World Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging

James L. Crowley
Institut National Polytechnique de Grenoble
Grenoble, France

1989 IEEE Conference on Robotics and Automation, ICRA 89, Scottsdale Az., Vol 3, pp 1574-1579, May 1989.

February 1989

This work was sponsored by Transitions Research Corporation and Project EUREKA EU 110: MITHRA

Abstract

This paper describes a system for dynamically maintaining a description of the limits to free space for a mobile robot using a belt of ultrasonic range devices. These techniques are based on the principle of explicitly representing the uncertainty of the vehicle position as well as the uncertainty inherent in the sensing process.

A model is presented for the uncertainty inherent in ultrasonic range sensors, and the projection of range measurements into external Cartesian coordinates is described. A representation for line segments is then presented which permits a segment to be expressed by a set of parameters represented by an estimate and a precision. A process is presented for extracting line segments from adjacent co-linear range measurements and a fast algorithm is presented for matching these line segments to a model of the limits to free-space of the robot.

A side effect of matching observations to a local model is a correction to the estimated position of the robot at the time that the observation was made. A Kalman filter update equation is developed to permit the correspondance of a line segment to the model to be applied as a correction to estimated position. The process for updating the model is then described. Examples of segment extraction, position correction and modeling are presented using real ultrasonic data.

1 Introduction

Perception serves two fundamentally important roles for navigation:

1) Detection of the limits to free space, and
2) Position estimation.

Free space determines the set of positions and orientations that the robot may assume without "colliding" with other objects. Position estimation permits the robot to locate itself with respect to goals and knowledge about the environment.

Unfortunately, the sensing devices which are available for mobile robots often fail in a variety of circumstances. This is especially true of the less expensive devices such as ultrasonic and infrared range sensors. Combining data from several sensors and from a pre-stored model of the domain provides a way to enhance the reliability of a perception system. Such combination may be accomplished by integrating range measurements into a geometric model of the local environment.

In this paper we describe a technique for dynamically modeling the environment of the robot using ultrasonic sensors and a pre-stored world model. We call such a model a "composite local model. Such a model is "composite" because it is composed of several points of view and (potentially) several sources. The model is local, because it contains only information from the immediate environment. A major topic of this paper is the maintenance of such a model.

Navigation is the problem of commanding the robot to locations with respect to the external world. While it is relatively easy to maintain an estimate of the position and orientation of a vehicle using odometry and other proprioceptive sensors, such techniques permit small errors in position and orientation to compound. One of the side effects of the process of updating the composite model is a correction for the estimated position and orientation of the vehicle.

The following section describes our model for the ultrasonic sensor data which is produced by our robot vehicle. In particular, this section presents a model for the uncertainty of a sonar range measurement. Section three presents a representation for line segments in terms of a set of parameters. Each parameter is composed of an estimate and its uncertainty. Such a representation simplifies line segment matching and model updating, particularly in the presence of noise.

Section four presents a technique which uses the the model of the uncertainty of range measurements to extract parametric line segments from adjacent, co-linear range measurements. Section five presents a very simple algorithm which uses the segment uncertainties to match observed segments to the composite local model. Section six shows how a Kalman filter may be used to update the position and orientation of a vehicle from the match of an observed segment to the model. Section seven describes the process for updating the composite local model with an observed segment. Section eight presents examples of these processes running onboard our robot and using its ultrasonic sensors.

2 Modeling Ultrasonic Range Data

The techniques described below have been implemented using both rectangular mobile robot and a circular mobile robot, each equipped with a ring of 24 ultrasonic range sensors. These techniques, however, may be easily extended to include other types of range sensors and other vehicle configurations.

The sensor configuration on our rectangular mobile robot is illustrated in figure 2.1. Each of the four sides of the vehicle carries a set of three parallel sensors. Each corner also carries a set three sensors mounted with orientations of 30°, 45° and 60°. Range data from all 24 sensors are acquired continuously stored in an buffer. This raw range data is used as the basis for reflex level obstacle detection and avoidance. It is also used to construct a description of the visible free-space around the robot known as the "sonar horizon".

The sonar horizon is a array of 24 positions in external Cartesian coordinates. The points in this array are the vertices of a polygon of immediately visible free space around the robot. An uncertainty is stored along with each point in the sonar horizon. The points and their uncertainty are obtained by projecting the range measurement from its observation point according to a model of the sonar ranging process.

