

# Weighted Range Sensor Matching Algorithms for Mobile Robot Displacement Estimation

Sam T. Pfister, Kristo L. Kriechbaum, Stergios I. Roumeliotis, Joel W. Burdick  
{sam,klk,stergios,jwb}@robotics.caltech.edu  
Mechanical Engineering, California Institute of Technology  
Pasadena, California 91125

## Abstract

This paper introduces a “weighted” matching algorithm to estimate a robot’s planar displacement by matching two-dimensional range scans. The influence of each scan point on the overall matching error is weighted according to its uncertainty. We develop uncertainty models that account for effects such as measurement noise, sensor incidence angle, and correspondence error. Based on models of expected sensor uncertainty, our algorithm computes the appropriate weighting for each measurement so as to optimally estimate the displacement between two consecutive poses. By explicitly modeling the various noise sources, we can also calculate the actual covariance of the displacement estimates instead of a statistical approximation of it. A realistic covariance estimate is necessary for further combining the pose displacement estimates with additional odometric and/or inertial measurements within a localization framework [1]. Experiments using a Nomad 200 mobile robot and a Sick LMS-200 laser range finder illustrate that the method is more accurate than prior techniques.

## 1 Introduction and Preliminaries

A robot’s ability to determine and maintain knowledge of its absolute position is a basic requirement for long term autonomous navigation and operation. Consequently, the subjects of localization and mapping have received considerable attention (e.g., see [2, 3, 4, 5, 6]). Two-dimensional range finders, such as laser range finders [7] or rings of ultrasonic range sensors [8], are commonly used as a part of many mobile robot localization and mapping procedures. This paper introduces a “weighted” range sensor matching algorithm to estimate a robot’s displacement between the configurations where range scans are obtained. This novel algorithm takes into account several important physical phenomena that affect range sensing accuracy, and that have been neglected in prior work. Our experiments (Section 6) show that this algorithm is not only efficient, but more accurate than non-weighted matching methods, such as that of Ref. [9]. In addition, by computing the actual covariance of the displacements, the weighted matching algorithm provides the basis for optimal fusion of these estimates with odometric and/or inertial measurements [1] and subsequently support localization and mapping tasks.

To best understand the content of this paper and its contributions, we first describe the basic problem, and how our solution approach differs from previous ones. We focus on mobile robots operating in planar environments. Our method is best suited to indoor environments, though it can be extended to structured outdoor environments. We assume that the robot is equipped with odometry and a steerable range sensor. These distance measurements can come from sensors such as sonars, infrareds, cameras, radars etc. The basic principle behind our approach of incorporating sensor noise models into the displacement estimation algorithm generally applies to any case of dense range data matching processes. Different sensor noise/uncertainty models, which will be based on the particular characteristics of each sensor, are needed for different applications.

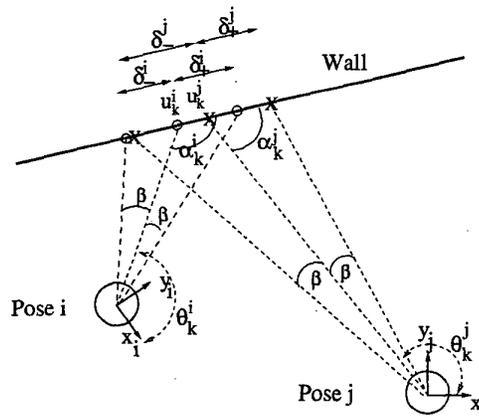


Figure 1: Geometry of the range sensing process

The robot starts at an initial configuration,  $g_1$ , and moves through a sequence of configurations, or poses,  $g_i$ ,  $i = 2, \dots, m$ . Here  $g_i \in SE(2)$  denotes the robot’s position and orientation relative to a fixed reference frame,  $g_0$ . We assume that at each pose, the robot measures the range to the boundary of its nearby environment along rays which are separated by a uniform angle,<sup>1</sup>  $\beta$  (see Fig. 1). Let the set of  $n_i$  scan points in the  $i^{th}$  pose be denoted by  $\{u_k^i\}$ ,  $k = 1, \dots, n_i$ . The scan point coordinates are described in the robot’s body frame, and the  $k^{th}$  scan point in pose  $i$  takes the form:

$$u_k^i = l_k^i \begin{bmatrix} \cos \theta_k^i \\ \sin \theta_k^i \end{bmatrix} \quad (1)$$

<sup>1</sup>The extension to non-uniform angle  $\beta$  is straight forward.

where  $l_k^i$  is the measured distance to the environment's boundary in the direction denoted by  $\theta_k^i$  (see Fig. 1).

Our main goal is to accurately estimate the robot's displacement between poses by matching range data. First, assume that the range scans at poses  $i$  and  $j$  have a sufficient number of corresponding points to be successfully matched (see Section 4). Let  $\{u_k^i, u_k^j\}$  for  $k = 1, \dots, n_{ij}$  be the set of corresponding scan point pairs. From these pairs we first want to estimate the relative displacement between poses  $i$  and  $j$ :  $g_{ij} = g_i^{-1}g_j = (R_{ij}, p_{ij})$  where

$$R_{ij} = \begin{bmatrix} \cos \Delta\phi_{ij} & -\sin \Delta\phi_{ij} \\ \sin \Delta\phi_{ij} & \cos \Delta\phi_{ij} \end{bmatrix} \quad p_{ij} = \begin{bmatrix} \Delta x_{ij} \\ \Delta y_{ij} \end{bmatrix} \quad (2)$$

i.e., the displacement is described by a translation  $(\Delta x_{ij}, \Delta y_{ij})$  and a rotation,  $\Delta\phi_{ij}$ . We next wish to estimate the covariance,  $P^{ij}$ , of the displacement estimate. This covariance is necessary mainly for two reasons. First, it is an indicator of the quality of the displacement estimates. Large diagonal elements of the covariance matrix indicate increased uncertainty. Any localization process should be aware of the level of confidence in the computed pose estimates. Second, the covariance is also required when combining the displacement estimates with measurements provided by other sensors. For example, within a Kalman filter framework, the contribution of different sensor measurements to the state estimate is weighted by the Kalman gains. The values of these gains depend on the covariances of all the sources of information contributing to the filter.

