Proceedings of the 2003 IEEE
International Conference on Robotics & Automation
Taipei, Taiwan, September 14-19, 2003

Weighted Line Fitting Algorithms for Mobile Robot Map Building and Efficient Data Representation

Samuel T. Pfister, Stergios I. Roumeliotis, Joel W. Burdick
{sam,stergios,jwb}@robotics.caltech.edu
Mechanical Engineering, California Institute of Technology, Pasadena, CA 91125

***Abstract*— This paper presents an algorithm to find the line-based map that best fits sets of two-dimensional range scan data. To construct the map, we first provide an accurate means to fit a line segment to a set of uncertain points via a maximum likelihood formalism. This scheme weights each point's influence on the fit according to its uncertainty, which is derived from sensor noise models. We also provide closed-form formulas for the covariance of the line fit, along with methods to transform line coordinates and covariances across robot poses. A Chi-sqared based criterion for "knitting" together sufficiently similar lines can be used to merge lines directly (as we demonstrate) or as part of the framework for a line-based SLAM implementation. Experiments using a Sick LMS-200 laser scanner and a Nomad 200 mobile robot illustrate the effectiveness of the algorithm.**

I. INTRODUCTION

Mobile robot localization and mapping in unknown environments is a fundamental requirement for effective autonomous robotic navigation. A key issue in the practical implementation of localization and mapping schemes concerns how map information is represented, processed, stored, updated, and retrieved. A number of different solutions to this problem are used in practice. In one approach, the map consists of all the raw sensor data samples that have been gathered, for example [1]. In another approach, a map is a collection of features which must be robustly extracted from the sensor data, for example [2]. These methods represent some of the possible trade-offs between the simplicity and efficiency of the map representation, the computational complexity of the localization procedure, and the map's overall accuracy and self-consistency.

This paper introduces some useful algorithms for creating line-based maps from sets of dense range data that are collected by a mobile robot from multiple poses. First, we consider how to accurately fit a line segment to a set of uncertain points. For example, Fig. 1 shows actual laser scan data points, and the uncertainty of these data points, as calculated using the methods of Section II. Our fitting procedure weights each point's influence on the overall fit according to its uncertainty. The point's uncertainty is in turn derived from sensor noise models. These models, which were first presented in [3], are briefly reviewed. We also provide closed-form formulas for the covariance of the line fit (see Fig. 1). This measure of uncertainty allows one to judge the quality of the fit. It can also be used in subsequent localization and navigation tasks that are based on the line-maps. Next we show how to "knit" together line segments across multiple range scan data sets, while taking the uncertainty of the robot's configuration into account. This leads to further efficiencies in the map's representation.

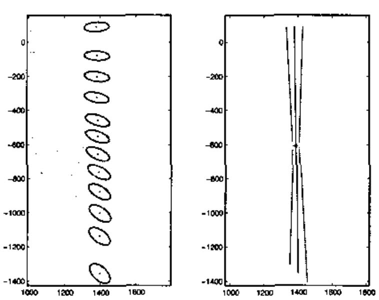

Fig. 1. Example of line segment fit: data points (left) and fitted line with a representation of its uncertainty (right).

A line segment is a simple feature. Hence, line-based maps represent a middle ground between highly reduced feature maps and massively redundant raw sensor-data maps. Clearly, line-based maps are most suited for indoor applications, or structured outdoor applications, where straight edged objects comprise many of the environmental features. The line segments produced by our algorithm can be used in a number of ways. They can replace the raw range scan data to efficiently and accurately represent a global map. This is a form of map compression. The sets of segments can be input to another algorithm that extracts high level features such as doors or corners. The line segments can be used as part of or all of the local map representation at the core of a SLAM algorithm. They can be used for subsequent localization operations (e.g., solving the "kidnapped robot" problem). Or, they can be used for motion planning operations.

The idea of fitting lines to range data is not a new one. The solution to the problem of fitting a line to a set of uniformly weighted points can be found in textbooks (e.g., [4],[5]). Others have presented algorithms for extracting line segments from range data (e.g. [6], [7], [8]). Since the algorithms do not incorporate noise models of the range data, the fitted lines do not have a sound statistical interpretation. Several authors have used the Hough Transform to fit lines to laser scan or sonar data (e.g. [9], [10], [11]). The Hough Transform alone does not take noise and uncertainty into account when estimating the line parameters.

The recursive mode of the Kalman-Filter was used to extract and fit line segments to groups of noisy pixels by Ayache and Faugeras in [12] and has since been applied to range data in [13] and [14]. The methods in [12] and [13] both specify constant weighting for all point contributions. Castellanos and Tardos in [14] account for the individual point uncertainties in estimating the parameters of the line. However, they choose to calculate the covariance of the line parameters using an ad-hoc approach that uses only the uncertainty of the line segment endpoints, and ignores the uncertainty contribution of the interior points. To our knowledge, the line fitting procedure presented here for a polar line representation in the case of range data with varied uncertainty appears to be new. A key feature and contribution of our approach are the concrete formulas for the covariance of the line segment fits while allowing for individual weighting of each measured point. These covariances allow other algorithms that use the line-maps to appropriately interpret and incorporate the line-segment data. Ayache and Faugeras, as well as Castellanos and Tardos also present methods to merge line segments across multiple scans using a Kalman filter. Our more accurate covariance estimate should allow for better line merging in a statistically sound fashion. Addtionally, our line parametrization allows for comparison and merging of intermittently interrupted line segments.

