2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Sonar Feature Map Building for a Mobile Robot

Hong-Ming Wang, Zeng-Guang Hou, Jia Ma, Yun-Chu Zhang, Yong-Qian Zhang and Min Tan

***Abstract*— This paper presents an approach for sonar feature map building. The approach is composed of extracting features at the data-level fusion stage and fusing the extracted features with the registered features in the map at the feature-level fusion stage. A data-level fusion model, termed three measurements association model (TMAM), has been developed for associating three measurements with a line or a point feature. By use of TMAM, different sets of measurements obtained from a single sonar sensor at consecutive steps are associated with the line and point features. Subsequently, the parameters of the identified features are estimated by use of the iterated least square estimation method. Finally, when a feature is extracted, a simple feature-level fusion strategy is used to update the map. The proposed approach has been tested both in simulation and on real data.**

I. INTRODUCTION

The problem of mobile robot navigation had been summarized into answering the following three questions: "Where am I?", "Where am I going?" and "How should I get there?" [1]. In our views, another question should be added as the first of all. That is "What can I see?". In this paper, we are principally concerned with this question and build a feature-based map using sonar data. Map building is the process of recognizing the environment from sensing information. It mainly comprises three questions: how to represent the sensor information, how to represent the environment and how to map the sensor information into the map.

A. Representations of Sonar Data

Due to the wide beam of the sonar sensor, it is difficult to determine the location of an object from a single sonar reading. Furthermore, specularity makes some sonar readings unexplained. However, Kuc, Siegel, Barshan [2] [3] and Leonard [4] have provided strong evidences that sonar measurements are indeed explicable and predictable in indoor environment. The different views on whether the sonar sensor is an attractive sensor or not lie in the different models used to interpret the sonar readings.

Prior work has specified three models to describe a single sonar reading: centerline model, uniform distribution model and gaussian distribution model. The centerline model assumes that the echo originates from the middle point of the circular arc, without considering angle uncertainty. The uniform or Gaussian distribution model assumes that the location where the echo comes from follows a uniform or Gaussian distribution respectively on the arc. In our opinion, it is safe and reasonable to model the angle uncertainty of a single sonar reading by use of the uniform distribution model. The reason is that, we really don't have enough knowledge to make our decision on the location where a single sonar echo comes.

Hong-Ming Wang, Zeng-Guang Hou, Jia Ma, Yun-Chu Zhang, Yong-qian Zhang and Min Tan are with Key Laboratory of Complex Systems and Intelligence Science, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China. {hongming.wang, zengguang.hou, yunchu.zhang, yongqian.zhang, jia.ma, min.tan}@ia.ac.cn

B. Representations of Environment

Besides semantic and topological maps [5], there are two different approaches to represent the environment at different levels. One is feature-based map, and the other is grid-based map [6]. In the grid-based map, the environment is divided into a discrete, two or three-dimensional grid of cells. Each cell is assigned a single value between 0 and 1 to represent the probability that the cell is unknown, occupied or free space. In the feature-based map, the environment is modeled by a set of geometric primitives such as line segments and points. Grid-based representations make weaker assumption about the environment than feature-based approaches. Also, computational requirements are much less sensitive to environment complexity. However they are less powerful than the geometric models for the purpose of position estimation. The sonar sensors can detect four types of targets in indoor environment: planes, cylinders, corners and edges [4]. Different types of targets return the transmitted pluses in different ways. We refer to corners and edges as point features, cylinders as arc features and planes as line features. In this paper, we consider only the line and point features.

