IEEE/RSJ International Workshop on Intelligent
Robots and Systems IROS '91, Nov. 3-5, 1991,
Osaka, Japan. IEEE Cat. No. 91TH0375-6

Simultaneous Map Building and Localization for an Autonomous Mobile Robot

John J. Leonard*
NEC Research Institute
4 Independence Way
Princeton, NJ 08540
U.S.A.

Hugh F. Durrant-Whyte
Department of Engineering Science
University of Oxford
Parks Road, Oxford OX1 3PJ
ENGLAND

Abstract

In this paper, we discuss a significant open problem in mobile robotics: simultaneous map building and localization, which we define as long-term globally referenced position estimation without *a priori* information. This problem is difficult because of the following paradox: to move precisely, a mobile robot must have an accurate environment map; however, to build an accurate map, the mobile robot's sensing locations must be known precisely. In this way, simultaneous map building and localization can be seen to present a question of "which came first, the chicken or the egg?" (The map or the motion?) When using ultrasonic sensing, to overcome this issue we equip the vehicle with multiple servo-mounted sonar sensors, to provide a means in which a subset of environment features can be precisely learned from the robot's initial location and subsequently tracked to provide precise positioning.

1 Introduction

We consider mobile robot navigation to be a problem of *tracking* geometric features (targets) which are present in the environment. Building a map requires the interpretation of sensor information to estimate the locations of geometric features in a global reference frame. Localization is the task of using geometric feature location estimates contained in the map to determine the robot's position. In our previous research, we have investigated each of these problems on their own. The present discussion is aimed at combining these two capabilities.

Our initial research in sonar-based navigation concentrated on position estimation given an accurate environment model. We developed an extended Kalman filter (EKF) based localization algorithm to provide fast vehicle position updates by matching individual sonar returns from a ring of sensors to an *a priori* map of "geometric beacons" (planes, corners, and cylinders)[9][7]. In more recent experimental testing, "stop, look, and move" navigation around a sequence of pre-specified goal locations without human intervention was achieved in two different laboratories, with a typical time to failure of about one hour and some runs longer than three hours[10].

While localization is a top-down, expectation-driven process of matching current observations with *predictions* from the geometric model, map building requires the more difficult bottom-up task of generating *explanations* for observed sensor data. Using time-of-flight information alone, the surface geometry (plane, corner, wall, multiple reflection) from which a single sonar return originated cannot be determined[6]. We have developed an algorithm to differentiate planes and corners that uses a circle intersection/circle tangency test to group sonar returns of common origin and compute a composite line or point state estimate[7]. Using eighteen sonar scans taken from known sensing locations with the standard Polaroid ranging system[16], an uncluttered office scene was mapped with sub-centimeter accuracy.[1] More recently, we have formulated this bottom-up sonar interpretation task more rigorously using probabilistic data association[3].

Unfortunately, putting these two capabilities together is not as straightforward as one might hope. The major reason for this is the "the correlation problem": if a mobile robot uses an observation of an imprecisely known target to update its position, the resulting vehicle position estimate becomes correlated with the feature location estimate. Likewise, correlations are introduced if an observation taken from an imprecisely known position is used to update the location estimate of a geometric feature in the map. A rigorous solution to simultaneous map building and localization must explicitly represent all the correlations between the estimated vehicle and geometric feature locations.

One solution was provided by Smith, Self and Cheeseman, who developed algorithms for building a "stochastic map" of spatial relationships, using the EKF. Moutarlier and Chatila have presented a framework similar to the stochastic map, but with the added feature that colored and correlated noise is accommodated, and have implemented their approach in two dimensions using laser range data[14].

Our engineering instincts tell us that the stochastic map would be tremendously difficult to implement in practice. One issue is computational complexity: in a two-dimensional environment containing n geometric features, the stochastic map requires a system state vector of size $2n+3$. Because the EKF is $\mathcal{O}(n^3)$ [12], the computational burden of the approach would be substantial in typical application environments with hundreds of map features.