Our approach is based on the following assumptions. The robot operates in a planar environment, and is equipped with a 2-dimensional sensor that provides dense range measurements (such as a laser scanner). The robot moves through multiple poses, $g_1, g_2, \dots, g_n$, where g_k represents the robot's k^{th} pose, $g_k = (x_k, y_k, \theta_k)$, relative to a fixed reference frame. At each pose the robot gathers a range scan. The scan point coordinates are described in the robot's body frame, and the k^{th} scan point in pose i takes the form:

$$u_k^i = d_k^i \begin{bmatrix} \cos\phi_k^i \\ \sin\phi_k^i \end{bmatrix} \tag{1}$$

where d_k^i is the measured distance to the environment's boundary in the direction denoted by ϕ_k^i. We also assume that a covariance estimate, Q_k^i, is available for the uncertainty in this scan point's position (See Section II for details).

Additionally, for the purpose of "knitting" line segments together across different scan sets gathered from different poses, the robot must possess an estimate of its displacement, $\hat{g}_{ij}$ between poses i and j (where $g_{ij} = g_i^{-1} g_j$). This can be done via odometry, matching of the range scans, or other means. We also assume that one can estimate the covariance, P^{ij}, of the displacement estimate $\hat{g}_{ij}$, and it has the form:

$$P^{ij} = \begin{bmatrix} P_{pp} & P_{p\phi} \\ P_{\phi p} & P_{\phi\phi} \end{bmatrix} \tag{2}$$

where the 2×2 matrix P_{pp} describes the uncertainty in the translational estimate, the scalar $P_{\phi\phi}$ describes the uncertainty in orientation, and $P_{\phi p}^T = P_{p\phi}$ describes cross coupling effects. For example, in [3] we presented an algorithm for estimating the robot's displacement by matching range scans, and gave explicit formulas for the terms in Eq. (2). In the simplest case, the displacement estimate is uncorrelated with the range scan (e.g., it is derived from odometry). However, when the displacement estimate is partially or fully derived from the range data, the covariance estimate may be correlated with range scan data uncertainty. These dependencies must be taken into account (Section V).

This paper is structured as follows. Section II reviews the range measurement error models of [3]. Section III describes the weighted line fitting problem and our solution. Section IV reviews the use of the Hough Transform to estimate an initial guess of the line's parameters. Section V describes how to merge lines across data gathered in different robot poses. Experiments in Section VI demonstrate our algorithm's effectiveness.

II. SENSOR NOISE MODELS

Range sensors can be subject to both random noise effects and bias. For a discussion of bias, see [3]. Here we briefly review a general model for measurement noise. Recall the polar representation of scan data, Eq. (1). Let the range measurement, d_k^i, be comprised of the "true" range, $\mathcal{D}_k^i$, and an additive noise term, ε_d:

$$d_k^i = \mathcal{D}_k^i + \varepsilon_d. \tag{3}$$

The noise ε_d is assumed to be a zero-mean Gaussian random variable with variance σ_d^2 (see e.g., [15] for justification). Also assume that error exists in the measurement of ϕ_k^i, i.e. the actual scan angle differs (slightly) from the reported or assumed angle. Thus,

$$\phi_k^i = \Phi_k^i + \varepsilon_\phi, \tag{4}$$

where Φ_k^i is the "true" angle of the k^{th} scan direction, and ε_ϕ is again a zero-mean Gaussian random variable with variance σ_ϕ^2. Hence:

$$u_k^i = (\mathcal{D}_k^i + \varepsilon_d) \begin{bmatrix} \cos(\Phi_k^i + \varepsilon_\phi) \\ \sin(\Phi_k^i + \varepsilon_\phi) \end{bmatrix} . \tag{5}$$

Generally, we can think of the scan point u_k^i as made up of the true component, $\mathcal{U}_k^k$, and the uncertain component, δu_k^i:

$$u_k^i = \mathcal{U}_k^i + \delta u_k^i. \tag{6}$$

If we assume that $\varepsilon_\phi \ll 1^o$ (which is a good approximation for most laser scanners), expanding Eq. (5) and using the relationship $\delta u_k^i = u_k^i - \mathcal{U}_k^i$ yields

$$\delta u_k^i = (\mathcal{D}_k^i + \varepsilon_d)\varepsilon_\phi \begin{bmatrix} -\sin\Phi_k^i \\ \cos\Phi_k^i \end{bmatrix} + \varepsilon_d \begin{bmatrix} \cos\Phi_k^i \\ \sin\Phi_k^i \end{bmatrix} . \quad (7)$$