Our approach differs from prior work in that we incorporate, within the estimation algorithm, models of the uncertainty associated with the sensor measurements as well as with the matching process itself. This can be better understood by examining Fig. 1. This diagram shows a set of adjacent scan points that would be obtained when a range sensor samples points on a nearby wall. The boundary points sampled in pose  $i$  are indicated by circles, and labeled by  $u_{k-1}^i, u_k^i, \text{ and } u_{k+1}^i$ . The boundary points sampled in pose  $j$  are indicated by  $X$ 's and are labeled by  $u_{k-1}^j, u_k^j, \text{ and } u_{k+1}^j$ . Prior range matching methods (e.g., [10, 9]) have made the simplifying assumption that the range scans of different poses sample the environment's boundary at *exactly* the same points—i.e., point  $u_k^i$  corresponds exactly to  $u_k^j$ , etc. *This is generally not true.* In this paper, we model this *correspondence error*, which has been neglected in prior work, and incorporate this effect into our matching algorithm. As described in Sections 3.1 and 3.3, the range measurements are corrupted by noise and a bias term that is a function of the range sensing direction,  $\theta_k^i$ , and the incidence angle,  $\alpha_k^i$  (see Fig. 1). While the existence of these sources of uncertainty has previously been suggested [11, 12, 7, 10], our algorithm is the first to model their effect and account for it within the estimation process. Finally, by explicitly incorporating these models of uncertainty, our algorithm computes a realistic covariance estimate that accurately reflects the true uncertainty in the displacement estimates. Previous displacement estimation algorithms have neglected these effects in the covariance estimate, and thus their results may be overly optimistic [13].

Section 2 describes the general weighted point feature matching problem and its solution. Section 3 develops the correspondence and range measurement error models, and derives the associated weighting terms. Sections 4 and 5 summarize the point pairing selection and sensor incidence angle estimation procedures. The experiments in Section 6 demonstrate the accuracy, robustness and convergence range of our algorithm. Direct comparisons between this algorithm and previous methods (e.g. [9]) validate the effectiveness of our approach.

## 2 The Weighted Range Sensor Matching Problem

This section describes a general point feature matching problem and its formulation. The weighted matching solution for any specific implementation will depend upon models of a given range sensor's operation, which are subsequently developed.

Consider the range data from poses  $i$  and  $j$ :  $\{u_k^i\}$  and  $\{u_k^j\}$ . The actual range measurements will be imperfect. Let  $\{r_k^i\}$  and  $\{r_k^j\}$  be the "true" range measurements. The actual measurements will consist of:

$$\begin{aligned} u_k^i &= r_k^i + \delta u_k^i + b_k^i \\ u_k^j &= r_k^j + \delta u_k^j + b_k^j \end{aligned} \quad (3)$$

where  $\delta u_k^i$  and  $\delta u_k^j$  represent range measurement "process noise," while  $b_k^i$  and  $b_k^j$  denote the range measurement "bias." These terms are discussed in Sections 3.1 and 3.3. The term  $\delta u_k$  is generally well modelled as a zero-mean Gaussian noise process. The bias  $b_k$  in this case is an unknown offset. It can be approximated by a nonzero constant  $o_k$  (estimated based on a statistical model derived by measurement data), corrupted by a zero-mean additive Gaussian noise  $\delta b_k$  [12]. The covariance of this noise component reflects the level of confidence in the choice of the value  $o_k$ . Contingent on this approximation,  $b_k^i$  and  $b_k^j$  take the form:

$$b_k^i = o_k^i + \delta b_k^i; \quad b_k^j = o_k^j + \delta b_k^j \quad (4)$$

We will ignore the offsets for now (i.e., assume that  $o_k^i = o_k^j = 0$ ), but will consider their effect in Section 3.3.

Let  $(u_k^i, u_k^j)$  be corresponding points from the range scans at poses  $i$  and  $j$ . Accounting for the fact that scan data is measured in a robot-fixed frame, the error between the two corresponding points is

$$\epsilon_k^{ij} = u_k^i - R_{ij}u_k^j - p_{ij} \quad (5)$$

for a given value of  $R_{ij}$  and  $p_{ij}$ . Substituting from Eq.s (3) and (4) into Eq. (5) results in

$$\epsilon_k^{ij} = \underbrace{(r_k^i - R_{ij}r_k^j - p_{ij})}_{(i)} + \underbrace{(\delta u_k^i - R_{ij}\delta u_k^j)}_{(ii)} + \underbrace{(\delta b_k^i - R_{ij}\delta b_k^j)}_{(iii)} \quad (6)$$

A relative pose estimation algorithm aims to find the displacement  $g_{ij} = (R_{ij}, p_{ij})$  that suitably minimizes Eq. (6) over the

set of all correspondences. If the range scans do sample the exact same boundary points, then  $r_k^i - R_{ij} r_k^j - p_{ij} = 0$  when  $R_{ij}$  and  $p_{ij}$  assume their proper values. However,  $r_k^i$  and  $r_k^j$  generally do not correspond to the same boundary point. Hence, term (i) in Eq. (6) is the *correspondence error*, denoted by  $c_k^{ij}$ :  $c_k^{ij} = r_k^i - R_{ij} r_k^j - p_{ij}$ . The matching error  $\varepsilon_k^{ij}$  for the  $k^{\text{th}}$  corresponding point is also a function of: (ii) the error due to the actual measurement noise, and (iii) the measurement bias error.

We first make the realistic assumption that the correspondence errors, noise, and bias errors are mutually independent. The act of taking range measurements in the  $i^{\text{th}}$  pose will generally be independent of the measurement process in the  $j^{\text{th}}$  pose, and thus  $\delta u_k^i$  will be independent of  $\delta u_k^j$  (similarly for  $\delta b_k^i$  and  $\delta b_k^j$ ). Hence, the covariance of the matching error at the  $k^{\text{th}}$  point correspondence of poses  $i$  and  $j$  is:

$$\begin{aligned} P_k^{ij} &\triangleq E[\varepsilon_k^{ij}(\varepsilon_k^{ij})^T] = E[c_k^{ij}(c_k^{ij})^T] + E[\delta u_k^i(\delta u_k^i)^T] \\ &+ R_{ij} E[\delta u_k^j(\delta u_k^j)^T] R_{ij}^T + E[\delta b_k^i(\delta b_k^i)^T] \\ &+ R_{ij} E[\delta b_k^j(\delta b_k^j)^T] R_{ij}^T \quad (7) \\ &= C P_k^{ij} + N P_k^i + R_{ij} N P_k^j R_{ij}^T + B P_k^i + R_{ij} B P_k^j R_{ij}^T \\ &= Q_k^{ij} + R_{ij} S_k^{ij} R_{ij}^T \end{aligned}$$