C. Mapping Sensor Measurements into Map

Crowly [7] developed one of the earliest feature-based approaches. In his work, a specific configuration of 24 ultrasonic sensors was used to find the line segments, based on some detecting rules. A centerline model, with a simple ellipse shaped uncertain region, was used to represent a single sonar measurement. Line segment was created from three consecutive depth readings aligning within a tolerance and updated by a form of kalman filter. Wijk and Christensen developed a point feature extraction method, triangulation-based fusion (TBF) algorithm [8]. In the TBF algorithm, uniform distribution model was used to describe the sonar data. Point features were found at the intersections of the sonar arcs in a data window (moderate size: 10 steps and 16 sensors). The parameters were estimated by averaging the intersections on the present arcs. And local grid maps were used to refine the triangulation points. From the observed point features, line features were extracted by use of Hough transform method. The arc-transversal median (ATM) algorithm developed by Choset and Nagatani introduced the concept of transversal intersections [9]. The transversal intersections, considered as the stable intersections of two arcs by the authors were used to estimate the parameters of point features. The implementation of ATM is similar to that of TBF. Another different approach for feature-based map building is RCDs (regions of constant depth) method developed by Leonard [2], which considers both the line and point features. In their work, a rotating sonar scanner was used to obtain dense sonar data. At the first stage, based on the centerline model, RCDs were extracted from the sonar scans obtained at different robot locations. For each RCD, a series of angles to constrain the true bearing to the target were defined. Target was assumed to distribute uniformly on the arc between the constraint angles. Owing to this data-level fusion, angle uncertainty is reduced. Subsequently, classification of the clustered RCDs was determined by the percentage of RCDs in the cluster which matches one another according to each hypothesis, based on two RCDs matching model. At the final stage the parameters of the known clusters were estimated using orthogonal regressions for the line features or average for the point features.

D. Our Philosophies and Approaches

Our task is to determine the locations of the objects from sonar data. Without taking the measurement noises and environment complexity into account, we can determine the location of a line feature or a point feature from two measurements that both originate from a line or a point feature. The principles are: if the two measurements originate from a line feature, then determine the location by finding the common tangent line of two sonar arcs; if the two measurements originate from a point feature, the intersection of two sonar arcs is the location of the point feature. Unfortunately, the measurements don't provide direct information for associating them with specific type of features. A natural method to solve the problem is that: firstly clustering the measurements; then identifying the type of each clustering by use of the measurements in the clustering; finally estimating the locations of the identified features from the associated measurements. RCDs method is a case of this philosophy, unfortunately at the cost of stacking dense sonar data. Our concern is to find a method that can associate sparse sonar data, rather than dense sonar data, with a line feature or a point feature. The next section will describe our three measurements association model (TMAM) that can do so. Subsequently, a method to estimate the parameters of an identified feature from the associated measurements will be introduced in Section III. Finally, a feature-level fusion strategy for updating and maintaining the map will be presented in Section IV.

II. Data-level Fusion Model for Feature Identification

A. Two Measurements Association Model

When matching two sonar arcs, we use the similar definitions reported in [4]: contact point, which is the tangent point on each arc, where the common tangent line of these arcs pass; and intersection point, which is the intersection of two arcs. As shown in Fig. 1, r_1 and r_2 denote the two sonar readings at two sensor locations with a distance d. Angle ϕ_1 is the possible bearing to the object. The general problem is to find a third circle of known radius R, which is tangent to the two circles that the sonar arcs define. If R is known, we can compute ϕ_1 by

$$\cos(\phi_1) = \frac{(r_1+R)^2-(r_2+R)^2+d^2}{2d(r_1+R)} \tag{1}$$

Taking the limit of this equation as $R \to \infty$, it yields a result for line features:

$$\cos(\phi_1) = \frac{r_1-r_2}{d} \tag{2}$$

While setting $R = 0$, it yields

$$\cos(\phi_1) = \frac{r_1^2-r_2^2+d^2}{2dr_1} \tag{3}$$

It means that if we know the type of the feature, we can know where the readings originate. However our task is to obtain the knowledge of R and identify the type of features. Thus we write another form of (1) as

$$R = \frac{z_1^2-z_2^2-2d\cos(\phi_1)z_1}{2(d\cos(\phi_1)+z_2-z_1)} \tag{4}$$

We calculate all the possible R corresponding to all the possible ϕ_1 using (1). If the minimum of all the possible R is very large, line feature is determined. While if the maximum of all the possible R is very small, point feature is determined. Otherwise, arc feature is determined. This is the two measurements association model in [10]. This model seems a simple model to determine the type of a feature. However there are many occasions where not only the maximum of all the possible R is very large, but also the minimum of all the possible R is small. As a result, arc feature will be determined although there is no arc feature at all.