Assuming that ε_θ and ε_d are independent, the covariance of the range measurement process is:

$$Q_k^i \triangleq E[\delta u_k^i (\delta u_k^i)^T] = \frac{(\mathcal{D}_k^i)^2 \sigma_\phi^2}{2} \begin{bmatrix} 2\sin^2\Phi_k^i & -\sin 2\Phi_k^i \\ -\sin 2\Phi_k^i & 2\cos^2\Phi_k^i \end{bmatrix} + \frac{\sigma_d^2}{2} \begin{bmatrix} 2\cos^2\Phi_k^i & \sin 2\Phi_k^i \\ \sin 2\Phi_k^i & 2\sin^2\Phi_k^i \end{bmatrix} . \quad (8)$$

For practical computation, we can use ϕ_k^i and d_k^i as a good estimates for the quantities Φ_k^i and $\mathcal{D}_k^i$.

The following analysis assumes that the covariance Q_k^i of the k^{th} range measurement in the i^{th} pose can be found. It can arise from Eq. (8), or from other considerations.

III. THE WEIGHTED LINE FITTING PROBLEM

This section describes the weighted line fitting problem and its general solution. We first consider a set of range data taken from a single pose. Section V considers how to "knit" together line segments across multiple poses.

The range data from scan i is first sorted into subsets of roughly collinear points using the well known Hough Transform (see Section IV). These range measurements are uncertain, as described in Section II.

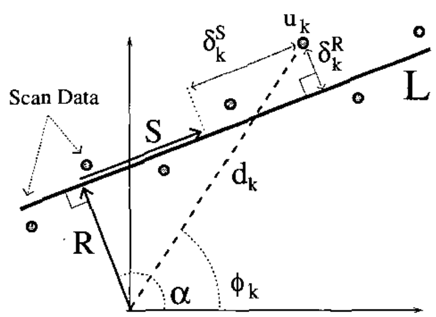

Fig. 2. Geometry of candidate line and data errors

We chose a three parameter representation of a line segment $L(R, \alpha, S)$ with (R, α) the vector to the normal of the infinite line and S the position of the weighted mean of the contributing points along the line. The S measurement represents the effective center of the line segment tangential to the infinite line.

We define a coordinate frame RS associated with line L as the coordinate frame of the robot rotated by α with the $\vec{R}$ direction perpendicular to the line and the $\vec{S}$ direction parallel to the line. The coordinates of the kth range measurement u_k with respect to the RS reference frame are calculated as follows:

$$u_k^R = d_k^i \cos(\alpha - \phi_k) \quad (9)$$
$$u_k^S = d_k^i \sin(\alpha - \phi_k) \quad (10)$$

The distances between u_k and line L are therefore

$$\delta_k^R = u_k^R - R \quad (11)$$
$$\delta_k^S = u_k^S - S \quad (12)$$

with

$$\delta_k = \begin{bmatrix} \delta_k^R \\ \delta_k^S \end{bmatrix} . \quad (13)$$

The goal of the line fitting algorithm is to find the line $L(R, \alpha, S)$ that minimizes the errors δ_k^R and δ_k^S in a suitable way over the set of measurements.

In our approach the contribution of each of the virtual errors is weighted according to its modelled uncertainty. Note that due to the symmetry of a line and our choice of reference frame, only the errors δ_k^R must be minimized to fit the infinite line (R, α) while the only errors δ_k^S are minimized to calculate the parameter S. It follows that for the purposes of line-fitting, the errors in the $\vec{R}$ and $\vec{S}$ directions can be considered uncorrelated. We then derive the scalar covariances of the virtual errors in the RS reference frame as:

$$P_k^R = Q_{11}\cos^2\alpha + 2Q_{12}\sin\alpha\cos\alpha + Q_{22}\sin^2\alpha \quad (14)$$
$$P_k^S = Q_{11}\sin^2\alpha - 2Q_{12}\sin\alpha\cos\alpha + Q_{22}\cos^2\alpha \quad (15)$$

where Q_k^i is a 2×2 symmetric covariance matrix defined in (8) and Q_{ij} are the matrix elements. The covariance matrix for the contribution of point u_k is then:

$$P_k = \begin{bmatrix} P_k^R & 0 \\ 0 & P_k^S \end{bmatrix} . \quad (16)$$

Maximum Likelihood Formulation. We use a maximum likelihood approach to formulate a general strategy for estimating the best fit line from a set of nonuniformly weighted range measurements. Alternatively, one could estimate the line parameters using an extended Kalman filter (EKF). In theory, the EKF and our maximum likelihood approach will yield the same results when applied to systems with Gaussian noise. However, as shown in the propositions below, this problem has enough mathematical structure that the maximum likelihood analysis leads to simple equations that are more efficient to compute than the Kalman filter approach. Additionally, our experience has shown that this algorithm is less sensitive to poorer guesses of the initial conditions.