where  $E[\cdot]$  is the expectation operation, and

$$\begin{aligned} C P_k^{ij} &= \text{covariance due to correspondence error} \\ N P_k^i &= \text{noise covariance in the } i^{\text{th}} \text{ pose scan points} \\ N P_k^j &= \text{noise covariance in the } j^{\text{th}} \text{ pose scan points} \\ B P_k^i &= \text{bias covariance in the } i^{\text{th}} \text{ pose scan points} \\ B P_k^j &= \text{bias covariance in the } j^{\text{th}} \text{ pose scan points} \\ Q_k^{ij} &\triangleq C P_k^{ij} + N P_k^i + B P_k^i \\ S_k^{ij} &\triangleq N P_k^j + B P_k^j. \end{aligned}$$

The matrices  $Q_k^{ij}$  and  $S_k^{ij}$  represent the configuration independent and dependent terms of  $P_k^{ij}$ . As shown below, the correspondence and bias errors depend on the sensor's incidence angle. The noise covariances are functions of the variables  $\theta_k^i$ ,  $\theta_k^j$ ,  $l_k^i$ , and  $l_k^j$ . In summary, the covariance matrix  $P_k^{ij}$  varies significantly for each scan point pair. Hence, it is not suitable to assume, as in prior work (e.g. [14, 9]), that  $P_k^{ij}$  is the identity matrix for all scan point pairs.

**Maximum Likelihood Formulation.** We use a maximum likelihood (ML) approach to formulate a general strategy for estimating the robot's displacement from a set of nonuniformly weighted point correspondences. Let  $\mathcal{L}(\{\varepsilon_k^{ij}\}|g_{ij})$  denote the *likelihood function* that captures the likelihood of obtaining the matching errors  $\{\varepsilon_k^{ij}\}$  given a displacement  $g_{ij}$ . Assuming the independence of the  $k = 1, \dots, n_{ij}$  measurements, the likelihood can be written as a product:

$$\mathcal{L}(\{\varepsilon_k^{ij}\}|g_{ij}) = \mathcal{L}(\varepsilon_1^{ij}|g_{ij}) \mathcal{L}(\varepsilon_2^{ij}|g_{ij}) \cdots \mathcal{L}(\varepsilon_n^{ij}|g_{ij}).$$

Recall that the measurement noise is considered to be a zero-mean Gaussian process. Also the bias is approximated as a

zero-mean Gaussian noise superimposed on a constant offset. Finally, as it is shown in Section 3.2, the correspondence noise can be approximated as a zero-mean Gaussian process. Neglecting the bias offset for the moment (see Section 3.3),  $\varepsilon_k^{ij}$  is the sum of zero-mean Gaussian random variables. Thus,  $\mathcal{L}(\{\varepsilon_k^{ij}\}|g_{ij})$  takes the form:

$$\mathcal{L}(\{\varepsilon_k^{ij}\}|g_{ij}) = \prod_{k=1}^{n_{ij}} \frac{e^{-\frac{1}{2}(\varepsilon_k^{ij})^T (P_k^{ij})^{-1} \varepsilon_k^{ij}}}{2\pi \sqrt{\det P_k^{ij}}} = \frac{e^{-M^{ij}}}{D^{ij}} \quad (8)$$

$$\text{where } M^{ij} = \frac{1}{2} \sum_{k=1}^{n_{ij}} (\varepsilon_k^{ij})^T (P_k^{ij})^{-1} \varepsilon_k^{ij} \quad (9)$$

$$D^{ij} = \prod_{k=1}^{n_{ij}} 2\pi \sqrt{\det P_k^{ij}} \quad (10)$$

The optimal estimate of the displacement maximizes  $\mathcal{L}(\{\varepsilon_k^{ij}\}|g_{ij})$  with respect to displacement. One can use any numerical optimization scheme to obtain this displacement estimate. Note however that maximizing Eq. (8) is equivalent to maximizing the log-likelihood function:

$$\ln[\mathcal{L}(\{\varepsilon_k^{ij}\}|g_{ij})] = -M^{ij} - \ln(D^{ij}) \quad (11)$$

and from the numerical point of view, it is often preferable to work with the log-likelihood function.

This problem's inherent structure allows for efficiency in the maximization procedure. Appendix A proves that the optimal estimate of the robot's translation can be found as follows.

**Proposition 1** *The weighted scan match translational displacement estimate,  $\hat{p}_{ij}$ , is:*

$$\hat{p}_{ij} = P_{pp} \left( \sum_{k=1}^{n_{ij}} (P_k^{ij})^{-1} (u_k^i - q_k) \right) \quad (12)$$

where  $\hat{R}_{ij} = \hat{R}_{ij}(\hat{\phi}_{ij}^-)$  is the rotational matrix calculated with the current estimate of the absolute orientation  $\hat{\phi}_{ij}$  before iteration, and:

$$P_{pp} = \left( \sum_{k=1}^{n_{ij}} (P_k^{ij})^{-1} \right)^{-1}, \quad q_k = \hat{R}_{ij} u_k^j. \quad (13)$$

There is not an exact closed form formula to estimate  $\Delta\phi_{ij}$ . However, there are two efficient approaches to this problem. In the first approach, the estimate of  $\Delta\phi_{ij}$  can be found by numerically maximizing Eq. (8) (or Eq. (11)) with respect to  $\Delta\phi_{ij}$  for a constant  $\hat{p}_{ij}$  calculated according to Prop. 1. This procedure reduces to numerical maximization over a single scalar variable  $\Delta\phi_{ij}$ , for which there are many efficient algorithms. Alternatively, one can develop the following second order iterative solution to this non-linear optimization problem (see [15] for details):

**Proposition 2** The weighted scan match rotational displacement estimate is updated as  $\hat{\phi}_{ij}^+ = \hat{\phi}_{ij}^- + \delta\hat{\phi}_{ij}$ , where:

$$\delta\hat{\phi}_{ij} \simeq -\frac{\sum_{k=1}^{n_{ij}} p_k^T (P_k^{ij})^{-1} J q_k}{\sum_{k=1}^{n_{ij}} q_k^T J (P_k^{ij})^{-1} J q_k} \quad (14)$$

with  $q_k$  as above, and

$$J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \quad p_k = u_k^i - \hat{p}_{ij} - \hat{R}_{ij} u_k^j \quad (15)$$

Using experimental data, this approximation agrees with the exact numerical solution up to 5 significant digits.