Two additional considerations not addressed in this paper are *data association uncertainty* and *environment dynamics.* We define data association to be the process determining the origins of sensor observations, that is, associating measurements with the geometric features that produced them while at the same time rejecting spurious measurements[3]. Because both Smith, Self and Cheeseman and Moutarlier and Chatila implicitly assume perfect data association, spurious measurements and incorrect matches or groupings would have an undesirable impact on the stability

*Current address: Sea Grant College Program, Bldg. E38, Massachusetts Institute of Technology, Cambridge, MA, 02139

[1] In Section 3 we show the output of the same algorithm using just four sonar scans.

of the stochastic map representation. Dynamic environments require an inference procedure in which missed detections provide information pertaining to state changes in the map, such as the disappearance of previously learned geometric features[8].

The next section defines the problem under consideration in a multitarget tracking framework. In Section 3, we offer some thoughts on how a sensing strategy can be used to avoid the correlation problem. Section 4 presents experimental results to illustrate how the "getting started" paradox can be avoided with sonar.

2 Problem Statement

We assume that the actual three dimensional environment geometry is orthogonal to the horizontal plane in which the robot moves, so that the world can be adequately represented by a two dimensional model.

2.1 The System Model

We denote the position and orientation of the robot at time step k by the vector $\mathbf{x}_R(k) = [x(k), y(k), \theta(k)]^T$ comprising a Cartesian location and a heading defined with respect to a global coordinate frame. The vehicle's motion through the environment is described by a plant model of the form

$$\mathbf{x}_R(k+1) = \mathbf{f}_R(\mathbf{x}_R(k), \mathbf{u}(k)) + \mathbf{v}(k) \tag{1}$$

where $\mathbf{u}(k)$ is a control input, $\mathbf{v}(k)$ is a noise disturbance, and $\mathbf{f}_R(\mathbf{x}_R(k), \mathbf{u}(k))$ is the (non-linear) state transition function determined by the kinematics of the vehicle[15].

We model the environment in two dimensions with three types of target: points, lines and arcs. The point target classification encompasses both corners (concave dihedrals) and edges (convex dihedrals[6]). Lines represent 3-D planes and arcs represent 3-D cylinders. The location of feature i at time k is specified by the parameter vector $\mathbf{x}_i(k)$. The form of the parameter vector depends on the type of the feature:

1. A line is represented by the vector $\mathbf{x}_i(k) = [R_i(k), \theta_i(k)]^T$ where $R_i(k)$ is the perpendicular distance from the origin of the global coordinate frame to the line and $\theta_i(k)$ is the orientation of a perpendicular drawn from the origin to the line.
2. A point is represented by the vector $\mathbf{x}_i(k) = [x_i(k), y_i(k)]^T$ where $x_i(k)$ and $y_i(k)$ specify the two dimensional location of the point.
3. An arc is represented by the vector $\mathbf{x}_i(k) = [x_i(k), y_i(k), R_i(k)]^T$ where $x_i(k)$ and $y_i(k)$ specify the center of the circle and $R_i(k)$ specifies its radius.

In addition, each line and arc target has two endpoints associated with it, to facilitate visibility prediction. For map building, these parameters are estimated outside the EKF framework by projecting new observations onto the circle or infinite line defined by the feature state vector.

The state of the environment is specified by the set of geometric feature locations (the true map):

$$M = \{\mathbf{x}_i | 1 \le i \le n\} \tag{2}$$

where n is the number of geometric features in the environment. In the general setting of a dynamic environment, the true map is time-varying as feature locations can change with time according to (unknown and possibly different) feature plant models, but, as stated previously, for our purposes here we assume a static environment in which feature locations do not change with time.

