A Comparison of Position Estimation Techniques Using Occupancy Grids

Bernt Schiele and James L. Crowley
LIFIA (IMAG) - I.N.P. Grenoble
46 Avenue Félix Viallet
38031 Grenoble Cedex
FRANCE

Abstract

A mobile robot requires perception of its local environment for both sensor based locomotion and for position estimation. Occupancy grids, based on ultrasonic range data, provide a robust description of the local environment for locomotion. Unfortunately, current techniques for position estimation based on occupancy grids are both unreliable and computationally expensive. This paper reports on experiments with four techniques for position estimation using occupancy grids.

A world modeling technique based on combining global and local occupancy grids is described. Techniques are described for extracting line segments from an occupancy grid based on a Hough transform. The use of an extended Kalman filter for position estimation is then adapted to this framework. Four matching techniques are presented for obtaining the innovation vector required by the Kalman filter equations. Experimental results show that matching of segments extracted from the both the local and global occupancy grids gives results which are superior to a direct matching of grids, or to a mixed matching of segments to grids.

1 Introduction

Occupancy grids (or certainty grids) were first proposed by Moravec and Elfes as a way to construct an internal model of static environments based on ultrasonic range data [8], [9]. This method takes into account the uncertainty of sensory data by working with probabilities or certainty values. The occupancy grid representation can be used directly in robotic planning [10] or navigation [7]. Other authors have used a certainty grid method for collision avoidance [4], [3]. Two drawbacks of the occupancy grid method have so far not been solved satisfactorily. One is the modelling of dynamic obstacles and the second is the position estimation process for the robot vehicle. This paper report the results of an investigation into the position estimation problem.

Moravec [8] attempted to solve the position estimation problem by multi-resolution matching. The result was an algorithm which was quite costly from a computational standpoint, and often gave wrong results. We have come back to this problem, applying lessons learned from position estimation using a local model based on lines segments [5]. In our framework, the environment is modelled with two Occupancy Grids. The first grid is centered on the robot and models its vicinity using the most recent sensor readings. The second grid is a global model of the environment furnished either by learning or by some form of computer aided design tool. The position and orientation (or pose) at which the local model best matches the global model provides an innovation vector for a Kalman filter update of the estimated position of the robot. The problem addressed in this paper is how best to perform the matching.

Section 2 introduces the modelling of the environment using two occupancy grids. The two following sections describe the position estimation procedures we have implemented in our robot. Section 3 describes the matching procedures for the position estimation and section 4 describes updating of the estimated robot position using a Kalman filter. Section 5 introduces the integration of the local occupancy grid data into the global occupancy grid. Experimental results are described and discussed in section 6. Our results show that it is best to extract line segment descriptions from the two grids and then match these descriptions.

2 Occupancy Grids

Two representations have been demonstrated for dynamic modeling of the environment of a mobile robot using ultrasonic range data: Parametric primitives [12] and occupancy grids [8]. Parametric primitives describe the limits to free-space in terms of segments or surfaces defined by a list of parameters. Such a description is easily entered by hand and displayed on a computer terminal. Parametric primitives are well adapted to local path planning and to position estimation using an extended Kalman Filter [5]. Unfortunately, noise in the sensor signals can make the process for grouping sensor readings to form geometric primitives somewhat unreliable. In particular, small obstacles such as table legs are practically impossible to distinguish from noise.

The occupancy grid method for modelling a mobile robot environment avoids the problem of grouping adjacent sensor measures. The occupancy grid is also well

adapted to local path planning and reactive locomotion using a variety of algorithms. The principle disadvantage has been the difficulty in using grids to correct a vehicle's position estimation.

In the occupancy grid method, space is represented by a regular grid with each cell holding a certainty value that a particular patch of space is occupied. The following section introduces a modelling of a robot environment based on two occupancy grids. The first is centered on the robot and models the vicinity of the robot. The second has global parameters and models the entire environment. In these experiments the global model was formed by "exploring" the environment.

2.1 Local Occupancy Grid

The local occupancy grid contains information about the environment of the robot which is directly perceivable. Therefore, the grid is centered on the robot and the robot integrates the sensory data into the local grid. This local occupancy grid is suitable to avoid collisions with static and with dynamic obstacles. To avoid collisions, the updating process of the local occupancy grid must have two important characteristics: On the one hand, the updating process must be rapid so that we can update frequently. And on the other hand, the data of the local grid becomes outdated. Consequentially, the information in the local occupancy grid must decay with time.

The region covered by the grid, and the required resolution depend on the velocity of the robot. When the robot is moving relatively fast, the region of the local grid must grow at the expense of the resolution. It is obvious that we do not obtain the best correction of the estimated robot position with a poor resolution. However, at the moment of high robot velocity, it is much more important to avoid collisions with static and dynamic obstacles. The opposite is true for a low vehicle velocity where we have a smaller region and a better resolution of the local grid. With such a local grid we can correct the estimated position more accurately. Lower velocities are typical in the vicinity of the goal position so that the robot can reach this position accurately.