Let $\mathcal{L}(\{\delta_k\}|L)$ denote the *likelihood function* that captures the likelihood of obtaining the errors $\{\delta_k\}$ given a line L and a set of points. If the $k = 1, \ldots, n$ range measurements are assumed to be independent (which is usually a sound assumption in practice), the likelihood can be written as a product:

$$\mathcal{L}(\{\delta_k\}|L) = \mathcal{L}(\delta_1|L)\mathcal{L}(\delta_2|L)\cdots\mathcal{L}(\delta_n|L).$$

Recall that the measurement noise is assumed to arise from zero-mean Gaussian processes, and that δ_k is a function of zero-mean Gaussian random variables. Thus, $\mathcal{L}(\{\delta_k\}|L)$ takes the form:

$$\mathcal{L}(\{\delta_k\}|L) = \prod_{k=1}^{n} \frac{e^{-\frac{1}{2}(\delta_k)^T(P_k)^{-1}\delta_k}}{2\pi\sqrt{\det P_k}} = \frac{e^{-M}}{D} \quad (17)$$

$$\text{where} \quad M = \frac{1}{2}\sum_{k=1}^{n}(\delta_k)^T(P_k)^{-1}\delta_k \quad (18)$$

$$D = \prod_{k=1}^{n} 2\pi\sqrt{\det P_k} \quad (19)$$

The optimal estimate of the displacement maximizes $\mathcal{L}(\{\delta_k\}|L)$ with respect to line representation R, α and S. Note that maximizing Eq. (17) is equivalent to maximizing the log-likelihood function:

$$ln[\mathcal{L}(\{\delta_k\}|L)] = -M - ln(D) \quad (20)$$

and from the numerical point of view, it is often preferable to work with the log-likelihood function. Using the log-likelihood formula, we can prove that the optimal estimate of the radial position R and tangential position S of the line can be found as follows [16].

Proposition 1: The weighted line fitting estimate for the line's radial position R and tangential position S is:

$$\begin{bmatrix} R \\ S \end{bmatrix} = \begin{bmatrix} P_{RR} & 0 \\ 0 & P_{SS} \end{bmatrix} \sum_{k=1}^{n} \left(P_k^{-1} \begin{bmatrix} u_k^R \\ u_k^S \end{bmatrix} \right) \quad (21)$$

with

$$P_{RR} = \frac{1}{\sum_{k=1}^{n}\left(\frac{1}{P_k^R}\right)} \quad (22)$$

and

$$P_{SS} = \frac{1}{\sum_{k=1}^{n}\left(\frac{1}{P_k^S}\right)} \quad (23)$$

where u_k^R $u_k^S$and P_k are calculated using Eqs. (9), (10) and (16) with $\widehat{\alpha}$ as the estimated orientation of the line.

There is not an exact closed form formula to estimate α. However, there are two efficient approaches to this problem. First, the estimate of α can be found by numerically maximizing Eq. (17) (or Eq. (20)) with respect to α for a constant R and S calculated according to Prop. 1. This procedure reduces to numerical maximization over a single scalar variable α, for which there are many efficient algorithms. Alternatively, one can develop the following second order iterative solution to this non-linear optimization problem:

Proposition 2: The weighted line fitting estimate for the line's orientation α is updated as $\alpha = \widehat{\alpha} + \delta\alpha$, where:

$$\delta\alpha = -\frac{\sum_{k=1}^{n}\left(\frac{\delta_k^R\delta_k^S}{P_k^R}\right)}{\sum_{k=1}^{n}\left(\frac{(\delta_k^S)^2}{P_k^R}\right)} \quad (24)$$

with δ_k^R, δ_k^S and P_k^R (defined in Eqs. (9), (10) and (14)) calculated using $\widehat{S}$, $\widehat{S}$ and $\widehat{\alpha}$.

Using experimental data, this approximation agrees with the exact numerical solution.

Prop.s 1, and 2 suggest an iterative algorithm for estimating displacement. First an initial guess $\widehat{\alpha}$ for α is determined (see Section IV for details). The estimates $\widehat{R}$ and $\widehat{S}$ are then computed using Prop. 1. The estimates are next employed by Prop. 2 to calculate the current rotational estimate $\widehat{\alpha}$. The improved estimate $\widehat{\alpha}$ is the basis for the next iteration. The iterations stop when a convergence criterion is reached.

Letting $\delta R = R - \widehat{R}$, $\delta S = S - \widehat{S}$ and $\delta\alpha = \alpha - \widehat{\alpha}$ (i.e, line parameter error estimates), a direct calculation yields the following.

Proposition 3: The covariance of the line position is:

$$P_L = \begin{bmatrix} E\{\delta R(\delta R)^T\} & E\{\delta R(\delta\alpha)^T\} & E\{\delta R(\delta S)^T\} \\ E\{\delta\alpha(\delta R)^T\} & E\{\delta\alpha(\delta\alpha)^T\} & E\{\delta\alpha(\delta S)^T\} \\ E\{\delta S(\delta R)^T\} & E\{\delta S(\delta\alpha)^T\} & E\{\delta S(\delta S)^T\} \end{bmatrix}$$

which reduces to

$$P_L = \begin{bmatrix} P_{RR} & P_{R\alpha} & 0 \\ P_{\alpha R} & P_{\alpha\alpha} & 0 \\ 0 & 0 & P_{SS} \end{bmatrix} \quad (25)$$

with

$$P_{\alpha\alpha} = \frac{1}{\sum_{k=1}^{n}\left(\frac{(\delta_k^S)^2}{P_k^R}\right)} \quad (26)$$

$$P_{R\alpha} = -P_{RR}P_{\alpha\alpha}\sum_{k=1}^{n}\left(\frac{\delta_k^S}{P_k^R}\right) \quad (27)$$

and P_{RR} and P_{SS} defined in Eqs. (22) and (23) along with the definitions from Eqs. (11), (12) and (14).
See [16] for a detailed derivation.