Prop.s 1 and 2 suggest an iterative algorithm for estimating displacement. An initial guess  $\hat{\phi}_{ij}^-$  for  $\phi_{ij}$  is chosen (usually based on the odometry estimate). A translation estimate  $\hat{p}_{ij}$  is computed using Prop. 1. This estimate is employed by Prop. 2 to update the current rotational estimate  $\hat{\phi}_{ij}^-$ . The improved  $\hat{\phi}_{ij}^+$  is the basis for the next iteration. The iterations stop when a convergence criterion is reached. We prefer an iterative algorithm for two reasons. First, nonlinear ML problems are suited to iterative computation. Second, the correct correspondence between point pairs cannot be guaranteed, especially in the first few algorithm iterations, where some inaccurate initial pairings are unavoidable.

Letting  $\tilde{p}_{ij} = p_{ij} - \hat{p}_{ij}$ ,  $\tilde{\phi}_{ij} = \phi_{ij} - \hat{\phi}_{ij}$  (i.e.,  $\tilde{p}_{ij}$ ,  $\tilde{\phi}_{ij}$  are translational and the rotational displacement error estimates), a direct calculation yields the following.

**Proposition 3** The covariance of the displacement estimate is:

$$P^{ij} = \begin{bmatrix} P_{pp} & P_{p\phi} \\ P_{\phi p} & P_{\phi\phi} \end{bmatrix} = \begin{bmatrix} E\{\tilde{p}_{ij}\tilde{p}_{ij}^T\} & E\{\tilde{p}_{ij}\tilde{\phi}_{ij}^T\} \\ E\{\tilde{\phi}_{ij}\tilde{p}_{ij}^T\} & E\{\tilde{\phi}_{ij}\tilde{\phi}_{ij}^T\} \end{bmatrix}$$

with  $P_{pp}$  as above and

$$P_{p\phi} = \frac{1}{r_T} \left( \sum_{k=1}^{n_{ij}} (P_k^{ij})^{-1} \right)^{-1} \sum_{k=1}^{n_{ij}} \left( (P_k^{ij})^{-1} J q_k \right)$$

$$P_{\phi\phi} = \frac{1}{r_T}$$

$$r_T = -\sum_{k=1}^{n_{ij}} q_k^T J (P_k^{ij})^{-1} J q_k.$$

For a given sensor, one must derive appropriate uncertainty models which are the substituted into the above procedure.

### 3 Scan Matching Error/Noise Models

In order to derive explicit expressions for the covariances of Eq. (7), this section develops models for the errors inherent in the range scan matching process.

### 3.1 Measurement Process Noise

Many range sensing methods are based on the time of flight (e.g., ultrasound and some laser scanners) or modulation of emitted radiation [12, 7]. The circuits governing these measurement methods are subject to noise. These effects often can be well modelled in a simple way, enabling the computation of  ${}^N P_k^i$  and  ${}^N P_k^j$ . We focus on the computation of  ${}^N P_k^i$ , as the one for  ${}^N P_k^j$  is completely analogous.

Recall the polar representation of scan data, Eq. (1). Let the range measurement,  $l_k^i$ , be comprised of the "true" range,  $L_k^i$ , and an additive noise term,  $\varepsilon_l$ :  $l_k^i = L_k^i + \varepsilon_l$ . The noise  $\varepsilon_l$  is assumed to be a zero-mean Gaussian random variable with variance  $\sigma_l^2$  (see e.g., [12] for justification). Also assume that error exists in the measurement  $\theta_k^i$ , i.e. the actual scan angle differs (slightly) from the reported or assumed angle. Thus,  $\theta_k^i = \Theta_k^i + \varepsilon_\theta$ , where  $\Theta_k^i$  is the "true" angle of the  $k^{\text{th}}$  scan direction, and  $\varepsilon_\theta$  is again a zero-mean Gaussian random variable with variance  $\sigma_\theta^2$ . Hence:

$$u_k^i = (L_k^i + \varepsilon_l) \begin{bmatrix} \cos(\Theta_k^i + \varepsilon_\theta) \\ \sin(\Theta_k^i + \varepsilon_\theta) \end{bmatrix} \quad (16)$$

If we assume that  $\varepsilon_\theta \ll 1$  (which is a good approximation for most laser scanners), expanding Eq. (16) and using the relationship  $\delta u_k^i = u_k^i - r_k^i$  yields

$$\delta u_k^i = (L_k^i + \varepsilon_l) \varepsilon_\theta \begin{bmatrix} -\sin \Theta_k^i \\ \cos \Theta_k^i \end{bmatrix} + \varepsilon_l \begin{bmatrix} \cos \Theta_k^i \\ \sin \Theta_k^i \end{bmatrix} \quad (17)$$

Assuming that  $\varepsilon_\theta$  and  $\varepsilon_l$  are independent, then:

$${}^N P_k^i = E[\delta u_k^i (\delta u_k^i)^T] = \frac{(L_k^i)^2 \sigma_\theta^2}{2} \begin{bmatrix} 2 \sin^2 \Theta_k^i & -\sin 2\Theta_k^i \\ -\sin 2\Theta_k^i & 2 \cos^2 \Theta_k^i \end{bmatrix} + \frac{\sigma_l^2}{2} \begin{bmatrix} 2 \cos^2 \Theta_k^i & \sin 2\Theta_k^i \\ \sin 2\Theta_k^i & 2 \sin^2 \Theta_k^i \end{bmatrix}$$

For practical computation, we can use  $\theta_k^i$  and  $l_k^i$  as a good estimates for the quantities  $\Theta_k^i$  and  $L_k^i$ .

### 3.2 Correspondence Error

Here we analyze the correspondence error for the general point correspondence method of Section 4. We then derive a second order probabilistic approximation to this error. While our derivation assumes that the sensor beam strikes a locally straight line segment (Fig. 1), the derivation can be extended to other boundary geometries, or serve as a good approximation for modestly curved boundaries.