It is obvious that if there exist both contact points and intersection point on the two arcs, then the minimum of all the possible R is zero and the maximum is infinite. On this occasion two measurements association model will fail. It can't determine the type of the feature by use of two measurements. That is the problem using two measurements

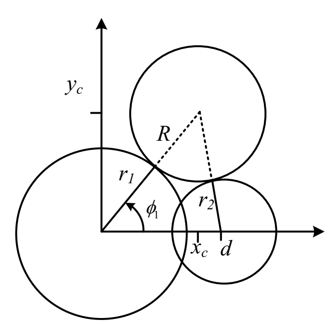

Fig. 1. Two measurements association model.

association model. For a general understanding, think of relations of two circles: the fact that two circles have common tangent lines doesn't deny the hypothesis that they have common intersections, and vice versa.

B. Three Measurements Association Model

In order to determine the type of a feature from sparse sonar data, we make a little stronger assumption that three measurements originate from a feature. A geometrical theorem is given by

Theorem 2.1: If three distinct circles with their different centers on a line have a common tangent line, then these circles don't have a common intersection, and vice versa. The exception is that these circles are commonly tangent at the intersection.[1]

Remark 2.1: Theorem 2.1 gives us a theoretical guidance to identify the type of a feature from three measurements. In the ideal case, if three sonar arcs originate from a common feature, the type of the feature can be determined by finding whether there exists a common tangent line or a common intersection. The assumptions that three sonar arcs originate from a common feature and their centers lie on a line are not too restricted. Three measurements obtained from a sonar sensor, when the robot moves in a straight line, always satisfy these assumptions.

In our views, consecutive measurements obtain from a sonar sensor have more chance to detect the same feature than those from different senors with different orientations, when the robot moves in a straight line. So at this data-level fusion stage, the fusions are implemented among the consecutive measurements obtained from a single sonar sensor, rather than among the measurements from different sensors. The information from different sensors will be fused at the feature-level fusion stage.

Theorem 2.1 is an ideal case. In practice, the measurements do originate from the planes and corners, rather than from the lines and points. Noises from the measurement devices and environments make it difficult to find a common tangent line or a common intersection of three sonar arcs. Fortunately, it is easy to find a common tangent line or a common intersection of two arcs.[2]

As shown in Fig. 2, when matching *arc*1 and *arc*2, we obtain two contact points and one intersection point; and when matching *arc*2 and *arc*3, we obtain two contact points and one intersection point too.

Theorem 2.2: If the two contact points on *arc*2 are identical, then *arc*1, *arc*2 and *arc*3 have a common tangent line.

Theorem 2.3: If the two intersection points on *arc*2 are identical, then *arc*1, *arc*2 and *arc*3 have a common intersection.[3]

[1]This theorem can be proofed by contradiction, and the details of the proof are omitted.

[2]The contact or intersection points of two circles can be obtained using (2) or (3). Thus by checking whether these points fall on the two arcs, we could determine whether the two arcs have a common tangent line or intersections.

[3]It is easy to prove these two theorems, so the proof is omitted.

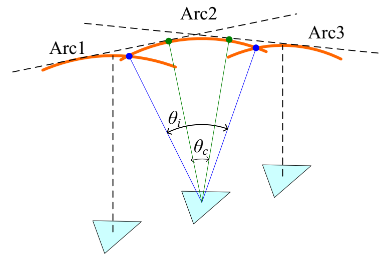

Fig. 2. Three measurement association model: the green and blue points denote the contact points and the intersection points respectively.

Remarks 2.2 and 2.3: These two theorems provide us with a different method to find a common tangent line or a common intersection of three arcs by checking the contact or intersection points on the *arc*2. Thus we can obtain following useful clues: More closely the two contact points distribute on arc2, more possibly these three arcs have a common tangent line; more closely the two intersection points distribute on arc2, more possibly these three arcs have a common intersection.

Then we use the angles θ_c and θ_i to measure the differences, as shown in Fig. 2. If there is only one contact point on arc2, we let $\theta_c = 22.5°$, and let $\theta_i = 22.5°$ when there is only one intersection point.