Together, the location of the vehicle and the state of the environment comprise the *system state vector* $\mathbf{x}(k)$:

$$\mathbf{x}(k) = [\mathbf{x}_R(k), \mathbf{x}_1, \ldots, \mathbf{x}_n]^T \tag{3}$$

2.2 The Measurement Model

At time step k, the robot obtains $m(k)$ sonar measurements, which form the current *data set* $Z(k)$

$$Z(k) = \{\mathbf{z}_j(k) | 1 \le j \le m(k)\} \tag{4}$$

We define the cumulative data set Z^k as the set of data sets up through time k:

$$Z^k \triangleq \{Z(j) | 0 \le j \le k\} \tag{5}$$

We introduce the notation $\mathbf{z}_j(k) \leftarrow \mathbf{x}_i(k)$ to indicate that feature $\mathbf{x}_i(k)$ **generates** measurement $\mathbf{z}_j(k)$ at time k. Each measurement $\mathbf{z}_j(k)$ in a data set is assumed to be generated by (*i.e.* originate from) a single geometric feature in the environment or to be a false alarm (in which case we write $\mathbf{z}_j(k) \leftarrow \emptyset$). The value of a measurement $\mathbf{z}_j(k)$ is a function of the vehicle location at time k and the location of the feature from which it originated, subject to a noise disturbance, as given by the measurement model

$$\mathbf{z}_j(k) = \mathbf{h}_i(\mathbf{x}_R(k), \mathbf{x}_i(k)) + \mathbf{w}_j(k) \tag{6}$$

where the measurement function $\mathbf{h}_i(\cdot, \cdot)$ takes a different form depending on the type of feature i (wall, corner, or cylinder.) Our sonar sensor model follows in part from [6] and is described in detail in [7].

2.3 State Estimation

Within this framework, the general solution requires the computation of the number of features present, the type of each feature (wall, corner, cylinder), and $p(\mathbf{x}(k)|Z^k)$, the *a posteriori* probability distribution of vehicle and feature states conditioned on the cumulative measurement set Z^k [13]. Our objective is to use the EKF to recursively compute an MMSE estimate for $\mathbf{x}(k)$:

$$\hat{\mathbf{x}}(k \mid k) = [\hat{\mathbf{x}}_R(k \mid k), \hat{\mathbf{x}}_1(k), \ldots, \hat{\mathbf{x}}_n(k)]^T \tag{7}$$

which is designated the *system state estimate*, and its covariance $\Lambda(k \mid k)$ (the system covariance matrix):

$$\Lambda(k \mid k) = \begin{bmatrix} \mathbf{P}(k \mid k) & \mathbf{C}_{R1}(k \mid k) & \cdots & \mathbf{C}_{Rn}(k \mid k) \\ \mathbf{C}_{R1}(k \mid k) & \Lambda_1(k) & \cdots & \mathbf{C}_{1n}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{Rn}(k \mid k) & \mathbf{C}_{1n}(k) & \cdots & \Lambda_n(k) \end{bmatrix} \tag{8}$$

where $\mathbf{C}_{Ri}(k \mid k)$ is a vehicle to feature cross-covariance matrix and $\mathbf{C}_{ij}(k)$ is a feature to feature cross-covariance matrix. In addition to the strong (and often justifiably criticized[4]) assumptions that the Kalman filter requires, such as Gaussianity and independence of noise disturbances, it is assumed that data association can be resolved using the Mahalanobis distance and a "nearest neighbor" assignment strategy, with heuristics to initialize new targets and eliminate outliers.

While space prevents a detailed presentation of either [17] or [14], we briefly summarize each method here. Smith, Self and

Cheeseman advocate a two stage process of

1. vehicle position prediction
   compute: $\hat{\mathbf{x}}_R(k+1 \mid k)$ and $\{\mathbf{C}_{Ri}(k+1 \mid k) | 1 \le i \le n\}$
   given: $\mathbf{u}(k)$ and $\hat{\mathbf{x}}_R(k \mid k)$
2. system state update
   compute: $\hat{\mathbf{x}}(k+1 \mid k+1)$ and $\Lambda(k+1 \mid k+1)$
   given: Z^k, $\hat{\mathbf{x}}_R(k+1 \mid k), \{\mathbf{C}_{Ri}(k+1 \mid k) | 1 \le i \le n\}$
   and $\{\mathbf{C}_{ij}(k) | 1 \le i < j \le n\}$