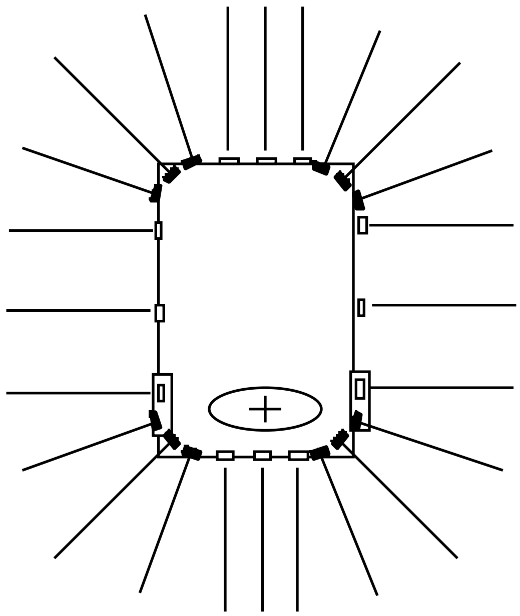

Figure 2.1 Configuration of 24 Ultrasonic Range Sensors

The position and orientation of the sensors with respect to the origin of the robot are defined in a sensor configuration table. For each sensor, the sensor configuration table gives:

r	The distance from the robot's origin to the sensor.
γ	The angle from the robot's axis to the sensor
β	The orientation of the sensor with respect to the robot's axis.

By changing the table, the sensor description algorithm can be made to work with a variety of sensor configurations.

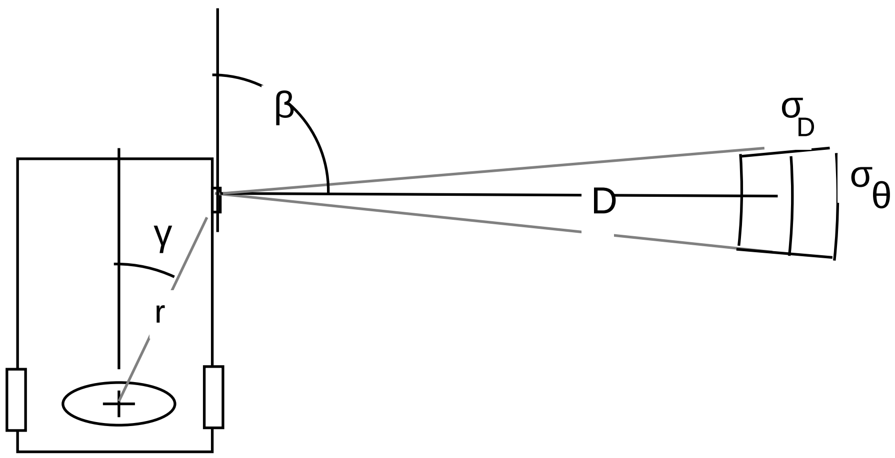

Figure 2.2 Projection of a Range Reading to External Coordinates

A sensor data description process reads range measurements from the sonar table, as well as the estimated position of the robot from the vehicle controller. With this information, the depth measure, d, for each sensor, s, is projected to external coordinates, (x_S, y_S), using the estimated position of the robot, (x, y, α), as shown in figure 2.2.

$$x_S = x + r\ \mathrm{Cos}(\gamma + \alpha) + d\ \mathrm{Cos}(\beta + \alpha)$$
$$y_S = y + r\ \mathrm{Sin}(\gamma + \alpha) + d\ \mathrm{Sin}(\beta + \alpha)$$

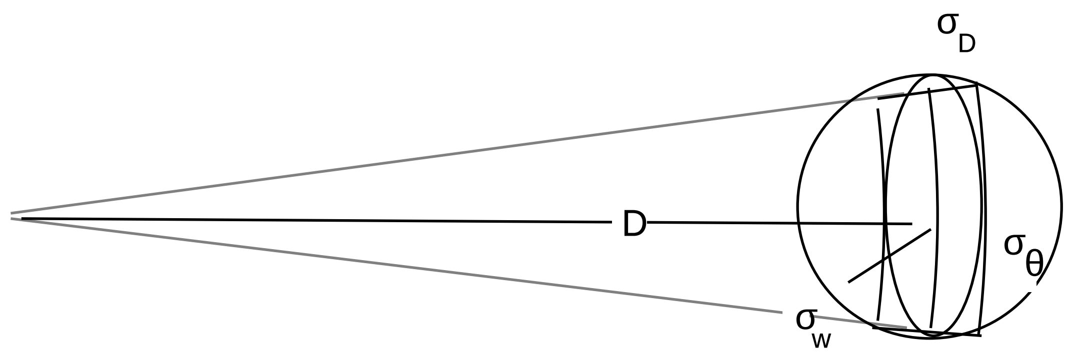

Figure 2.3 Model of the Ultrasonic Range Sensor and its Uncertainties

In order to combine data from different viewpoints and sensors, we must estimate the inherent precision of the data. We have developed a model of an ultrasonic range sensor which predicts that an echo comes from an arc shaped region as illustrated in figure 2.3. This region is determined by the composition of an arc blurred in a perpendicular direction. The length of the arc is given by the uncertainty in orientation, σ_W, while the perpendicular blurring is caused by an uncertainty in depth σ_D.