If $\theta_c < \alpha\theta_i$ and $\theta_c < \gamma < 22.5°$, we label *arc*2 as contact arc, where α has a value less than 1 and γ is a gating value less than 22.5 degree. If $\theta_c > \beta\theta_i$ and $\theta_i < \gamma < 22.5°$, we label *arc*2 as intersection arc, where β has a value larger than 1. Otherwise, we label *arc*2 unidentified arc.

Denote the consecutive readings as $\{r(k), k = 1,2,\ldots\}$ or $\{arc(k), k = 1,2,\ldots\}$ in an another form, where k is the time step, then we obtain the following rules:

- If $arc(k), arc(k+1), \ldots, arc(k+n)$ are contact arcs, then associate $r(k-1), r(k), r(k+1), \ldots, r(k+n+1)$ with a line feature. If $arc(k), arc(k+1), \ldots, arc(k+n)$ are intersection arcs, then associate $r(k-1), r(k), \ldots, r(k+n+1)$ with a point feature.
- If $arc(k), arc(k+1), \ldots, arc(k+n)$ are contact arcs and $arc(k+n+1)$ is a intersection or unidentified arc, then associate $r(k-1), r(k), r(k+1), \ldots, r(k+n+1)$ with a line feature.
- If $arc(k), arc(k+1), \ldots, arc(k+n)$ are intersection arcs and $arc(k+n+1)$ is a contact or unidentified arc, then associate $r(k-1), r(k), r(k+1), \ldots, r(k+n+1)$ with a point feature.

where $n = 0,1,\ldots$ is the number of consecutive labeled arcs associating with an identified feature. If $n = 0$, the number of the associated measurements is 3 and if $n > 0$, the number of the associated measurements is $n+2$.

III. Estimating Parameters of Identified Features

A. Two Observation Models

We denote a line as

$$p_r = x\cos(p_\theta) + y\sin(p_\theta) \quad (5)$$

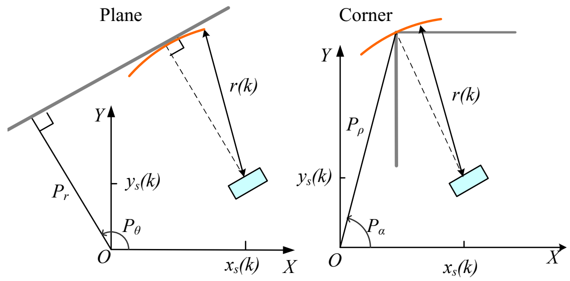

Fig. 3. Observation model of the line and point features.

where p_r is normal distance from the origin to the line and p_θ is the angle with respect to x axis, as shown in Fig. 3. Denote the location of the sensor as $p_s(k) = (x_s(k), y_s(k))$, and the measurement as $r(k)$ at step k. The observation model for line feature is given by

$$\begin{aligned} r(k) &= h_l(P_L, p_s(k)) + w(k) \\ &= |p_r - x_s(k)\cos(p_\theta) - y_s(k)\sin(p_\theta)| + w(k) \end{aligned} \tag{6}$$

where $P_L = (p_r, p_\theta)$ is the parametric vector of the line, and $w(k)$ is the zero-mean white Gaussian noises with variance σ^2.

The observation model for point feature is written as

$$\begin{aligned} r(k) &= h_p(P_C, p_s(k)) + w(k) \\ &= \sqrt{(p_\rho\cos(p_\alpha) - x_s(k))^2 + (p_\rho\sin(p_\alpha) - y_s(k))^2} + w(k) \end{aligned} \tag{7}$$

where $P_C = (p_\rho, p_\alpha)$ is parametric vector; $w(k)$ is the noises and (p_ρ, p_α) are the polar coordinates of the point feature in the world coordinate system.