We first develop a formula for the maximum correspondence error. Consider how points will be matched in the vicinity of points  $u_k^i$  and  $u_k^j$  in Fig. 1. Let

$$\delta_+^i = \|u_{k+1}^i - u_k^i\|, \quad \delta_-^i = \|u_k^i - u_{k-1}^i\| \quad (18)$$

denote the distance to the adjacent scan points (from pose  $i$ 's scan) near the candidate matching point  $u_k^i$  (see Fig. 1). Similarly, let  $\delta_+^j = \|u_{k+1}^j - u_k^j\|$  and  $\delta_-^j = \|u_k^j - u_{k-1}^j\|$  denote the

distances to the adjacent scan points (from pose  $j$ 's scan) near the candidate matching point  $u_k^j$ . The maximum distance (or error) between any pair of points that are chosen to be in correspondence will be half of the minimum distance between adjacent scan points. If the error is greater than this value, the point will be matched to another point, or it will not be matched at all. On average, this error will be the minimum of  $(\delta_+^i + \delta_-^i)/4$  or  $(\delta_+^j + \delta_-^j)/4$ . Simple geometric analysis of Fig. 1 shows that

$$\begin{aligned} \frac{\delta_+^i + \delta_-^i}{4} &= \frac{l_k^i \sin \beta}{4} \left[ \frac{1}{\sin(\alpha_k^i + \beta)} + \frac{1}{\sin(\alpha_k^i - \beta)} \right] \\ &= \frac{l_k^i \sin \beta}{2} \left[ \frac{\sin \alpha_k^i \cos \beta}{\sin^2 \alpha_k^i - \sin^2 \beta} \right] \end{aligned} \quad (19)$$

Substituting  $j$  for  $i$  yields a formula for  $(\delta_+^j + \delta_-^j)/4$ .

We now derive the first two moments of the correspondence error distribution. For simplicity, let the robot be situated so that  $\delta_+^i + \delta_-^i < \delta_+^j + \delta_-^j$  (i.e., the correspondence error is defined by pose  $i$ ). Recall the correspondence error formula of Eq. (6):  $c_k^{ij} = r_k^i - R_{ij} r_k^j - p_{ij}$ . The correspondence error is collinear with the boundary's tangent. Hence, let  $\mu_k^{ij} = c_k^{ij} \cdot t_k$  be the projection of  $c_k^{ij}$  onto the unit boundary tangent vector,  $t_k$ , at  $u_k^i$ . The vector  $t_k$  is positive pointing from  $u_k^i$  to  $u_{k+1}^i$ . Hence,  $\mu_k^{ij}$  is a signed quantity, and  $c_k^{ij} = \mu_k^{ij} t_k$ . Letting  $x$  be the position along the boundary relative to  $u_k^i$ , the correspondence error is locally a function of  $x$ . The expected value (mean) of the error in the interval  $x \in [-\delta_-^i, \delta_+^i]$  is:

$$E[\mu_k^{ij}] = \int_{-\delta_-^i}^{\delta_+^i} \mu_k^{ij}(x) \mathcal{P}(x) dx \quad (20)$$

where  $\mathcal{P}(x)$  is the probability that the  $k^{\text{th}}$  scan point from pose  $j$  will be located at  $x$ . We reasonably assume that  $\mathcal{P}(x)$  has an a priori uniform probability. Hence  $\mathcal{P}(x) = 1/(\delta_+^i + \delta_-^i)$ . Realizing that  $\mu_k^{ij}(x) = x$  in the interval  $[-\delta_-^i, \delta_+^i]$ , evaluation of Eq. (20) yields:

$$E[\mu_k^{ij}] = \frac{(\delta_+^i)^2 - (\delta_-^i)^2}{\delta_+^i + \delta_-^i} = \delta_+^i - \delta_-^i = -2 \frac{l_k^i \sin^2 \beta \cos \alpha_k^i}{\sin^2 \alpha_k^i - \sin^2 \beta}$$

Note that when the incidence angle is not normal ( $\alpha_k^i \neq 90^\circ$ ), the mean is non-zero. However, since the mean is proportional to  $\sin^2 \beta$ , this term is negligible when  $\beta$  is small. Hence, we can practically consider the correspondence error to be a zero-mean quantity when  $\beta$  is small (this holds for the experiments described in Section 6). To compute the variance of the correspondence error (using the zero-mean assumption),

$$E[(\mu_k^{ij})^2] = \int_{-\delta_-^i}^{\delta_+^i} \frac{x^2}{\delta_+^i + \delta_-^i} dx = \frac{(\delta_+^i)^3 + (\delta_-^i)^3}{3(\delta_+^i + \delta_-^i)} \quad (21)$$

Letting  $\eta_k^i = \alpha_k^i + \theta_k^i$ , and keeping the above results in mind, the covariance  ${}^C P_k^i$  of Eq. (7) can be found as

$$\begin{aligned} {}^C P_k^i &= E[c_k^{ij} (c_k^{ij})^T] = E[(\mu_k^{ij})^2] t_k t_k^T \\ &= E[(\mu_k^{ij})^2] \begin{bmatrix} \cos^2(\eta_k^i) & \cos(\eta_k^i) \sin(\eta_k^i) \\ \cos(\eta_k^i) \sin(\eta_k^i) & \sin^2(\eta_k^i) \end{bmatrix} \end{aligned} \quad (22)$$

### 3.3 Measurement Bias Effects

Range measurement bias is an artifact of some range sensing methods (e.g., see [12]). Since bias models will strongly depend upon the given range sensing method, it is not possible to give a complete summary of bias models for common sensing methods. Instead, we consider the effect of bias on the displacement estimate.