This crescent shaped uncertainty region may be approximated by an ellipse of which σ_W and σ_D are the major and minor axes [Smith-Cheeseman 87]. This ellipse may be expressed in Cartesian coordinates by a transformation based on the derivative of Cartesian coordinates with respect to the directions of σ_W and σ_D. This derivative is the Jacobien given by

$$J = \frac{\partial\,[\,dx, dy]}{\partial[\,D, \alpha + \beta]} = \begin{bmatrix} -\sin(\alpha+\beta) & \cos(\alpha+\beta) \\ D\cos(\alpha+\beta) & -D\sin(\alpha+\beta) \end{bmatrix}$$

The transformation of this elliptical region to Cartesian coordinates is given by:

$$\mathbf{C}_s = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{bmatrix} = \mathbf{J}^T \begin{bmatrix} \sigma_D^2 & 0 \\ 0 & \sigma_w^2 \end{bmatrix} \mathbf{J}$$

The most important criteria for the sensor data uncertainty is that the estimate of the uncertainty be larger than any true errors. The data interpretation process can be greatly simplified by approximating the covariance, $\mathbf{C_S}$, by a circular variance given by the parameter $\sigma_W{}^2$. In our system, this parameter has been determined, by calibration, to be given by the formula:

$$\sigma_W = 0.10 + d \tan^{-1}(5°) \text{ (in meters)}$$

Thus each reading in the sonar horizon is represented by the triple $S = (x_S, y_S, \sigma_{WS})$ expressed in external coordinates.

3 A Parametric Representation for Line Segments

The modeling process begins by constructing a description of the raw sensor data. This description serves to filter sensor noise by detecting range measurements which are mutually consistent. It also provides a representation with which the estimated position and orientation of the robot may be constrained.

Both the raw ultrasonic range data, and the composite local model are described as parametric line segments [Crowley-Ramparany 87], represented with a data structure, illustrated in figure 3.1. This parametric representation contains a number of redundant parameters which are useful during matching and updating.

A parametric line segment is a structure composed of a minimal set of parameters and a set of redundant parameters. The minimal set of parameters are:

P: Mid-point of the line segment in external coordinates
θ: The orientation of the line segment.
h: The half-length of the line segment.
σ_θ: The uncertainty (standard deviation) in the orientation.
σ_c: Uncertainty in position perpendicular to line segment.

The redundant parameters for a line segment are:

a, b: For the line equation. $a = \mathrm{Sin}(\theta)$, $b = -\mathrm{Cos}(\theta)$
c: The perpendicular distance to the origin. $c = -ax-by$.
d: The distance from the perpendicular intercept to the origin to the mid-point of the segment.
P_r: The end-point to the right of the segment.
P_l: The end-point to the left of the segment.

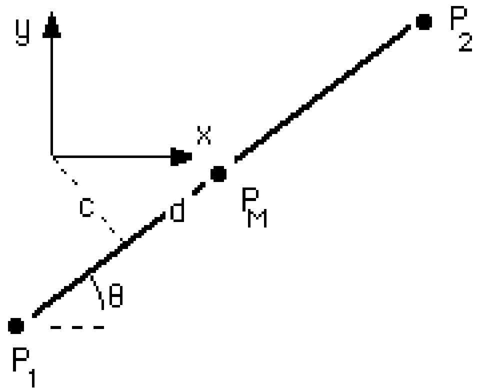

Figure 3.1 The Parametric Representation for a Line Segment.

Line segments in the composite model are also labeled with a confidence factor, noted CF. When a segment is first added to the composite model, whether it is observed or asserted from the pre-learned world knowledge, it is entered with CF = 1. When a new segment (observed or pre-learned) is found to match a model segment, the CF of the model segment is increased by 1 to a maximum of 5. At the end of each sonar scan, the confidence of all segments is reduced by 1. A segment for which the confidence is smaller than 0 is removed from the model.

4 Finding Line Segments in Range Data

Ultrasonic range data are seriously corrupted by reflections and specularities. One researcher [Brown 85] recently compared sensing with ultrasonics to trying to navigate in a house of mirrors using only a flash light. To overcome these effects we use redundancy.

The first source of redundancy is alignment. Reflected readings rarely align. Thus points that are aligned are likely to correspond to actual surfaces. A second source of redundancy is mobility. As the observation position changes, depth measurements based on reflections project to widely varying surfaces, while measurements corresponding to a surface project to that surface. Updating the model from different viewpoints provides a technique by which correct range measures reinforce each other, while measures based on reflections do not.

Line segments are formed in the external coordinate system of the robot using the model of the uncertainty of the ultrasonic sensors described above. Forming line segments in external coordinates permits the integration of range measurements while the robot is moving. The uncertainty of the robot's position is not used in forming line segments because this error is common to the depth measurements. After a segment has been detected and formed, the uncertainty of the robot's position is added to the segment.

4.1 Detecting Segments in Noisy Range Data

To avoid defining line segments with errorful data, a line segment is not formed unless three consecutive depth readings align within a tolerance. The tolerance is provided by the uncertainty parameter σ_W of the range measurement. The process for detecting line segments begins by testing the distance between projections of the depth from sensors S_{i-1}

to S_i, and the distance from the projection of S_i to S_{i+1}. Both distances must be less than the width of the robot. This criteria prevents the process from closing off passages through which the robot might wish to pass.