B. Iterated Least Square Estimation

The Iterated Least Square (ILS) Estimator is a technique for iteratively improving the current estimate using the measurements until convergence (or up to a certain maximum number of iterations) based on the Least Square principle [11]. It is an approximate one by use of a first order series expansion to solve the nonlinear estimation. Let T denote transpose, then we write the consecutive measurements obtained from an identified feature as

$$\mathbf{r} = [r(k), r(k+1), ..., r(k+n)]^T = \mathbf{h}(P, p_s) + w \tag{8}$$

where

$$\mathbf{h}(P, p_s) = \begin{pmatrix} h(P, p_s(k)) \\ h(P, p_s(k+1)) \\ \vdots \\ h(P, p_s(k+n)) \end{pmatrix} \tag{9}$$

are the observation equations and P is the parametric vector. Given the estimate $\hat{P}_j$ at the end of iteration j, the updated $\hat{P}_{j+1}$ is obtained as

$$\hat{P}_{j+1} = \hat{P}_j + (J_j^T R^{-1} J_j)^{-1} J_j^T R^{-1}[\mathbf{r} - \mathbf{h}(\hat{P}_j, p_s)] \tag{10}$$

where

$$J_j = \left.\frac{\partial \mathbf{h}(P, p_s)}{\partial P}\right|_{P=\hat{P}_j} \tag{11}$$

is the Jacobian matrix and R is the measurement covariance matrix give by $diag(\sigma^2, \sigma^2, ...)$. An initial estimate $\hat{p}_0$ for the estimator can be obtained from the intersection of any two sonar arcs for a point feature, or from the common tangent line of any two sonar arcs for a line feature. The mean square error matrix of the final estimate $\hat{p}$ at iteration $j+1$ is obtained by

$$E[(\hat{P}_{j+1} - P)(\hat{P}_{j+1} - P)^T] = (J^T R^{-1} J)^{-1} \tag{12}$$

where J is the Jacobian, evaluated at the final estimate. The Jacobian matrix J_j for a point feature is given by

$$J_j = \begin{bmatrix} \frac{\partial h(P, p_s(k))}{\partial p_\rho} & \frac{\partial h(P, p_s(k))}{\partial p_\alpha} \\ \frac{\partial h(P, p_s(k+1))}{\partial p_\rho} & \frac{\partial h(P, p_s(k+1))}{\partial p_\alpha} \\ \vdots & \vdots \\ \frac{\partial h(P, p_s(k+n))}{\partial p_\rho} & \frac{\partial h(P, p_s(k+n))}{\partial p_\alpha} \end{bmatrix}_{P=\hat{P}_j} \tag{13}$$

with

$$\frac{\partial h(k)}{\partial p_\rho} = \frac{p_\rho - x_s(k)\cos(p_\alpha) - y_s(k)\sin(p_\alpha)}{\sqrt{(p_\rho\cos(p_\alpha) - x_s(k))^2 + (p_\rho\sin(p_\alpha) - y_s(k))^2}} \tag{14}$$

$$\frac{\partial h(k)}{\partial p_\alpha} = \frac{p_\rho(x_s(k)\sin(p_\alpha) - y_s(k)\cos(p_\alpha))}{\sqrt{(p_\rho\cos(p_\alpha) - x_s(k))^2 + (p_\rho\sin(p_\alpha) - y_s(k))^2}} \tag{15}$$

where $h(k) = h(P, p_s(k))$.

However, because of presence of an absolute symbol in (6), it is difficult to induce directly the Jacobian matrix from the observation model of a line feature. In order to get rid of the absolute symbol, we define a local coordinate system with the origin at the location where the sensor first observes the line. The orientation of the sensor is defined as the y axis of the local coordinate system, as shown in Fig. 4. Thus the equation of a line in the local coordinate system is written as

$$p_r^S - x^S\cos(p_\theta^S) - y^S\sin(p_\theta^S) = 0 \tag{16}$$

where superscript S denotes the representation in the local coordinate system. As the width of sonar arc is 22.5°, it yields

$$\sin(p_\theta^S) > 0 \tag{17}$$

for all possible lines tangent to the arc. Additionally, the consecutive locations of the senors are all below any possible line. Thus we obtain following inequations:

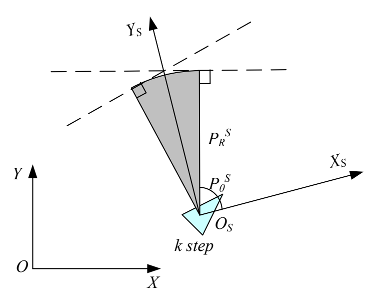

Fig. 4. Observation model of line feature in local coordinate system.