Subject to the assumptions mentioned earlier, this provides the optimal linear estimate for the system state at time k. However, Moutarlier and Chatila have observed that in practice, this approach is subject to divergence, principally because of biases introduced by linearization.[2] To alleviate this effect, they propose a suboptimal three stage procedure, called the *relocation-fusion* approach, in which the system update step above is replaced by:

2 (a). vehicle position update (relocation)
compute: $\hat{\mathbf{x}}_R(k+1 \mid k+1)$ and $\{\mathbf{C}_{Ri}(k+1 \mid k+1) | 1 \le i \le n\}$
given: Z^k, $\hat{\mathbf{x}}_R(k+1 \mid k)$ and $\{\mathbf{C}_{Ri}(k+1 \mid k) | 1 \le i \le n\}$

2 (b). map update (fusion)
compute: $\{\hat{\mathbf{x}}_i(k+1) | 1 \le i \le n\}$ and $\{\mathbf{C}_{ij}(k+1) | 1 \le i < j \le n\}$
given: Z^k, $\hat{\mathbf{x}}_R(k+1 \mid k+1), \{\mathbf{C}_{Ri}(k+1 \mid k+1) | 1 \le i \le n\}$ and $\{\mathbf{C}_{ij}(k) | 1 \le i < j \le n\}$

By updating the vehicle state and then re-linearizing before attempting any feature state updates, stability is enhanced[14].

We note that in either approach, $\Lambda(k \mid k)$ is a $2n+3$ by $2n+3$ matrix whose cross-covariance sub-matrices $\mathbf{C}_{Ri}(k \mid k)$ and $\mathbf{C}_{ij}(k)$ are non-zero for all i and j. Because the vehicle and feature estimates are correlated, then each time the vehicle position is updated, each vehicle-feature covariance matrix $\mathbf{C}_{Ri}(k \mid k)$ must be recomputed. Similarly, each time a feature state estimate is updated, one must update all cross-covariance matrices involving that feature.

2.4 Motion Estimation *vs* Position Estimation

A closely related problem which has received widespread attention in the robotics and computer vision communities is usually referred to as "motion and structure from motion[1][11][5]", and can be stated as: estimate the displacement $\mathbf{d}_k$ (comprised of a rotation $\mathbf{R}$ and a translation $\mathbf{t}$) between two sensing locations $\mathbf{x}_R(k-1)$ and $\mathbf{x}_R(k)$, and position estimates for features visible at time k in the *relative* coordinate defined by the vehicle's current position. By compounding successive motion estimates starting at time $k=0$, a globally referenced estimate of vehicle location can be computed[17]:

$$\hat{\mathbf{x}}_R(k \mid k) = \hat{\mathbf{x}}_R(0 \mid 0) \oplus \hat{\mathbf{d}}_1 \oplus \hat{\mathbf{d}}_2 \oplus \ldots \oplus \hat{\mathbf{d}}_k \tag{9}$$

Our objection to this formulation rests with the covariance $\mathbf{P}(k \mid k)$ that accompanies this global position estimate: it increases without bound, *i.e.*

$$\lim_{k \to \infty} \operatorname{tr}(\mathbf{P}(k \mid k)) = \infty \tag{10}$$

The robot is gradually getting lost, albeit perhaps quite slowly. No globally referenced geometry is computed to enable the robot to determine its position when it travels through the same part of the world an hour or a week later. One difficult question that can be asked is "could the robot return to its starting position, within a desired tolerance, such as 1 cm and 1 degree, after traveling a considerable distance?" To do this, one needs to compute globally referenced geometry, which forces one to confront the correlation problem.

[2] In our own sonar-based experiments with the direct, single stage approach of [17], we have also observed this divergent behavior.

3 Sensing Strategy For Decoupling $\Lambda(k \mid k)$

Qualitatively, the utility of an observation from a map building perspective depends on the accuracy with which the vehicle position from which the observation was taken is known. Conversely, the utility of an observation for position estimation depends on the accuracy with which one knows the location of the feature being observed. Our proposal is to take this view to its extreme: measurements from inaccurately known positions should not be used to update the location estimates of any geometric features; rather, they should be simply thrown away.