Line Segments. The above method estimates the parameters R and α which define an infinite line, and S which determines the center point of the line. Once the optimal infinite line has been found, the relevant line segment bounds are defined by the contributing points with the maximum and minimum values of δ_k^S as calculated from Eq. (12). We retain these scalar end-measurements as well as the scalar variance of each as determined by the value P_k^S calculated from Eq. (15). Because nothing in our representation is dependent on the coordinates of the endpoints, it is trivial for a line to represent two colinear but separated segments. It is simply a matter of retaining multiple end-measurement pairs.

IV. INITIAL ESTIMATES AND GROUPING

Our line fitting method assumes a set of range scan points to be sampled from the same straight line and benefits from an initial guess of the orientation of that line. Given a raw range scan, we first need to detect collinear points and roughly estimate the line through these points. Both of these requirements can be met using the Hough Transform [17]. In this general line finding technique, each scan point $\{d_k^i, \phi_k^i\}$ 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, $\mathcal{R}$, and a normal angle, β.

$$\mathcal{R} = d_k \cos(\beta - \phi_k) \tag{28}$$

Values of $\mathcal{R}$ and β are discretized with $\beta \in \{0, \pi\}$ and $\mathcal{R} \in \{-\mathcal{R}_{max}, \mathcal{R}_{max}\}$ where $\mathcal{R}_{max}$ is the maximum sensor distance reading. The Hough space is the array of discrete cells, where each cell corresponds to a $\{\mathcal{R}, \beta\}$ value and thus a line in the scan point space. For each scan point, parameters $\mathcal{R}$ and β for all lines passing through that point (up to the level of discretization) are computed. Then the cells in Hough space which correspond to these lines are incremented. Peaks in the Hough space correspond to lines in the scan data set. When a cell in the Hough space is incremented, the coordinate of the associated scan point is stored. Hence, when a peak is determined, the set of points that contributed to that line can easily be found. In this way, we can sort range scans into collinear subsets of points and determine an estimate for the line segment orientation.

V. MERGING LINES

This section describes how to merge line segments found in the same scan, or across scans taken at distinct poses. This merging allow compression and simplification of large maps without sacrificing the precision or the knowledge of map uncertainty which we gained from our line fitting algorithm. We consider in detail the process of merging lines across two pose data sets. Merging across multiple data sets is a natural extension. The basic approach is simple. We first transform the candidate line pairs into a common reference frame. We are then able to compare the lines and determine whether they are similar enough to merge using a chi-squared test. Finally we use a maximum likelihood approach to determine the best estimate of the line pairs to be merged.

We first outline methods for transforming both line coordinates and the associated covariance matrix across poses. Clearly if two lines are from the same pose, these transformations are not necessary and one can proceed directly to the merge test. Consider L_1^i and L_2^j found in scans taken at poses i and j respectively.

$$L_1^i = \begin{bmatrix} R_1^i \\ \alpha_1^i \\ S_1^i \end{bmatrix} \quad L_2^j = \begin{bmatrix} R_2^j \\ \alpha_2^j \\ S_2^j \end{bmatrix} \tag{29}$$

We assume that we have an estimate of the robot's pose j with respect to pose i defined as $\widehat{g}_{ij} = [x, y, \gamma]$ and we also have the uncertainty of this measurement P^{ij}. If the measurement $\widehat{g}_{ij}$ is not independent of the range scan measurements (eg. if scan matching is used to calculate $\widehat{g}_{ij}$) then correlation terms need to be calculated that are specific to the measuring technique used. See [16] for more detail. For now we will assume that the measurement $\widehat{g}_{ij}$ is independent of the range scan measurements. To transform the parameters of L_2 from pose i to pose j we calculate:

$$\begin{aligned} L_2^i &= \begin{bmatrix} R_2^i \\ \alpha_2^i \\ S_2^i \end{bmatrix} \\ &= \begin{bmatrix} R_2^j + x\cos(\alpha_2^j + \gamma) + y\sin(\alpha_2^j + \gamma) \\ \alpha_2^j + \gamma \\ S_2^j - x\sin(\alpha_2^j + \gamma) + y\cos(\alpha_2^j + \gamma) \end{bmatrix} \end{aligned} \tag{30}$$

To transform the covariance of L_2^j into the coordinate frame of pose i we derive the following equation

$$P_{L_2}^i = B P_{L_2}^j B^T + K P^{ij} K^T \tag{31}$$

with

$$B = \begin{bmatrix} 1 & -x\sin(\alpha_2^i) + y\cos(\alpha_2^i) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{32}$$

$$K = \begin{bmatrix} \cos(\alpha_2^i) & \sin(\alpha_2^i) & 0 \\ -\sin(\alpha_2^i) & \cos(\alpha_2^i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{33}$$

and with P^{ij} being the covariance of the pose transformation defined in Eq. (2), and P_{L2}^j being the line uncertainty defined in Eq. (25). See [16] for derivation details.