To analyze the bias effect, let  $\varepsilon_k^{ij} \triangleq \tilde{\varepsilon}_k^{ij} + \tilde{o}_k^{ij}$ . I.e.,  $\tilde{\varepsilon}_k^{ij}$  represents the matching error if one ignores the bias offsets, and  $\tilde{o}_k^{ij} = o_k^i - R_{ij} o_k^j$  is the total bias offset effect at the  $k^{\text{th}}$  correspondence. Incorporating the offsets, the likelihood function takes the form:

$$\mathcal{L}(\{\varepsilon_k^{ij}\} | g_{ij}) = \prod_{k=1}^{n_{ij}} \frac{e^{-\frac{1}{2}(\varepsilon_k^{ij} - \tilde{o}_k^{ij})^T (P_k^{ij})^{-1} (\varepsilon_k^{ij} - \tilde{o}_k^{ij})}}{2\pi \sqrt{\det P_k^{ij}}} \quad (23)$$

Following the derivations that lead to Prop. 1, one can show that the translation estimate in this case is:

$$\hat{p}_{ij} = P_{pp} \left( \sum_{k=1}^{n_{ij}} (P_k^{ij})^{-1} (u_k^i - q_k + \tilde{o}_k^{ij}) \right) \quad (24)$$

One can interpret this result as follows. If a range sensor does suffer from bias offset, ignoring the offset will adversely affect the estimate. However, bias models can be used to compensate for bias effects in the estimate.

### 4 Selection of point correspondences

We select point correspondences following a methodology similar to the one in [9]. Given two scan sets  $\{u_k^i\}$  and  $\{u_k^j\}$ , the outliers are removed in the first step. These are the points visible in one scan, but not in the other. After removing the outliers, we attempt to find correspondences between scan point pairs in the two poses. For every point in pose  $i$ , we search for a corresponding scan point in pose  $j$  that satisfies a *range criterion*: the corresponding point must lie within a given distance:  $\|u_k^i - u_k^j\| < d$ . If no points in pose  $j$  satisfy this criteria, then the point is marked as having no correspondence. The parameter  $d$  is initially set at a value proportional to the odometry error for the step. As the matching iterations proceed,  $d$  is monotonically reduced to a value of the order of the maximum point error predicted by our noise model in order to speed convergence.

### 5 Estimating the Incidence Angle

The correspondence error model of Section 3.2 assumes knowledge of each scan point's incidence angle. To estimate this angle, the neighboring boundaries are approximated by fitting straight line segments to the range data employing a Hough transform. In this general line finding technique, each scan point  $\{x_k, y_k\}$  is transformed into a discretized curve in the

Hough space. The transformation is based on the parametrization of a line in polar coordinates with a normal distance to the origin,  $d_L$ , and a normal angle,  $\phi_L$ .

$$d_L = x_k \sin(\phi_L) + y_k \cos(\phi_L) \quad (25)$$

Values of  $\phi_L$  and  $d_L$  are discretized with  $\phi_L \in \{0, \pi\}$  and  $d_L \in \{-D, D\}$  where  $D$  is the maximum sensor distance reading. The Hough space is the array of discrete cells, where each cell corresponds to a single line in the scan point space. For each scan point, the Hough space cells which correspond to lines passing through that point are incremented. Peaks in the Hough space correspond to lines in the scan data set. When the cells in the Hough space are incremented, we record the scan point coordinate in the cell, so when a peak is determined, the set of cells that make up that peak contain the set of points that contributed to that line. The incidence angles can then be found for every point contributing to a line. The Hough transform can be generalized to detect and fit simple curves, but for most indoor environments the line fitting method is sufficient.

## 6 Experiments

We implemented our method on a Nomadics 200 mobile robot equipped with a Sick LMS-200 laser range scanner. For comparison, we also implemented an unweighted least squares scan matching algorithm (analogous to that of Lu and Milios [9], but with an improved point correspondence algorithm), hereafter called the "UWLS." This section summarizes our experimental findings on our algorithm's absolute and relative performance. In our experiments, we used the values  $\beta = 0.5^\circ$ ,  $\sigma_l = 5$  mm,  $\sigma_\theta = 10^{-4}$  radians obtained from the Sick LMS-200 laser specifications.

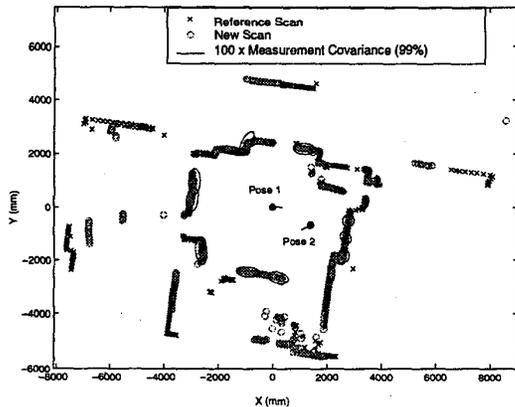


Figure 2: Scan points from two poses.

Fig. 2 shows matched scans taken at two poses inside our laboratory (at randomly selected boundary points, ellipsoids (scaled by factor of 100) indicate the 99% confidence region of the point pairing covariances). Fig. 3 shows the line segments fitted to the pose 1 data.

Fig. 4 graphically depicts the convergence properties of our algorithm and its comparison with the UWLS on the data set of

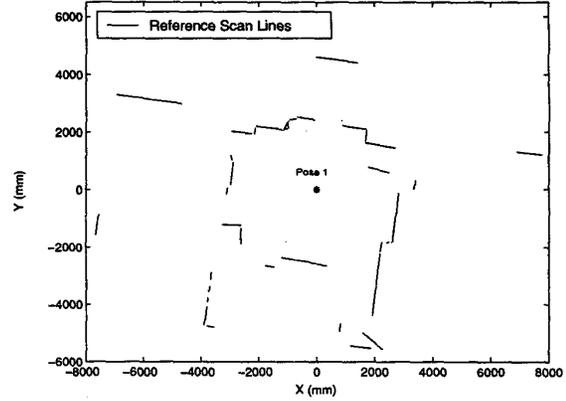


Figure 3: Line segments fitted to Fig. 2

Fig. 2. In this experiment, the robot displaced a manually measured distance of  $(\Delta x_{12}, \Delta y_{12}, \Delta \phi_{12}) = (1381$  mm,  $-690$  mm,  $2.06$  rad). To test the algorithm's robustness to poor odometry estimates, and its tightness of convergence, we added random noise (up to 200 mm in  $\Delta x$ ,  $\Delta y$ , and 0.1 rad in  $\Delta \phi$ ) to the vehicle's actual odometric displacement estimate to derive 1000 different starting points for the algorithms' iterations. Convergence was deemed successful if the change in error estimate between successive iterations remained below 0.05% for three iterations.