We propose the following strategy: the robot can choose where it *goes* and where it *looks* to incrementally build the map in a fashion such that 1) correlations are eliminated and 2) position in the global frame can always be estimated to within a desired tolerance, such as 1 cm and 1 degree. The robot must have the ability to precisely determine its position, and to do so it must precisely know the locations of some geometric features.[3]

To evaluate the accuracy of a state estimate, we shall compare the trace of its covariance matrix $\mathbf{P}(k \mid k)$ or $\Lambda_i(k)$ with a tolerance parameter ϵ_R or ϵ_F, as appropriate. The values of ϵ_R and ϵ_F reflect when a covariance matrix can be safely approximated as zero, and can be chosen experimentally. Feature estimates that meet the criterion $\operatorname{tr}(\Lambda_i(k)) < \epsilon_F$ will be designated *confirmed* targets. Likewise vehicle position estimates that satisfy $\operatorname{tr}(\Lambda_i(k)) < \epsilon_R$ will be designated confirmed vehicle locations.

The proposed sensing strategy can be stated as follows:

```
if z_j(k) ← x_i(k):
    if tr(Λ_i(k)) < ε_F update x̂_R(k | k) with z_j(k)
    else tr(P(k | k)) < ε_R update x̂_i(k) with z_j(k)
    else ignore observation z_j(k)
```

First, before moving, the robot needs to accurately learn the locations of an initial confirmed feature set. Subsequently, after each move, the robot takes observations of the initial feature set to accurately determine its position, then takes observations of new features to estimate their locations. Additional observations of these new features from subsequent accurately known sensing locations will result in sufficiently accurate estimates that these features can in turn be promoted to confirmed target status, and used for vehicle localization.

4 Initialization With Sonar

The objective of the sensing strategy of the previous section is to enforce a banded structure upon the system covariance matrix $\Lambda(k \mid k)$. To get the process started, some targets must be accurately learned before any vehicle motion takes place. However, walls, corners, cylinders and multiple reflections cannot be identified using a single sonar sensor and time-of-flight information, as discussed previously. Hence, we advocate the use of multiple servo-mounted sensors on the vehicle. One approach would be a *four corners* sensing configuration, in which servo-mounted sen-

[3] This does not mean that the robot must precisely know its position *all the time*, only that it must do so before updating any feature state estimates in its map.

sors are placed at each corner of the vehicle. This configuration provides a way out of the getting started paradox. The portion of the world visible from the initial vehicle location can be precisely learned *without* moving the vehicle, because of the known baseline between sensors. To provide evidence for this assertion, Figure 1 shows a map building run, with the algorithm of [7][8], using the data of just four sonar scans. Tables 1 and 2 show a comparison of estimated and hand-measured feature locations, showing accuracy on the order of 1.0 cm, from just four sonar scans.

In this way, at start-up an initial confirmed target set can be learned without vehicle motion. These confirmed targets can then be tracked as the vehicle moves to provide accurate localization, while other sensors can be used to map out new targets. We claim that this approach can sidestep the correlation problem; the use of individual vehicle and target covariances can suffice, without explicit representation of cross-correlations. Observations of confirmed targets update vehicle position but not target state; observations of tentative targets update target state but not vehicle position.

5 Discussion

To achieve genuine long-term autonomy, it is not enough just to *represent* uncertainty; we need to *reduce* uncertainty. The sensing strategy presented here aims to eliminate the vehicle position uncertainty associated with a new set of measurements before using those measurements to update the map, thereby eliminating cross-correlations between map features. This decouples the system covariance matrix, easing the state estimation computational burden and simplifying the data association procedure.

The global frame is anchored by an initial confirmed target set which is learned before any vehicle motion takes place, as illustrated by Figure 1. Because walls, corners, cylinders and multiple reflections are indistinguishable in a single sonar scan, we require that multiple servo-mounted sonar sensors be mounted on the vehicle, *e.g.* in a four corners sensing configuration.