To determine whether a given pair of lines are sufficiently similar to warrant merging, we apply a merge criterion based on the chi-squared test. The coordinates and covariance matrices of the two lines as found by our line fitting algorithm are first represented with respect to a common pose i using the above equations. We then apply the chi-squared test to determine if the difference between two lines is within a threshold defined by the combined uncertainties of the lines. Note that for the merge criterion we are interested in detecting line segments with similar infinite lines so we drop the measurement S from the calculations. The squared Mahalanobis distance D^2 is therefore:

$$D^2 = (\delta L)^T (P_{L_1}^i + P_{L_2}^i)^{-1} \delta L \tag{34}$$

with

$$\delta L = \begin{bmatrix} R_1^i - R_2^i \\ \alpha_1^i - \alpha_2^i \\ 0 \end{bmatrix}$$

and so the merge criterion is

$$D^2 \leq \chi^2 \tag{35}$$

where χ^2 is calculated from a chi-squared table for a two degree of freedom system. If this condition holds, then the lines are considered sufficiently similar to be merged. We can derive the final merged line estimate using a maximum likelihood formulation and can calculate the final merged line coordinates L_m^i and uncertainty $P_{L_m}^i$ with respect to pose i as follows:

$$L_m^i = P_{L_m}^i \left((P_{L_1}^i)^{-1} L_1^i + (P_{L_2}^i)^{-1} L_2^i \right) \tag{36}$$

$$P_{L_m}^i = \left((P_{L_1}^i)^{-1} + (P_{L_2}^i)^{-1} \right)^{-1} \tag{37}$$

When lines are merged, the end-measurements of each line as defined in Section III are projected into the new merged line. If there is no overlap between the pairs then both sets of end-measurements are retained resulting in a multi-segment representation of the line. Otherwise the end-measurments of the new line are set to be the maxima and minima of the end-measurements of the merged lines.

VI. EXPERIMENTS

We implemented our method on a Nomadics 200 mobile robot equipped with a Sick LMS-200 laser range scanner. In our experiments, we used the values $\sigma_d = 5$ mm, $\sigma_\phi = 10^{-4}$ radians obtained from the Sick LMS-200 laser specifications. We set the value of χ^2 to merge lines within the 3σ deviance threshold.

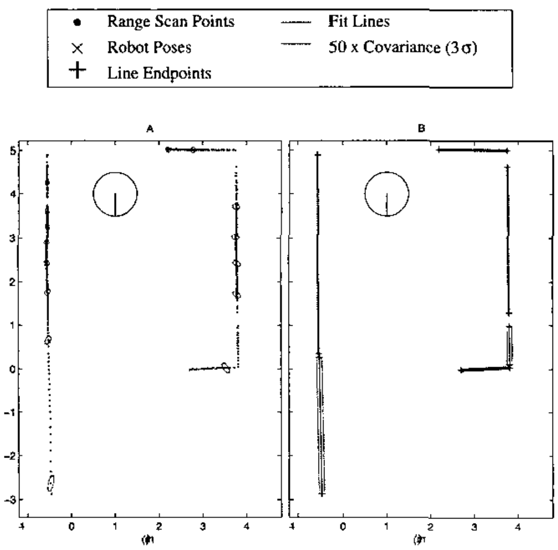

Fig. 3. Range Data – A: Raw points and selected point covariances B: Fit lines and line uncertainties

Fig.s 3, 4, 5 show a sequence of increasingly complex data sets that were gathered in the hallway outside of our laboratory. Fig. 3 graphically depicts the results of fitting lines to a single scan taken in the hallway. The left figure shows the raw range data along with the 3σ confidence region of selected points as calculated from our sensor noise model. The right figure shows the fit lines along with the 3σ confidence region in R. All uncertainty values have been multiplied by 50 for clarity. From the 720 raw range data points our algorithm fit 9 lines. If we assume that a line segment can be represented by the equivalent of two data points, we have effectively compressed the data by 97.%. This compression not only reduces map storage space, but it can also serve to reduce the complexity of any relevant algorithm (eg. scan matching) which scales to the order of number of features. Unlike other feature finders such as corner detectors, the lines abbreviate a large portion of the data set, so overall far less information is lost in compression.