$$p_r^S - x_s^S(i)\cos(p_\theta^S) - y_s^S(i)\sin(p_\theta^S) > 0 \quad (18)$$

where $i = k, k+1, \ldots, k+n$. Using (6) and (18), the observation model in the local coordinate system is given by

$$h(P^S, p_s^S(k)) = p_r^S - x_s^S(k)\cos(p_\theta^S) - y_s^S(k)\sin(p_\theta^S) + w(k) \quad (19)$$

which immediately yields

$$\frac{\partial h(P^S, p_s^S(k))}{\partial p_r^S} = 1 \quad (20)$$

$$\frac{\partial h(P^S, p_s^S(k))}{\partial p_\theta^S} = \sin(p_\theta^S)x_s^S(k) - \cos(p_\theta^S)y_s^S(k) \quad (21)$$

At the end of iterations, we obtain the estimate $\hat{P}_C = (\hat{p}_\rho, \hat{p}_\alpha)$ or $\hat{P}_L^S = (\hat{p}_r^S, \hat{p}_\theta^S)$ with the respective mean square error matrix

$$\Lambda_C = \begin{bmatrix} \sigma_{\rho\rho}^2 & \sigma_{\rho\alpha} \\ \sigma_{\alpha\rho} & \sigma_{\alpha\alpha}^2 \end{bmatrix} \quad (22)$$

or

$$\Lambda_L^S = \begin{bmatrix} \sigma_{rr}^2 & \sigma_{r\theta} \\ \sigma_{\theta r} & \sigma_{\theta\theta}^2 \end{bmatrix} \quad (23)$$

We describe an extracted point feature using a parametric vector $p_f = (\hat{p}_\rho, \hat{p}_\alpha, \Lambda_C, n_c)$, where n_c is the number of the associated measurements. As for a line segment, we use $\hat{P}_L = (\hat{p}_r, \hat{p}_\theta, \Lambda_L, x_m, y_m, h, n_l)$ to describe it in the world coordinate system:

- $\hat{p}_r$: the estimated perpendicular distance from the line segment to the origin of the world coordinate system.
- $\hat{p}_\theta$: the estimated orientation of line segment.
- Λ_L: the mean square error matrix transformed from the local coordinate system to the world coordinate system.
- (x_m, y_m): coordinates of the mid-point of the line segment.
- h: half length of the line segment.
- n_l: number of the associated measurements.

IV. Feature-level Fusions for Map Updating

The first stage for map updating is matching the extracted feature with the registered features. Successful matching is used to update the map. If there is no matching, the extracted feature will be registered as a new one in the map. If more than one matching happens, we should use a reasonable measure to decide which matching is the proper one. As for point features, we use Euclidian distance to determine whether the extracted feature matches with the registered one. If the distance is smaller than a gating value, a matching is found. The matching that has the smallest distance is used to update the map. If no match happens, the extracted point feature will be registered in the map.

In our philosophies, the measurements are considered as the samples taken from the corresponding feature in the environment. The parameters of both the extracted feature $p_{fe} = (\hat{p}_{\rho e}, \hat{p}_{\alpha e}, \Lambda_{Ce}, n_e)$ and the registered features $p_{fr} = (\hat{p}_{\rho r}, \hat{p}_{\alpha r}, \Lambda_{Cr}, n_r)$ are considered as the statistical results from different sets of samples. Then the total statistical results are naturally obtained as

$$\hat{p}_{\rho u} = \frac{n_e\hat{p}_{\rho e} + n_r\hat{p}_{\rho r}}{n_e + n_r}$$
$$\hat{p}_{\alpha u} = \frac{n_e\hat{p}_{\alpha e} + n_r\hat{p}_{\alpha r}}{n_e + n_r}$$
$$\Lambda_{Cu} = \frac{n_e\Lambda_{Ce} + n_r\Lambda_{Cr}}{n_e + n_r}$$
$$n_u = n_e + n_r$$