If the projections are sufficiently close, then the equation of the line passing through S_{i-1} and S_{i+1} is expressed as the normalized line coefficients a, b, and c. The perpendicular error from this line for reading S_i is then computed as

$$e = a\, x_i + b\, y_i + c$$

To form a line segment, the magnitude of this error must be less than or equal to half of the uncertainty, σ_{wi}.

$$| e | \leq \sigma_{wi} / 2.$$

If this test is passed a segment is created from the three points.

The uncertainty perpendicular to the segment is computed by the formula [Durrant-Whyte 87]:

$$\sigma_c^2 = \frac{\sigma_{wi+1}^2 \; \sigma_{wi-1}^2}{\sigma_{wi+1}^2 + \sigma_{wi-1}^2}$$

The middle reading, S_i, provides a constraint on the perpendicular distance of the segment from the origin, c, as well as on σ_c^2. This constraint is applied using a Kalman gain term, computed by

$$k_c = \frac{\sigma_c^2}{\sigma_c^2 + \sigma_{wi}^2}$$

The parameters c and σ_c^2 are then updated by:

$$c = c - k_c\, e$$

$$\sigma_c^2 = \sigma_c^2 - k_c\, \sigma_c^2$$

This correction is also applied to the center point by composing a correction vector $M = [a, b]^T$.

$$P_m = P_m - k_c\, e\, M$$

The orientation of the segment is computed using the end points. The uncertainty in the orientation is based on the uncertainty perpendicular to the segment as well as the half-length of the segment.

$$\sigma_\theta = \mathrm{Tan}^{-1} (\sigma_c / h).$$

The location of the points S_{i-1} and S_{i+1} are saved as the end points of the segment.

The line segment creation process is applied successively to the sensors in counter-clockwise order. If it succeeds in detecting a segment, then successive points in counter clockwise order are tested to see if they belong to the segment.

4.2 Extension of Segments

For point (x_j, y_j) to be included in a line segment, its perpendicular error, e, must be less than or equal to half of the uncertainty, σ_{wj}.

$$| e | = | a x_j + b y_j + c | \leq \sigma_{wj} / 2.$$

If this test is passed, the segment parameters c and σ_c^2 are updated by the equations given above. The orientation is corrected by determining $\Delta\theta$ and $\sigma_{\theta i}$ as

$$\Delta\theta = \mathrm{Tan}^{-1} (e / h)$$

$$\sigma_{\theta j} = \mathrm{Tan}^{-1} (\sigma_{wj}/ h)$$

The orientation θ and its uncertainty, σ_θ^2, are then updated by

$$k_\theta = \sigma_\theta^2 / (\sigma_\theta^2 + \sigma_{\theta j}^2)$$

$$\theta = \theta - k_\theta \Delta\theta$$

$$\sigma_\theta^2 = \sigma_\theta^2 - k_\theta \sigma_\theta^2$$

The line equation parameters are recomputed as using the new value of θ. The new P_j replaces the left most end point P_l. Both the rightmost end point, P_r, and the new point, P_j, are projected onto the line segment. For a point P = (x, y), such a projection to point P' = (x', y') is given by:

$$x' = x b^2 - y ab - a c$$

$$y' = -x a b + y a^2 - b c$$

A new half-length and midpoint are computed from the projected endpoints.

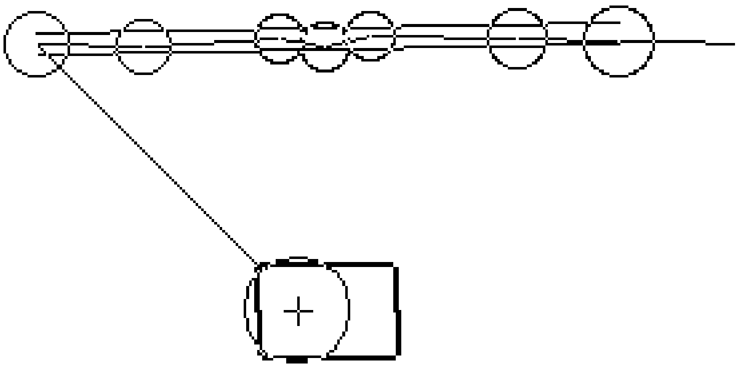

Figure 4.2a A vehicle (rectangle with cross) with an circular uncertainty in position of 40 cm (circle surrounding cross) is shown detecting a line segment. The top image shows a set of range data in which a line segment has been detected. The projections of ultrasound readings are illustrated as circles of radius σ_W. The projections provide the vertices of the sonar horizon. The detected segment is illustrated by a pair of parallel line at $\pm \sigma_C$.