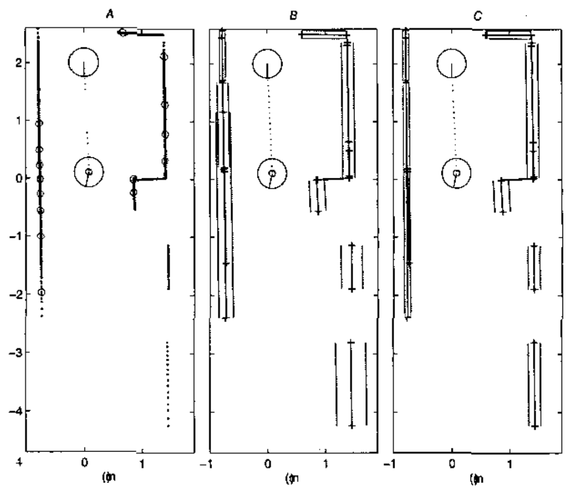

Fig. 4. Range Data From Two Poses – A: Raw points and selected point covariances B: Fit lines and line covariances C: Merged lines and line covariances

Merging lines across scans further improves compression of data. Fig. 4 graphically depicts the results of fitting lines to scans taken at two poses in a hallway. The left figure shows the raw range data, the center figure shows the lines fit to the two scans, and the right figure shows the resulting merged lines. From the 1440 raw range data points our algorithm fit 20 lines without merging, and 14 lines after merging. The merging step compresses the data a further 30% for a total compression of 98.0% from the original data. Note that the three vertical segments on the right are found to be colinear and are merged even though they do not overlap.

Compression achieved by line fitting and merging is equally pronounced in large data sets. Fig. 5 depicts the

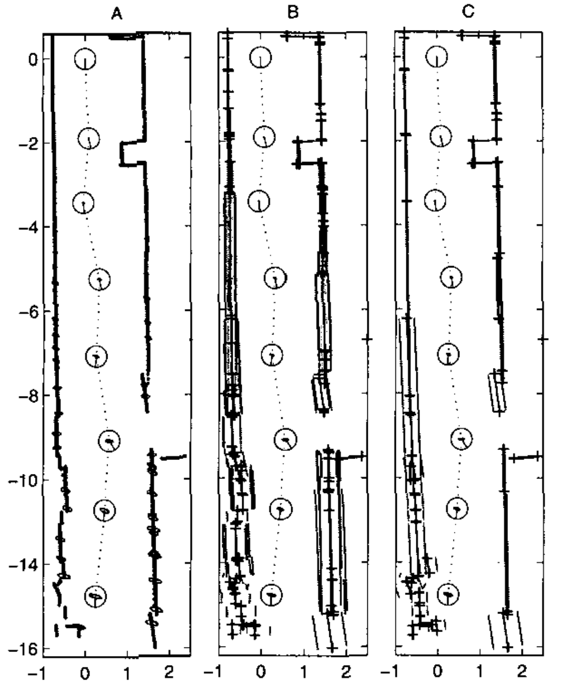

Fig. 5. Range Data From Eight Poses – A) Raw points and selected point covariances B) Fit lines and line covariances C) Merged lines and line covariances

results of fitting lines scans taken at eight poses in the hallway. As above, the left figure shows the raw range data, the center figure shows the lines fit to the ten scans, and the right figure shows the resulting merged lines. From the 5760 raw range data points our algorithm fit 93 lines without merging and 29 lines after merging. The merging step here compresses the data a further 68% for a total compression of 98.9% from the original data. Note that many of the jogs in the lower portion of the hallway arise from recessed doorways, water fountains, and other features. Note also how our method effectively merges the broken line defined by the right wall of the hallway.

Clearly the level of compression depends upon the environment. Hallways will likely have very high compression due to long walls that can be merged over many scans. In more cluttered environments, the compression may not be as high, but it can still be very effective. Fig.s 6, 7 and 8 show the results of fitting lines to range scans taken at ten poses in our laboratory. Fig. 6 shows the raw scan points, Fig. 7 shows the fitted lines, and Fig. 8 shows the resulting merged lines. From the 7200 raw range data points, the algorithm fit 141 lines without merging, and 60 lines with merging. The merging step compresses the data a further 57% for a total compression of 98.9% from the original data.