While we believe that Smith, Self and Cheeseman[17] and Moutarlier and Chatila[14] have made very important contributions in this area, we feel the general problem remains open, for three reasons: data association uncertainty, environment dynamics, and the impact of correlations on computational complexity. In ongoing research with Ingemar Cox[3], we are investigating the use of multiple hypothesis probabilistic data association techniques to address the former two problems. To address the correlation problem, we have sketched a sensing strategy that may permit one to sidestep the issue. However, we cannot at this stage verify the approach with experimental results. Experimentation is required to establish what level of correlations can be safely ignored, and at what stage significant performance degradation (*i.e.* EKF divergence) occurs.

One might argue that we overestimate these difficulties. Despite their large size, the covariance matrices of the stochastic map may be manageable in practice. Moutarlier and Chatila report that "The complexity of the computations are such that computing time is satisfactory (a few seconds for each step in the experiments shown ....)[14]." In some situations, data association ambiguity may be infrequent, as in Moutarlier and Chatila's results, where the use of the Mahalanobis distance suffices. However, if this is not the case, an optimal algorithm that considers all possible data association hypotheses and represents all cross-correlations would require *exponentially many* $2n + 3$ by $2n + 3$ system covariance matrices!

More generally, one might argue that mobile robots do not need accurate maps. Brooks, for example, would take issue with our assertion that precise (x, y, θ) knowledge of position with respect to a global coordinate frame is necessary[2]. Human beings do not answer the question "where am I?" in millimeters and degrees, so why must a robot? Our rebuttal is that many important applications require quantitative knowledge of position for navigation over large areas (hundreds of square meters) for long periods of time (days, weeks, months), without the aid of artificial beacons or *a priori* maps. This capability has not yet been realized using any sensing modality. We believe that a geometric approach, using sonar sensing, can indeed achieve this capability for a limited class of man-made environments. While the practical utility of such a system may be debated, from the standpoint of *Where is the Science?*, using sonar does allow one to investigate many fundamental aspects of the navigation problem, without having to solve "the vision problem" first.

Acknowledgements

Thanks to Ingemar Cox for many helpful discussions related to this topic. This work has been partially supported by ESPRIT 1560 (SKIDS) and SERC-ACME GR/G 38375.

References

[1] N. Ayache and O. Faugeras. Maintaining representations of the environment of a mobile robot. *IEEE Trans. Robotics and Automation*, 5(6):804–819, 1989.

[2] R. A. Brooks. Elephants don't play chess. *Robotics and Autonomous Systems*, pages 3–15, June 1990.

[3] I. J. Cox and J. J. Leonard. Probabilistic data association for dynamic world modeling: A multiple hypothesis approach. In *International Conference on Advanced Robotics*, pages 1287–1294, Pisa, Italy, 1991.

[4] G. Hager. *Active Reduction of Uncertainty in Multi-Sensor Systems.* PhD thesis, University of Pennsylvania, 1988.

[5] D. Kriegman, E. Triendl, and T. Binford. Stereo vision and navigation in buildings for mobile robots. *IEEE Trans. Robotics and Automation*, 5(6), December 1989.

[6] R. Kuc and M. W. Siegel. Physically based simulation model for acoustic sensor robot navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(6):766–778, November 1987.

[7] J. J. Leonard. *Directed Sonar Sensing for Mobile Robot Navigation.* PhD thesis, University of Oxford, 1990.

[8] J. J. Leonard, I. J. Cox, and H. F. Durrant-Whyte. Dynamic map building for an autonomous mobile robot. In *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, 1990.

[9] J. J. Leonard and H. F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Trans. Robotics and Automation*, 7(3):376–382, June 1991.

[10] J. J. Leonard, H. F. Durrant-Whyte, J. M. Brady, and H. Hu. Sensor guided control of an AGV. In *2nd Int. Symposium on Experimental Robotics*, Toulouse, France, 1991. Springer-Verlag.

[11] L. Matthies and S. Shafer. Error modeling in stereo navigation. *IEEE J. Robotics and Automation*, RA-3(3):239–248, June 1987.