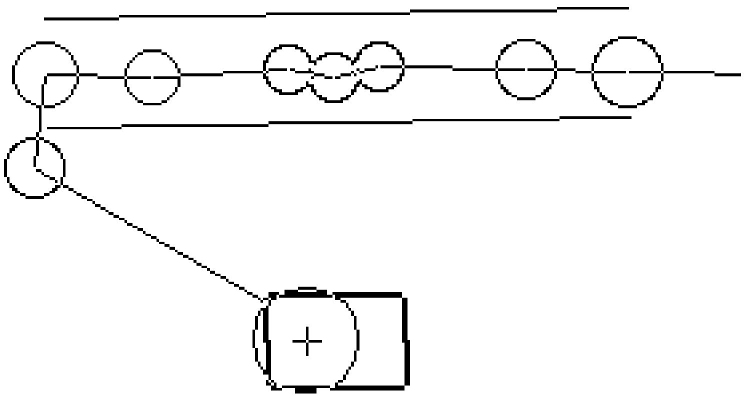

Figure 4.2b This image shows the segment after the uncertainty in the robot's position has been added to the segment uncertainties, σ_c^2 and σ_θ^2

The process halts when a point fails the test for inclusion in a segment. When the process halts, it returns the parameters of the segment. Figure 4.2a illustrates this process with crop from a screen dump from an execution of the program.

4.2 Adding the Uncertainty of the Robot to a Segment

The projection of a segment into the external coordinate system is based on the estimate of the position of the vehicle. Any uncertainty in the vehicle's estimated position must be included in the uncertainty of the segment before matching can proceed. This uncertainty affects both the position and orientation of the line segment. The position uncertainty of the segment is increased by computing the component of the robot's uncertainty in the direction of the uncertainty σ_c, that is perpendicular to the segment. This uncertainty σ_{cr} is determined from the product of the vector $\mathbf{M} = [a \;\; b]^T$ and the covariance of the robot's estimated position.

$$\begin{aligned} \sigma_r^2 &= \mathbf{M}^T \mathbf{C}_{xy} \; \mathbf{M} \\ &= [a \;\; b] \;\; \mathbf{C}_{xy} \, [a \;\; b]^T \\ &= a^2 \, \sigma_x^2 + 2 \, a \, b \, \sigma_{xy} + b^2 \, \sigma_y^2 \end{aligned}$$

This is added to the perpendicular uncertainty of the segment by

$$\sigma_c^2 = \sigma_c^2 \; + \; \sigma_r^2$$

The uncertainty in orientation of the robot, σ_α^2, is added to the segment by

$$\sigma_\theta^2 = \sigma_\theta^2 + \; \sigma_\alpha^2$$

This is illustrated in figure 4.2b.

5 Matching a Segment to the Composite Model

As each segment is obtained from the ultrasound data it is matched to the composite model. Matching is a process of comparing each of the segments in the composite local model against the observed segment to detect similarity in orientation, colinearity, and overlap. Each of these tests is made by comparing one of the parameters in the segment representation.

Given an observed segment, S_o, with parameters $\{x_o, y_o, C_o, h_o, \theta_o\}$ with uncertainties $\{\sigma_{co}, \sigma_{\theta o}\}$ and given a Model segment, S_m, with parameters $\{x_m, y_m, C_m, h_m, \theta_m\}$ with uncertainties $\{\sigma_{cm}, \sigma_{\theta m}\}$, the comparison begins with a test for similar orientation.

$$(\theta_m - \theta_o)^2 \leq \sigma_{\theta o}^2 + \sigma_{\theta m}^2.$$

If the result of this test is false, the process advances to the next segment in the composite local model. Otherwise, the segments are compared for alignment by testing

$$(c_m - c_o)^2 \leq \sigma_{co}^2 + \sigma_{cm}^2.$$

If this test is true then the two segment are tested for overlap by comparing the distance between center-points to the sum of the half lengths.

$$(x_o - x_m)^2 + (y_o - y_m)^2 \leq h_o + h_m.$$

The longest segment in the composite model which passes all three tests is selected as the matching segment. The segment is then used to correct the estimated position of the robot and to update the model.

6 Updating the Estimated Position of the Robot.

Each match of an observed segment provides an one dimensional constraint on the position of the robot, **P**, and its uncertainty $\mathbf{C}_{xy}$, as well as a constraint on the orientation of the robot. These constraints may be applied using a form of Kalman filter update formula [Melsa-Sage 73].