As for line feature, matching is a process of comparing an extracted line feature to each of registered line features in the map to detect similarity in orientation, co-linearity and overlap. Denote the extracted and registered features as $p_{fe} = (\hat{p}_{re}, \hat{p}_{\theta e}, \Lambda_e, x_{me}, y_{me}, h_e, n_e)$ and $p_{fr} = (\hat{p}_{rr}, \hat{p}_{\theta r}, \Lambda_r, x_{mr}, y_{mr}, h_r, n_r)$ respectively. The matching process are described by following steps:

- Check the similarity in orientation, co-linearity and overlap by

$$(\hat{p}_{\theta e} - \hat{p}_{\theta r})^2 < \sigma_{\theta\theta e}^2 + \sigma_{\theta\theta r}^2 \quad (24)$$

$$(\hat{p}_{re} - \hat{p}_{rr}) < \sigma_{rre}^2 + \sigma_{rrr}^2 \quad (25)$$

$$(x_{me} - x_{mr})^2 + (y_{me} - y_{mr})^2 < (h_e + h_r)^2 \quad (26)$$

- Calculate the deviation between the extracted and the registered features which pass above three tests by

$$Deviation = \frac{(\hat{p}_{\theta e} - \hat{p}_{\theta r})^2}{\sigma_{\theta\theta e}^2 + \sigma_{\theta\theta r}^2} + \frac{(\hat{p}_{re} - \hat{p}_{rr})}{\sigma_{rre}^2 + \sigma_{rrr}^2} + \frac{(x_{me} - x_{mr})^2 + (y_{me} - y_{mr})^2}{(h_e + h_r)^2} \quad (27)$$

- Select the registered feature with the smallest deviation as successful matching. Thus the parameters of the updated line feature are obtained as

$$\hat{p}_{ru} = \frac{n_e\hat{p}_{re} + n_r\hat{p}_{rr}}{n_e + n_r}$$
$$\hat{p}_{\theta u} = \frac{n_e\hat{p}_{\theta e} + n_r\hat{p}_{\theta r}}{n_e + n_r}$$
$$\Lambda_u = \frac{n_e\Lambda_e + n_r\Lambda_r}{n_e + n_r}$$
$$n_u = n_e + n_r$$

And the additional parameters (x_{mu}, y_{mu}) and h_u are obtained by projecting the extracted and registered segments on the updated line.

V. Experimental Results

In order to test the feasibility of our approach, we test it both in simulation and on real data. Fig. 5 shows the experimental environment composed of planes, corners and edges. The robot, with a ring of 16 Polaroid ultrasonic sensors, runs following the dashed line. In our implementation, the distances between consecutive sampling steps are measured by odometry.

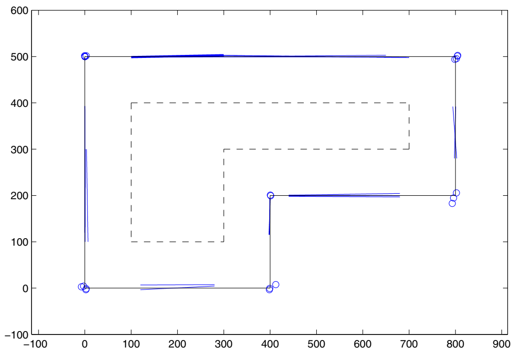

Fig. 5. Experimental results after applying TMAM.

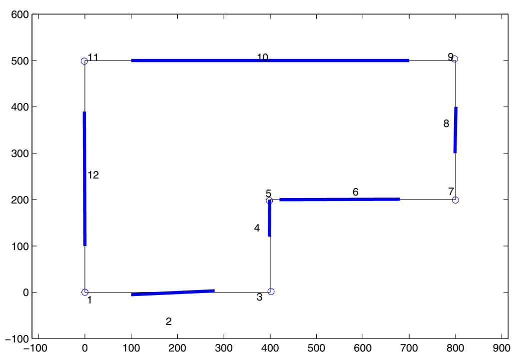

Fig. 6. Experimental results of final feature map.

