated in Fig. 3, we get following results:

²The 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 4.8 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”.

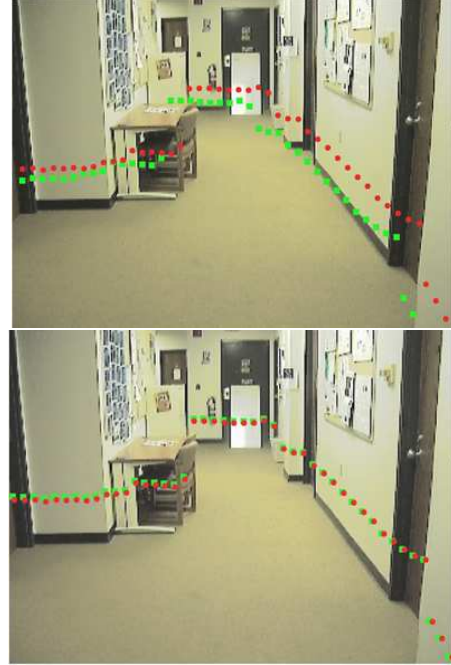


Figure 3: Results of running the auto-calibration for discrete motions. The green squares are the reprojected images of the range points measured by the laser range finder using the estimates of Φ and Δ . The red circles show reprojection using correct (hand calculated) camera pose. (Top) The result after one iteration of algorithm described in Section 4.2.2. (Bottom) After three iterations of alternating refinement of fiducial motion and camera position.

- After one iteration, with the computed camera pose (Φ, Δ) , the average distance from the feature points to their epipolar lines: 1.5 pixels.
- After three iterations, the average epipolar distance error drops to 0.89 pixel. As seen in Fig. 3, the estimate of the camera pose is improved through iterations.
- Fig. 4 illustrates results for a lounge scene after 3 iterations, which converged with an average epipolar distance error of 0.84 pixel.

Conclusions: Auto-calibration is an important tool for many real-world applications. The coherent 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 laser range finder. This work could be usefully extended to test these algorithms on a broader set of environments, especially if there is a way to give a useful parametrization (or other formalization) of “typical outdoor environments”. The coherent motion constraint can also be used as a framework for the auto-calibration of different varieties of sensors including an electronic compass, odometry, and inertial sensors etc.

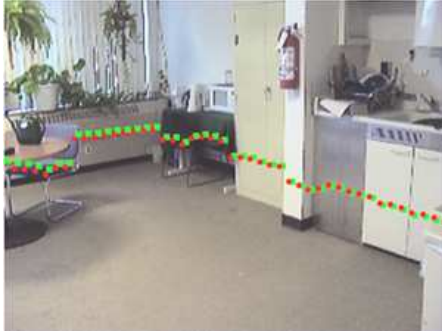


Figure 4: A similar experiment as Fig. 3, but with a lounge scene. Shown here is the result after 3 iterative refinement stages.

References

- [1] M. Adams and P. Probert. The interpretation of phase and intensity data from AMCW light detection sensors for reliable ranging. *Journal of Intelligent and Robotic Systems*, pages 441–458, 1996.
- [2] P. Baker, R. Pless, C. Fermuller, and Y. Aloimonos. New eyes for shape and motion estimation. In *Biologically Motivated Computer Vision (BMCV)*, pages 118–128, 2000.
- [3] A. Basu. Active calibration: Alternative strategy and analysis. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 93:495–500.
- [4] Jean-Yves Bouguet. Camera calibration toolbox for Matlab. <http://www.vision.caltech.edu/bouguetj/>, Jan. 2004.
- [5] T. Brodský, C. Fermüller, and Y. Aloimonos. Self-calibration from image derivatives. In *Proc. IEEE International Conference on Computer Vision*, pages 83–89, 1998.
- [6] J. Crowley, P. Bobet, and C. Schmid. Auto-calibration by direct observation of objects. *Image and Vision Computing*, 11(2):67–81, March 1993.
- [7] D. Demirdjian, A. Zisserman, and R. Horaud. Stereo auto-calibration from one plane. In *Proc. European Conference on Computer Vision*, volume II, pages 625–639. Springer Verlag, June 2000.
- [8] F. Du and M. Brady. Self-calibration of the intrinsic parameters of cameras for active vision systems. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 93:477–482.
- [9] O. Faugeras, Q. Luong, and S. Maybank. Camera self-calibration: Theory and experiments. In *Proc. European Conference on Computer Vision*, pages 321–334, 1992.
- [10] M. Fischler and R. 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.
- [11] G. Golub and C. Van Loan. *Matrix Computation*. John Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, Maryland, third edition, 1996.
- [12] S. Gottschalk and J. Hughes. Autocalibration for virtual environments tracking hardware. In *SIGGRAPH*, pages 65–71. ACM Press, 1993.
- [13] R. Hartley. An algorithm for self calibration from several views. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 908–912, 1994.
- [14] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [15] B. Hoff and R. Azuma. Autocalibration of an electronic compass in an outdoor augmented reality system. In *Proc. IEEE and ACM International Symposium on Augmented Reality*, pages 159–164, October 2000.
- [16] 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.
- [17] O. Jokinen. Self-calibration of a light striping system by matching multiple 3-d profile maps. In *Proc. the 2nd International Conference on 3-D Digital Imaging and Modeling*, pages 180–190. IEEE, 1999.
- [18] F. Lu and E. Miliotis. Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*, pages 249–275, May 1997.
- [19] 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.
- [20] C. Olson. Probabilistic self-localization for mobile robots. *IEEE Transactions on Robotics and Automation*, 16(1):55–65, Feb 2000.
- [21] S. Pfister, K. Kreichbaum, and et la. Weighted range sensor matching algorithms for mobile robot displacement estimation. In *Proc. 2002 IEEE International Conference on Robotics and Automation*, pages 1667–1674, May 2002.
- [22] M. Pilu. A direct method for stereo correspondence based on singular value decomposition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 261–266, 1997.
- [23] R. Pless and Q. Zhang. Extrinsic auto-calibration of a camera and laser range finder. Tech. Report WUSCE-2003-59. <http://cse.seas.wustl.edu/techreportfiles/getreport.asp?297>.
- [24] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. IEEE International Conference on Computer Vision*, pages 90–95, January 1998.
- [25] L. Wolf and A. Zomet. Sequence-to-sequence self calibration. In *European Conference on Computer Vision*, pages 370–382, 2002.
- [26] A. Zisserman, P. Beardsley, and I. Reid. Metric calibration of a stereo rig. In *Proc. IEEE Workshop on Representations of Visual Scenes*, pages 93–100, 1995.