The correction to position is provided by the difference in perpendicular position, $\Delta c = c_m - c_o$. The precision of this correction is provided by the variances in perpendicular position from the model and the observed segment, σ_{co}^2 and σ_{cm}^2. This correction applies only in the direction perpendicular to the segment, defined by the vector $M = [a \quad b]^T$ from the observed segment. The component of the robot's uncertainty in this direction is already included in the perpendicular uncertainty of the segment, σ_{co}^2. Thus a Kalman gain vector for the estimated position is given by

$$\mathbf{K}_p = \mathbf{C}_{xy} \; M \; 1 / (\sigma_{co}^2 + \sigma_{cm}^2)$$

Position correction may be given by Δc as described by the gain vector:

$$P = P - \mathbf{K}_p \; \Delta c$$

However, our vehicle controller is designed to accept a correction vector $\Delta P = [\Delta x, \Delta y]$, anda gain matrix $\mathbf{K}_{xy}$. The gain matrix is given by expanding $\mathbf{K}_p$ with a cross product with M^T.

$$\mathbf{K}_{xy} = \mathbf{K}_p \; M^T$$

The correction vector is given by $\Delta P = M\ \Delta c$. The position is then corrected by

$$P = P - \mathbf{K}_{xy}\ \Delta P$$

Which may be seen as equivalent to $P = P - \mathbf{K}_p\ \Delta c$ by noting that $M^T M = 1$. The vehicle's position covariance is then updated by

$$\mathbf{C}_{xy} = \mathbf{C}_{xy} - \mathbf{K}_{xy}\ \mathbf{C}_{xy}.$$

The vehicle's estimated orientation is updated by computing a Kalman gain, k_α:

$$k_\alpha = \sigma_\alpha^2 / (\sigma_{\theta m}^2 + \sigma_{\theta o}^2)$$

The orientation is then updated by the difference in angle between the model and observed segment:

$$\alpha = \alpha - k_\alpha (\theta_o - \theta_m).$$
$$C_\alpha = C_\alpha - k_\alpha C_\alpha$$

The matching, correction and update process are illustrated in figure 6.1 and 6.2.

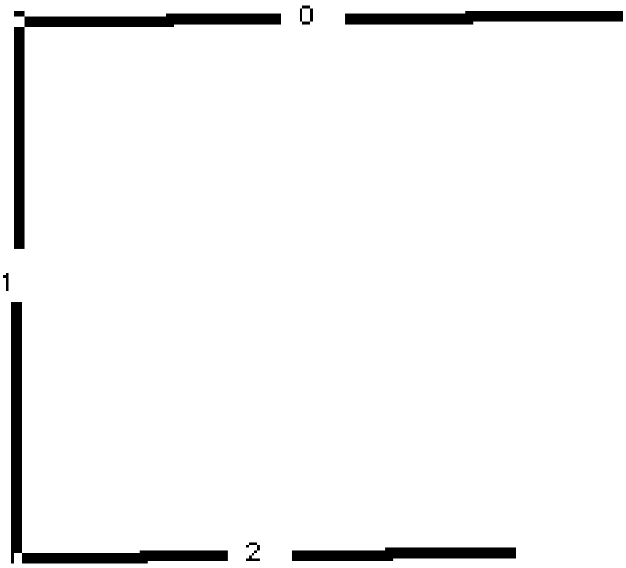

Figure 6.1 Continuing the example begun in figure 4.2, this crop from a screen dump shows three segments from the composite local model just before the model was updated by the segment detected in figure 4.2. An integer number indicates the identity of each segment. The width indicates the confidence

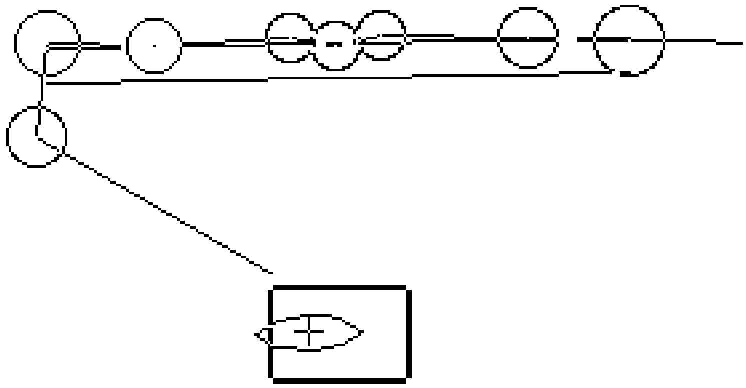

Figure 6.2 This figure shows the uncertainty in position after correction by matching with model segment number 0. The uncertainty is reduced to an ellipse with a minor axis of approximately 8 cm. An error of (x, y) = (20 cm, 20 cm) had been introduced to the estimated position before the segment was detected. Following matching, both the position of the robot and the observed segment have been shifted by $\Delta y = 17.7$ cm by the update process to correct for the error.