After the robot makes a complete turn in the environment, 2400 sonar data for 150 steps are obtained. By use of TMAM, 524 sonar data are associated with 20 point features and 17 line features respectively as shown in Fig. 5. The blue line denotes the extracted line feature, and the blue circle denotes the position of the extracted point feature. The parameters of these features are estimated by use of the iterated least square method. In our implementation, the ILS iterations terminate when the norm of the difference between two consecutive estimates is less than 1 or when the number of iterations exceeds 10. After the feature-level fusions, 6 point features and 6 line features are registered in the map, as shown in Fig. 6. The estimated and actual parameters of these features are listed in Table I.

VI. Conclusions

We have proposed an approach for building a feature-based map using sonar data. In the data-level fusion, a model has been developed to associate sparse sonar data with specific type of features. The parameters of an identified features are estimated by use of ILS methods from the associated measurements. In the feature-level fusion, information from the sonar senors mounted at different locations on the robot is fused to update and maintain the map. Our approach can be implemented in real-time, because of using sparse data to extract the features.

Experimental results show that the proposed approach has the capability to robustly extract the features and build a satisfactory feature-based map. In future work, we will exert

TABLE I

Experimental Results

Point Feature	Estimated		Actual	
	$\hat{p}_\rho$ (cm)	$\hat{p}_\alpha$	p_ρ (cm)	p_α
1	4.5357	−0.8192	0	0
3	405.22	0.004	400	0
5	444.4	0.4605	447.2136	0.46
7	824.13	0.2442	824.6211	0.245
9	943.11	0.5628	943.3981	0.5586
11	506.29	1.5735	500	1.5708
Line Feature	**Estimated**		**Actual**	
	$\hat{p}_r$ (cm)	$\hat{p}_\theta$	p_r (cm)	p_θ
2	0.5839	1.5686	0	1.5708
4	402.73	0.0161	400	0
6	197.98	1.5754	200	1.5708
8	807.12	0.0222	800	0
10	500.45	1.5702	500	1.5708
12	2.0828	0.006	0	0

on finding the potential performance of TMAM and more proper estimation methods. Furthermore, we will take the pose uncertainty of robot into account and focus on the SLAM problem.

VII. Acknowledgements

This research has been supported in part by the National Natural Science Foundation of China (Grant Nos. 60205004, 50475179, 60528002, 60621001 and 60635010), the Hi-Tech R&D Program (863) of China (Grant No. 2006AA04Z258).

References

[1] J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," IEEE Trans. on Robotics and Automation, vol. 7, no. 3, pp. 376-382, June 1991.
[2] R. Kuc and M. W. Siegel, "Physically based simulation model for acoustic sensor robot navigation," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 9, No. 6, pp. 766-778, 1987.
[3] B. Barshan and R. Kuc, "Differentiating sonar reflections from corners and planes by employing an intelligent sensor," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 12, no. 6, pp. 560-569, June 1990.
[4] J. J. Leonard, "Direct sonar sensing for mobile robot navigation," Kluwer Academic Publishers, 1992.
[5] R. Chatila and J. P. Laumond, "Position referencing and consistent world modeling for mobile robots," IEEE Intl. conf. on Robotics and Automation, pp. 138-145, 1985.
[6] A. Elfes, "Sonar-Based Real-World Mapping and Navigation," IEEE Trans. on Robotics and Automation, vol. 3, No. 3, June 1987.
[7] J. L. Crowly, "Navigation for an intelligent mobile robot," IEEE Jour. of Robotics and Automation, RA-1(1), pp. 31-41, 1985.
[8] O. Wijk, H. I. Christensen, "Triangulation-based fusion of sonar data with application in robot pose tracking," IEEE Trans. On Robotics and Automation, vol. 16, no. 6, pp. 740-752, 2000.
[9] H. Choset, K. Nagatani and N. A. Lazar, "The Arc-Transversal Median Algorithm: A Geometric Approach to Increasing Ultrasonic Sensor Azimuth Accuracy," IEEE Trans. on Robotics and Automation, vol. 19, pp. 513-522, 2003.
[10] S. J. Lee, J. H. Lim, D. W. Cho, C. U. Kang, and W. K. Chung, "Feature Based Map Building Using Sparse Sonar Data," IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems, pp. 492-496, 2005.
[11] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, Estimation with Applications to Tracking and Navigation, John Wiley and Sons, 2001.