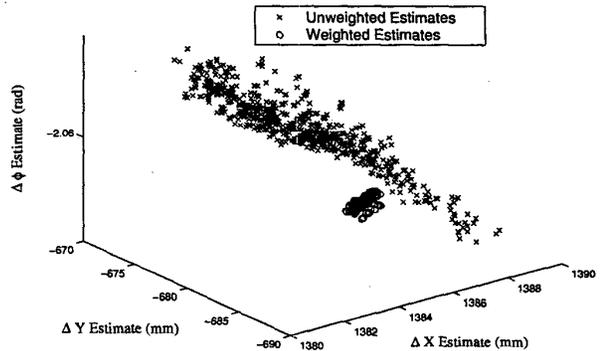
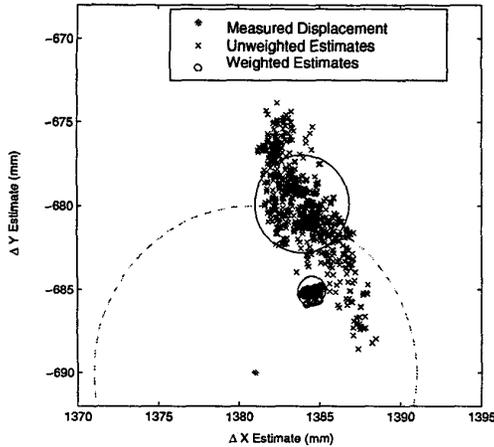


Figure 4: Convergence comparison of Unweighted (blue crosses) and our Weighted (red circles) algorithm for 1000 random initial estimates.

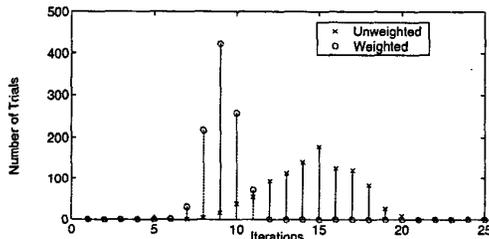
The blue crosses in Fig. 4 show the final estimates of the UWLS algorithm after its convergence, while the red circles show the final estimates of our weighted algorithm. Figure 5 shows the projection of this data onto the  $(\Delta x, \Delta y)$  axes. From a wide range of initial conditions, our algorithm converges to a very tight cluster of displacement estimates. These visibly better convergence properties of our algorithm suggest that it is more robust to errors in initial estimates, such as given by odometry. Moreover, our algorithm is absolutely more accurate. Its mean estimate has total translational and angular errors of 6.0 mm and 0.001 rad, while the mean UWLS estimate is in error by 10.7



**Figure 5:** Projection of Fig. 4 onto  $\Delta x_{12}$  and  $\Delta y_{12}$  coordinates. The solid circle and ellipse are the 99% confidence regions of the two algorithms' estimated covariances. The green dot and dashed circle are the manually measured displacement and its maximum error.

mm and 0.0013 rad. Some of the UWLS estimates were up to 16 mm in error.

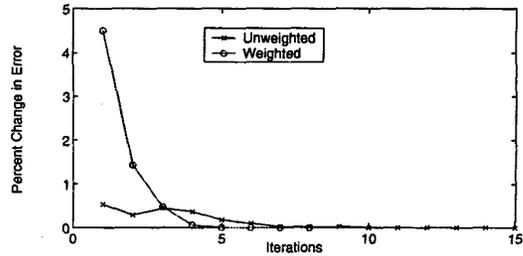
The solid large circle of Fig. 5 is centered at the mean UWLS ( $\Delta x_{12}, \Delta y_{12}$ ) estimate, and it circumscribes the 99% confidence region corresponding to the UWLS covariance estimate (see [14] for the UWLS covariance formula). The lower ellipse circumscribes the 99% confidence region for our algorithm's covariance estimate, as computed in Prop.3. The smaller size of our covariance estimate shows that in the presence of perfect point correspondences, our algorithm should potentially produce tighter displacement estimates, as it properly takes all of the noise factors into account. In cases where the point correspondences are uncertain, our algorithm may have a larger covariance than the UWLS algorithm. However, the UWLS covariance estimate will be overly optimistic in these cases.



**Figure 6:** Number of iterations of Unweighted and our Weighted algorithm.

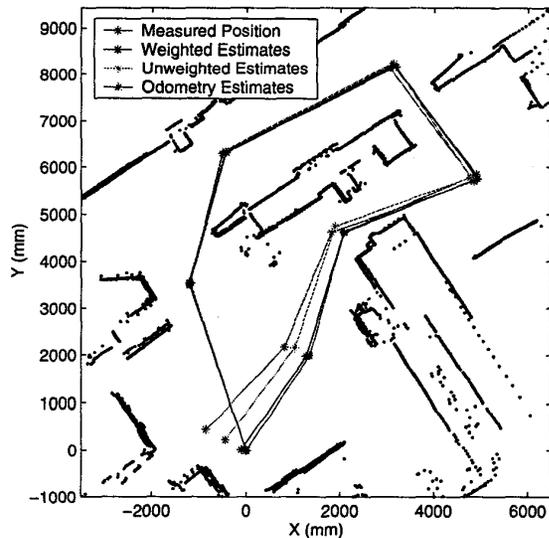
Figs. 6 and 7 provide another view of the convergence process. For the same data set, Fig. 6 shows how many iterations were respectively required by our algorithm and the UWLS. Our algorithm required roughly 40% fewer iterations on average to reach the same convergence criteria. Fig. 7 compares the rate of convergence for one particular initial condition.

Finally, Fig. 8 shows an eight-step robot path superimposed on



**Figure 7:** Convergence of Unweighted vs. Weighted algorithm.

the acquired range data. The total path length was 21.8 meters. The ratio of the final translation error to total path length is .6% for our weighted algorithm, 2.4% for the UWLS, and 4.5% for odometry. Fig. 9 shows a detail of the final position estimates for our algorithm, the UWLS, as well as the actual position. The ellipses in this figure denote the 99% confidence regions of the covariance estimates of our algorithm and the UWLS algorithm. Fig. 10 plots the total cumulative position estimation error for odometry, the UWLS, and our weighted algorithm. Figs. 9 and 10 show that over this multi-step path, our method provides significantly greater estimation accuracy. Moreover, the covariance estimates in Fig. 9 show that the UWLS provides an extremely optimistic covariance estimate, whereas the covariance estimate of our algorithm includes the actual error within its confidence region.