[12] J. Mendle. Computational requirements of a discrete Kalman filter. In H. W. Sorenson, editor, *Kalman Filtering: Theory and Application*. IEEE Press, 1985.

[13] S. Mori, C. Chong, E. Tse, and R. Wishner. Tracking and classifying multiple targets without a priori identification. *IEEE Transactions on Automatic Control*, AC-31(5), May 1986.

[14] P. Moutarlier and R. Chatila. Stochastic multisensory data fusion for mobile robot location and environment modeling. In *5th Int. Symposium on Robotics Research*, Tokyo, 1989.

[15] P. Muir. *Modeling and Control of Wheeled Mobile Robots*. PhD thesis, Carnegie-Mellon University, 1988.

[16] Polaroid Corporation, Commercial Battery Division. Ultrasonic ranging system, 1984.

[17] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In I. Cox and G. Wilfong, editors, *Autonomous Robot Vehicles*. Springer-Verlag, 1990.

Table 1: Comparison of wall locations learned from four scans with actual values, hand-measured to a few millimeters of accuracy. Range is given in meters, orientation in degrees.

Geometric Feature	Estimated		Actual		Difference	
	R	θ	R	θ	R	θ
1	1.706	0.6	1.712	0.0	−0.006	0.6
2	1.408	−89.0	1.402	−89.8	0.006	0.8
3	0.506	−89.8	0.500	−90.0	0.006	0.2
4	−1.013	0.1	−1.000	0.0	−0.013	0.1

Table 2: Comparison of corner locations learned from four scans with hand-measured values. Positions are given in meters, with respect to the first vehicle position.

Geometric Feature	Estimated		Actual		Difference	
	X	Y	X	Y	X	Y
5	−1.011	−1.396	−1.000	−1.406	−0.011	0.010
6	0.812	0.499	0.826	0.500	−0.014	−0.001
7	1.235	−0.994	1.230	−0.992	0.005	−0.002
8	1.651	0.490	1.654	0.500	−0.003	0.010
9	−1.016	0.500	−1.000	0,500	−0.016	0.000

List of Symbols

$\mathbf{x}_R(k)$	true vehicle position
$\hat{\mathbf{x}}_R(k \mid k)$	estimated vehicle position
$\mathbf{P}(k \mid k)$	vehicle position covariance matrix
$\mathbf{x}_i$	true geometric feature location
$\hat{\mathbf{x}}_i(k)$	estimated geometric feature location
$\mathbf{\Lambda}_i(k)$	geometric feature covariance
$M(k)$	set of true feature locations (the true map)
$\hat{M}(k)$	set of estimated feature locations (the estimated map)
$\mathbf{x}(k)$	true system state vector
$\hat{\mathbf{x}}(k \mid k)$	system state estimate
$\mathbf{\Lambda}(k \mid k)$	system state covariance matrix
$\mathbf{C}_{Ri}(k \mid k)$	cross-covariance matrix for vehicle position and feature i
$\mathbf{C}_{ij}(k)$	cross-covariance matrix for features i and j
$\mathbf{u}(k)$	vehicle control input
$\mathbf{d}_k$	true displacement from $\mathbf{x}_R(k-1)$ to $\mathbf{x}_R(k)$
$\hat{\mathbf{d}}_k$	estimated displacement from $\mathbf{x}_R(k-1)$ to $\mathbf{x}_R(k)$
$\mathbf{z}_j(k)$	measurement (observation) at time k
$\hat{\mathbf{z}}_i(k)$	predicted measurement at time k
$Z(k)$	current data set
$\hat{Z}(k)$	current prediction set
Z^k	cumulative data set
$\mathrm{tr}(\mathbf{A})$	The trace of matrix $\mathbf{A}$