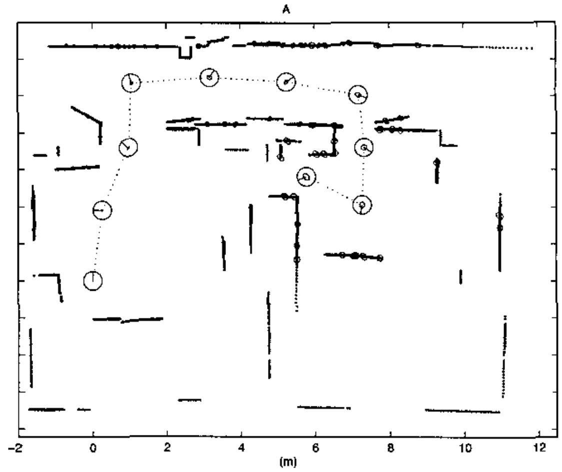

Fig. 6. Raw points and selected point covariances

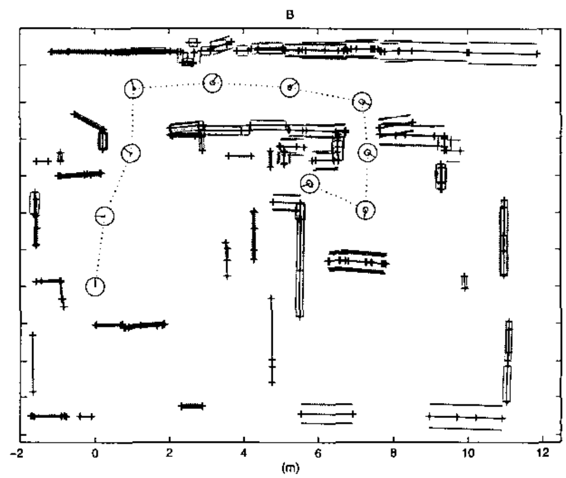

Fig. 7. Fit lines and line covariances

VII. CONCLUSION

This paper outlined a statistically sound method to best fit lines to sets of dense range data. Our experiments showed significant compression in map representation through the fitting and merging of these lines, while still maintaining a probabilistic representation of the entire data set. Future work includes implementation of SLAM and global localization algorithms based on the extracted lines.

Acknowledgments: This research was sponsored in part by a NSF Engineering Research Center grant NSF9402726 and NSF ERC-CREST partnership award EEC-9730980.

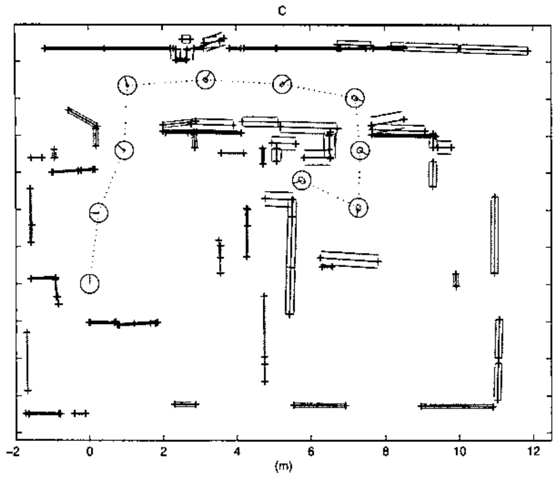

Fig. 8. Merged lines and line covariances

VIII. REFERENCES

[1] 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.

[2] R. Madhavan, H. Durrant-Whyte, and G. Dissanayake, "Natural landmark-based autonomous navigation using curvature scale space," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington D.C., May 11-15 2002.

[3] S.T. Pfister, K.L. Kriechbaum, S.I. Roumeliotis, and J.W. Burdick, "Weighted range sensor matching algorithms for mobile robot displacement estimation," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington, D.C., May 2002.

[4] W.H. Press et. al., *Numerical Recipes in C: the art of scientific computing*, Cambridge Univ. Press, Cambridge, 2nd ed. edition, 1992.

[5] D.A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, Prentice hall, Saddle River, NJ, 1st ed. edition, 2002.

[6] D.M. Mount, N.S. Netanyahu, K. Romanic, and R. Silverman, "A practical approximation algorithm for lms line estimator," in *Proc. Symp. on Discrete Algorithms*, 1997, pp. 473–482.

[7] G.A. Borges and M.J. Aldon, "A split-and-merge segmentation algorithm for line extraction in 2-d range images," in *Proc. 15th Int. Conf. on Pattern Recognition*, Barcelona, Sept. 2000.

[8] J. Gomes-Mota and M.I. Ribeiro, "Localisation of a mobile robot using a laser scanner on reconstructed 3d models," in *Proc. 3rd Portuguese Conf. on Automatic Control*, Coimbra, Portugal, Sept. 1998, pp. 667–672.

[9] P. Jensfelt and H.I. Christensen, "Laser based position acquisition and tracking in an indoor environment," in *Proc. Int. Symp. Robotics and Automation*, 1998.

[10] J. Forsberg, U. Larsson, and A. Wernersson, "Mobile robot navigation using the range-weighted hough transform," *IEEE Robotics and Automation Magazine*, pp. 18–26, March 1995.

[11] L. Iocchi and D. Nardi, "Hough transform based localization of mobile robots," in *Proc. IMACS/IEEE Int. Conf. Circuits, Syst.s, Comm., Computers*, 1999.

[12] N. Ayache and O.D. Fauregas, "Maintainting representations of the environment of a mobile robot," *IEEE Trans. on Robotics and Automation*, 1989.

[13] S.I. Roumeliotis and G.A. Bekey, "Segments: A layered, dual-kalman filter algorithm for indoor feature extraction.," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Takmatsu, Japan, 2000, pp. 454–461.

[14] *Mobile Robot Localization and Map Building: A Multisensor Fusion Approach*, Kluwer Academic Publishers, 1999.

[15] 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.

[16] S.T. Pfister, "Weighted line fitting and merging," Tech. Rep., California Institute of Technology, 2002, Available at:http://robotics.caltech.edu/~ sam/TechReports/LineFit/linefit.pdf.

[17] R.O. Duda and P.E. Hart, "Use of hough transform to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.