**Figure 8:** Multi-step path.

## 7 Conclusion

This paper investigated the effects of different error and noise sources on the convergence and accuracy properties of motion from structure algorithms. Our experiments showed that careful attention to the details of error modelling can significantly en-

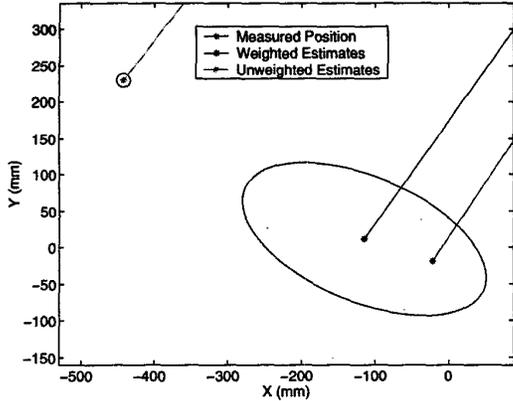


Figure 9: Detail of final position estimates.

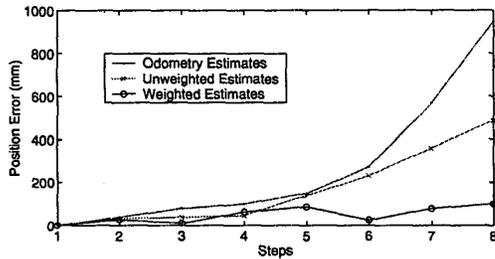


Figure 10: Cumulative position error along path.

hance overall displacement and covariance estimation accuracy. Although the analysis was mainly aimed at laser range sensors, the methods can likely be extended to other range sensors, such as stereo cameras, radar, ultrasound, etc. The specifics of our analysis must be modified to incorporate the appropriate error/noise models for each particular sensor.

**Acknowledgments:** This research has been sponsored in part by a National Science Foundation Engineering Research Center grant (NSF9402726) and NSF ERC-CREST partnership award EEC-9730980. We thank Akash Kumar and Ada Yu for helping to implement the UWLS algorithm, and Aisha Chambliss and Derek Jackson for hardware support.

### A Weighted Translation Solution

Recall the log-likelihood formula of Eq. (11). Since  $D^{ij}$  is independent of  $\Delta x_{ij}$  and  $\Delta y_{ij}$ , the necessary condition for an extremal in the log-likelihood function with respect to these variables is:

$$\frac{\partial M^{ij}}{\partial \Delta x_{ij}} = \frac{\partial M^{ij}}{\partial \Delta y_{ij}} = 0. \quad (26)$$

Starting with Eq. (26), and noting that  $\partial M^{ij} / \partial \Delta x = -[1 \ 0]^T$  and  $\partial M^{ij} / \partial \Delta y = -[0 \ 1]^T$ , we get

$$\begin{bmatrix} \frac{\partial M^{ij}}{\partial \Delta x} \\ \frac{\partial M^{ij}}{\partial \Delta y} \end{bmatrix} = - \sum_{k=1}^{n_{ij}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} (P_k^{ij})^{-1} \epsilon_k^{ij} = 0$$

or

$$\sum_{k=1}^{n_{ij}} (P_k^{ij})^{-1} (u_k^i - R_{ij} u_k^j - p_{ij}) = 0$$

Rearranging this formula results in Eq. (12).

### References

- [1] S. I. Roumeliotis and J. W. Burdick, "Stochastic cloning: A generalized framework for processing relative state measurements," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington D.C., May 11-15 2002, (this issue).
- [2] S. Atiya and G.D. Hager, "Real-time vision-based robot localization," *IEEE Trans. on Robotics and Automation*, vol. 9, pp. 785-800, Dec. 1993.
- [3] J. Leonard and H. Durrant-Whyte, "Mobile robot localization by tracking geometry beacons," *IEEE Trans. on Robotics and Automation*, vol. 7, no. 3, pp. 376-382, June 1991.
- [4] J. Neira, J.D. Tardós, J. Horn, and G. Schmidt, "Fusing Range and Intensity Images for Mobile Robot Localization," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 1, pp. 76-84, Feb. 1999.
- [5] S.I. Roumeliotis and G.A. Bekey, "Bayesian estimation and Kalman filtering: A unified framework for Mobile Robot Localization," in *Proc. IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, April 24-28 2000, pp. 2985-2992.
- [6] S. Thrun, D. Fox, and W. Burgard, "A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robots," *Machine Learning*, vol. 31, pp. 29-53, 1998.
- [7] M.C. Amann, T. Bosch, M. Lescure, R. Myllylä, and M. Rioux, "Laser Ranging: A critical review of usual techniques for distance measurement," *Opt. Eng.*, vol. 40, no. 1, pp. 10-19, Jan. 2001.
- [8] J.L. Crowley, "World modeling and position estimation for a mobile robot using ultrasonic ranging," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1989, pp. 674-680.
- [9] F. Lu and E. Milios, "Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans," *J. of Intelligent and Robotic Systems*, vol. 20, pp. 249-275, 1997.
- [10] I.J. Cox, "Blanche—An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle," *IEEE Trans. on Robotics and Automation*, vol. 2, pp. 193-204, 1991.
- [11] M.D. Adams, "Lidar Design, Use, and Calibration Concepts for Correct Environmental Detection," *IEEE Trans. Robotics and Automation*, vol. 16, no. 6, pp. 753-761, Dec. 2000.
- [12] M.D. Adams and P.J. Probert, "The Interpretation of Phase and Intensity Data from AMCW Light Detection Sensor for Reliable Ranging," *Int. J. of Robotics Research*, vol. 15, no. 5, pp. 441-458, Oct. 1996.
- [13] O. Bengtsson and A.J. Baereldt, "Localization in Changing Environments— Estimation of Covariance Matrix for the IDC algorithm," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Maui, Hawaii, Oct. 2001.
- [14] F. Lu and E. Milios, "Globally Consistent Range Scan Alignment for Environment Mapping," *Autonomous Robots*, vol. 4, pp. 333-349, 1997.
- [15] S. I. Roumeliotis, "Dense range feature matching: Weighted rotational displacement estimation," Tech. Rep., California Institute of Technology, 2001, Available at: [http://robotics.caltech.edu/~stergios/tech\\_reports/tr\\_wlsm-orientation.pdf](http://robotics.caltech.edu/~stergios/tech_reports/tr_wlsm-orientation.pdf).