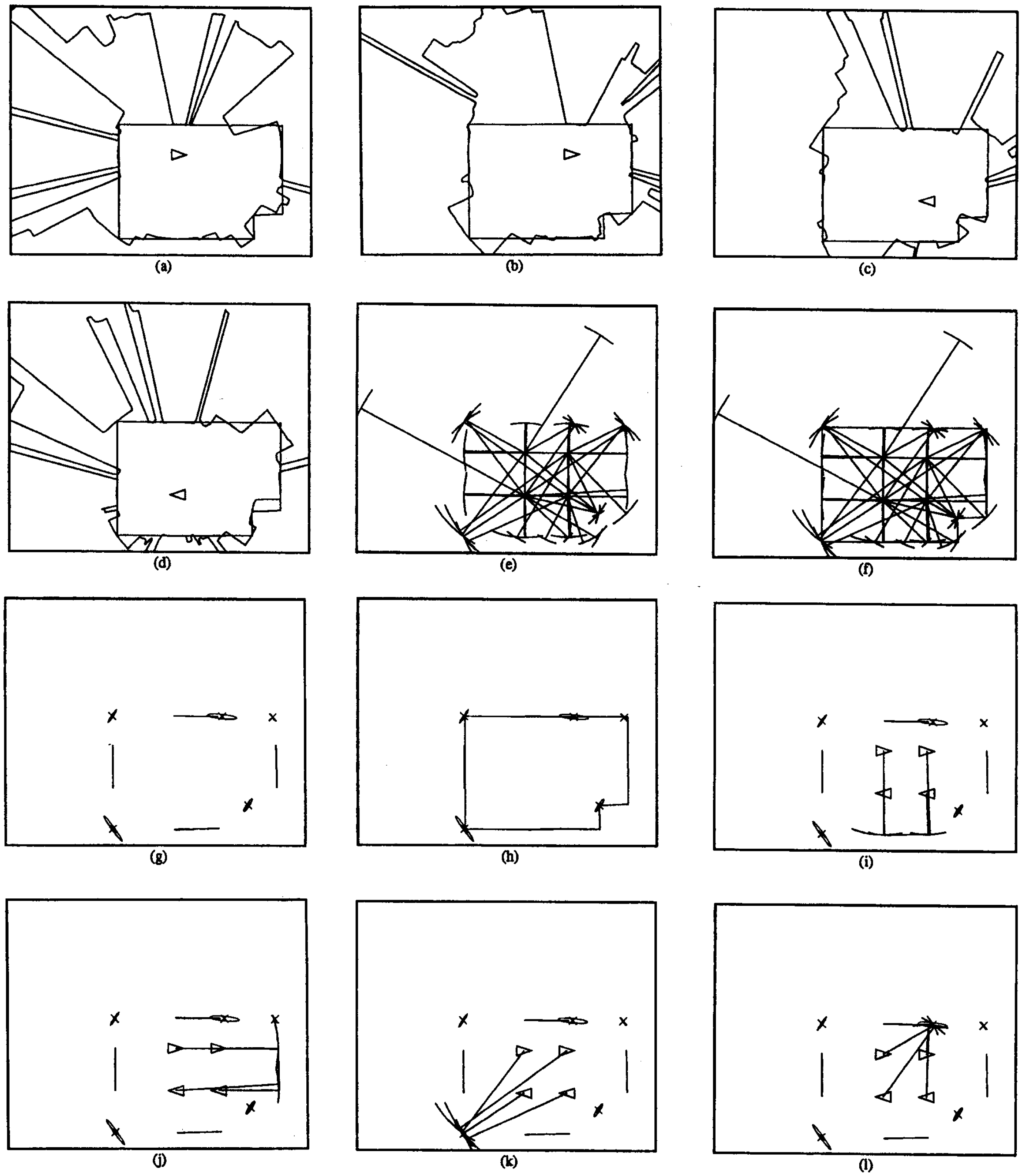

Figure 1: Map building with four sonar scans. (a) through (d) show four sonar scans. Each scan has 612 range measurements and has been median-filtered with a 5 point window. (e) shows the result of extracting Regions of Constant Depth (RCDs) from these sonar scans. (f) shows these same RCDs superimposed on a hand-measured room model. (g) shows the learned map, with 8σ error ellipses for point targets. (h) shows the map superimposed on the room model. (i) and (j) show clusters used to initialize line targets. (k) and (l) show clusters used to initialize point targets.