7 Updating the Composite Local Model

An observed segment acts as a constraint on the orientation and position uncertainty of segments in the model. At the same time, an observed segment serves to enlarge the spatial extent of a segment and to increase its confidence. Given that observed segment S_o matches model segment S_m, the parameter c, of S_m is corrected by computing a Kalman gain, K_c.

$$k_c = \sigma_{cm}^2 / (\sigma_{co}^2 + \sigma_{cm}^2)$$

From which the parameter and its uncertainty are updated

$$c_m = c_m - k_c (c_o - c_m)$$

$$\sigma_{cm}^2 = \sigma_{cm}^2 - k_c \sigma_{cm}^2$$

In a similar manner, the orientation of the model segment in updated by

$$k_\theta = \sigma_{\theta m}^2 / (\sigma_{\theta o}^2 + \sigma_{\theta m}^2)$$

$$\theta_m = \theta_m - k_\theta (\theta_o - \theta_m)$$

$$\sigma_{\theta m}^2 = \sigma_{\theta m}^2 - k_\theta \sigma_{\theta m}^2$$

Having determined θ, the new line equation coefficients are computed. The segment is extended by projecting the previous end points and the observed endpoints on to the segment. The two most distant points are selected as the new endpoints, P_l and P_r. The new midpoint and half length of the segment may be computed from the new end-points.

8 Sample Results

The vehicle controller includes a command to reset the estimated position, orientation and uncertainties. This command makes it possible to perform experiments with the ability of the system to correct errors in the estimated position. Such experiements are performed on a daily basis in our laboratory. This section presents the results from three such experiements.

In each of these experiments, the robot was placed in a corner of the laboratory, with relatively flat objects on three sides. The environment can be explained in terms of the composite model segments shown in figure 6.1, which are labeled with an index number. Segment 0 corresponds to a wall covered with a textured wall paper. Segment 1 corresponds to a metal cabinet with a sliding plastic door. Segment 2 corresponds to a set of laboratory chairs pushed up against two tables. The chairs are plastic with metal legs. The system has no a-priori knowledge of its environment. The location and orientation at which the sytem was started were taken as the origin and x axis of the external coordinate system. In each experiment, the system was allowed to run for 3 cycles of ultrasound acquisition and then an error was introduced in the estimated position.

8.1 Correction of Position Error

In the first experiment, after three cycles, the estimated position of the vehicle was set to (10 cm, 10 cm) and the orientation was set to 0.0. The uncertainty was set to a standard deviation of 20 cm in x and y, with an uncertainty in orientation of 0 degrees. The system was then allowed to detect the segments around it in a clockwise order. The estimated position and covariance after each cycle are shown in table 8.1.

Initial :				
estimated position:	x, y, α =	(0.100,	0.100,	0.0)
covariance:		0.040	0.000	0.000
		0.000	0.040	0.000
		0.000	0.000	0.000
After Match with Segment 0:				
estimated position:	x, y, α =	(0.101,	0.017,	0.0)
covariance:		0.039	0.000	0.000
		0.000	0.010	0.000
		0.000	0.000	0.000
After Match with Segment 1:				
estimated position:	x, y, α =	(0.031,	0.017,	0.0)
covariance:		0.010	0.000	0.000
		0.000	0.010	0.000
		0.000	0.000	0.000
After Match with Segment 2:				
estimated position:	x, y, α =	(0.031,	0.016,	0.0)
covariance:		0.010	0.000	0.000
		0.000	0.006	0.000
		0.000	0.000	0.000

Table 8.1 Position Correction Experiement

8.2 Correction of Orientation Error

In the second experiment, after three cycles, the estimated orientation of the vehicle was set to 5 degrees. The uncertainty was set to a standard deviation of 10 cm in x and y, with an uncertainty in orientation of 10 degrees. The system was then allowed to detect the segments around it in a clockwise order. The estimated position and covariance after each cycle are shown in table 8.2

Initial :

estimated position: x, y, α = (0.000, 0.000, 5.0)

covariance:

0.010	0.000	0.000
0.000	0.010	0.000
0.000	0.000	100.000

After Match with Segment 0:

estimated position: x, y, α = (-0.000, 0.004, 1.6)

covariance:

0.009	0.000	0.000
0.000	0.006	0.000
0.000	0.000	29.615

After Match with Segment 1:

estimated position: x, y, α = (0.010, 0.004, 0.30)

covariance:

0.005	0.000	0.000
0.000	0.006	0.000
0.000	0.000	16.167

After Match with Segment 2:

estimated position: x, y, α = (0.010, 0.011, 0.00)

covariance:

0.005	0.000	0.000
0.000	0.004	0.000
0.000	0.000	11.301

Table 8.2 Orientation Correction Experiment

8.3 Correction of Position and Orientation Error

In the third experiment, after three cycles, both the estimated position and the estimated orientation of the vehicle were set to false values. The position was set to (0.10, 0.10) and the orientaiton was set to 5 degrees. The uncertainty was set to a standard deviation of 20 cm in x and y, with an uncertainty in orientation of 10 degrees. The system was then allowed to detect the segments around it in a clockwise order. The estimated position and covariance after each cycle are shown in table 8.3.

Initial :

estimated position: x, y, α = (0.100, 0.100, 5.0)

covariance:

0.040	0.000	0.000
0.000	0.040	0.000
0.000	0.000	100.000

After Match with Segment 0:

estimated position: x, y, α = (0.102, 0.019, 1.3)

covariance:

0.039	0.000	0.000
0.000	0.010	0.000
0.000	0.000	26.287

After Match with Segment 1:

estimated position:	x, y, α = (0.033, 0.017, 0.20)			
covariance:		0.010	0.000	0.000
		0.000	0.010	0.000
		0.000	0.000	17.106

After Match with Segment 2:

estimated position:	x, y, α = (0.034, 0.012, 0.00)			
covariance:		0.010	0.000	0.000
		0.000	0.005	0.000
		0.000	0.000	11.703

Table 8.3 Correction of orientation and position

9 Discussion

Two conclusions can be drawn from this system:

1) An explicit model of uncertainty using covariances and Kalman filtering provides a tool for integrating noisy and imprecise sensor observations into a model of the geometric limits to free space of a vehicle.

2) Such a model provides a technique for a vehicle to maintain an estimate of its position as it travels, even in the case where the environment is unknown.

Concerning the first point, this is the third system which we have constructed using these mathematical techniques. Earlier systems include a technique for combining data from 3-D sensors [Crowley 86] and a technique for measuring the movement of edge lines in motion sequences [Crowley 88]. A number of other authors report similar positive results with these tools. Matthies et al have used similar techniques for motion and stereo [Matthies-Shafer 87]. Durrant-Whyte has used similar techniques for combining touch and stereo [Durrant-Whyte 87]. Faugeras and Ayache have used similar techniques for 3-D modeling using sequences of stereo images [Faugeras-Ayache 86].

Concerning the second point, a reviewer has remarked that correcting position and updating a model from the same data is a bit like pulling your self up by your bootstraps. Such a technique is possible provided that the their is a substantial overlap between the observed data and the current model. This overlap provides the basis for position correction, while the new information provides the basis for enlarging the model. In its normal mode of operation the local model is completed by a prestored model of the environment. While this prestored model is only a partial description of the environment, it assures that there is sufficient overlap between perceptions and the local model for the vehicle to maintain its estimated position.

Acknowledgements

The code for the segment extraction process was written and debugged by Francois Chabanel, Francois Royer, Pascal Sautot and the author. An earlier version of this segment extraction process was written by Franck Hocquet, Hervé Fontainé, Renaud Zigmann. The architecture and concepts of the composite local model were developed at Carnegie Mellon

University in projects sponsored sponsored by Commodore Business Machines, and Denning Mobile Robotics. This current system was sponsored by Transitions Research Corporation and Project EUREKA EU 110: MITHRA.

Bibliography

[Brown 85] Brown, M. K., "Feature Extraction Techniques for Recognizing Solid Objects with an Ultrasonic Range Sensor, IEEE Journal of Robotics and Automation, December, 1985.

[Chatila 85] Chatila, R. and J. P. Laumond, "Position Referencing and Consistent World Modeling for Mobile Robots", Proc of the 2nd IEEE Conf. on Robotics and Automation, St. Louis, March 1985.

[Crowley 86] Crowley, J. L., "Representation and Maintenance of a Composite Surface Model", IEEE Int. Conf. on Robotics and Automation, San Francisco, April 1986.

[Crowley 87] Crowley, J. L., "Coordination of Action and Perception in a Surveillance Robot", Proc. of the 1987 IJCAI, Milan, Italy, Kaufmann Press, August, 1987, (also appeared in IEEE Expert, Nov. 1987).

[Crowley-Ramparany 87] Crowley, J. L. and F. Ramparany, Mathematical Tools for Manipulating Uncertainty in Perception", AAAI Workshop on Spatial Reasoning and Multi-Sensor Fusion", Kaufmann Press, October, 1987.

[Crowley et. al. 88] Crowley, J. L., P. Stelmaszyk and C. Discours, "Measuring Image Flow by Tracking Edge-Lines", ICCV 88: 2nd International Conference on Computer Vision, Tarpon Springs, Fla., Dec. 1988.

[Durrant-Whyte 87] Durrant-Whyte, H. F., "Consistent Integration and Propagation of Disparate Sensor Observations", Int. Journal of Robotics Research, Spring, 1987.

[Elfes 87] Elfes, A. "Sonar Based Real World Mapping and Navigation", IEEE Journal of Robotics and Automation, Vol RA-3 (3), June 1987.

[Faugeras, et. al. 86] Faugeras, O. D. , N. Ayache, and B. Faverjon, "Building Visual Maps by Combining Noisey Stereo Measurements", IEEE Int. Conf. on Robotics and Automation, San Francisco, April 1986

[Matthies-Shafer 87], Matthies, L. and S. A. Shafer, "Error Modeling in Stereo Navigation", IEEE Journal of Robotics and Automation, Vol RA-3 (3), June 1987.

[Melsa-Sage 73] Melsa, J. L. and A. P. Sage, An Introduction to Probability and Stochastic Processes, Prentice Hall, 1973.

[Smith-Cheeseman 87], Smith, R. C. and P. C. Cheeseman, "On the Estimation and Representation of Spatial Uncertainty", International Journal of Robotics Research 5 (4), Winter, 1987.