2.2 Global Occupancy Grid

The global occupancy grid is a two-dimensional representation of the static environment. The initial state of the global occupancy grid is completely unknown because we do not provide an a-priori model of the environment. While moving, the robot explores the environment by integrating the observations into the global grid. As described in 2.1 the robot integrates the observations into the local occupancy grid that models the vicinity of the robot. The updating process of the global occupancy grid integrates the data of the local occupancy grid into the global occupancy grid as described above in section 5.

The global occupancy grid can be used for global path-planning and together with the local grid to correct the estimated position of the robot. We are currently investigating techniques to model dynamic objects in this global grid. Zhang and Webber [11] have proposed a modified Hough-transform to detect moving objects. However the described method currently works only with a few integer velocities and is relatively time consuming.

2.3 Finding Line Segments in Occupancy Grids

A characteristic of man made environments is that objects tend to lie in straight lines. Examples are walls and doorways. It is possible to use line segments for the correction of the estimated position as explained in section 3. Such straight lines can be found in the occupancy grids as aligned cells of high probability of occupation. By interpreting a grid and its probabilities as an image with different levels of resolution we can apply image processing operations the detection of straight lines.

Our method for searching straight lines is a form of Hough-transform. The ordinary Hough- transform [2] is based on a function shown in equation (1). In this function, (x, y) are the coordinates of a point and (C, θ) are the parameters of a straight line. As illustrated in figure 1, C is the perpendicular distance of the straight line to the origin and θ is the angle of the normal with the x-axis.

The function (1) computes for a certain (C, θ)-couple all points ((x, y)-couples) which lie on the same straight line. But this function computes as well for a certain point ((x, y)-couple) all the straight lines ((C, θ)-couples) passing that point. The Hough-transform enters for a certain point (possible member of a straight line) all straight lines passing that point into a Hough-parameter space (which is indexed with C and θ). Such a point is in our case a grid-cell with a high probability to be occupied. Local maxima of the Hough-space represent straight lines which exist in the grid.

$$f(x; y; C, \theta) = x \cos\theta + y \sin\theta + C = 0 \qquad (1)$$

Our extension of the Hough-transform is to calculate a probability for a (C, θ)-couple and to calculate the uncertainty $(\sigma_c^2, \sigma_\theta^2)$ of that couple. For a (C, θ)-couple representing a infinite line, we find the beginning and ending point of the line segment lying in our occupancy grid. To arrive at the probability, we look at the probabilities of the grid-cells lying on the line segment and in a small region around the line segment. The calculation of the uncertainty $(\sigma_c^2, \sigma_\theta^2)$ of a (C, θ)-couple is based on the Hough-space. The (C, θ)-couple of a line segment is a local maxima in the Hough-space. We calculate $(\sigma_c^2, \sigma_\theta^2)$ as the variance of the distribution in the Hough-space around the local maximum.

c: Perpendicular distance from the segment to the origin.
σ_C: Uncertainty in position perpendicular to line segment.
(x, y): The mid-point of the line segment
θ: Orientation of the line segment.
σ_θ: Standard deviation in the orientation.
h: Half-length of the line segment.

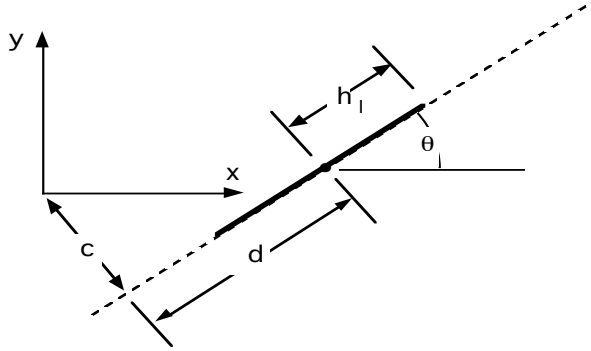

Figure 1 The Parametric Representation for a Line Segment.

3 Matching Occupancy grids

The estimated position of the robot $\hat{P}_r = (\hat{x}_r, \hat{y}_r, \hat{\theta}_r)^T$ and its covariance, $\hat{\boldsymbol{C}}_r$ are estimated by a vehicle controller using a Kalman filter driven by the wheel encoders [Cro 93]. This vehicle controller accepts a command to update the estimated position and its covariance. T he parameters of this command are the "innovation" vector ΔP_r and its covariance $\boldsymbol{C}_{\Delta p}$. In this section we describe four different methods to obtain the innovation and its covariance by matching a local grid to a global grid. These four processes can be summarized as matching "segment to segment", "grid to segment", "segment to grid" and "grid to grid".

The first of our matching processes involves matching segments extracted from the local grid to segments extracted from the global grid. This process is composed of three parts:

1. Finding line segments in the two grids with our extended Hough-transform,
2. Matching the segments (see 3.1) and
3. Applying a Kalman filters update for the correction of the orientation θ_r and the position $(x_r, y_r)^T$ of the robot (described below in section 4.1 and in [5]).

As described in 2.3, line segments are fit to the in the grids using a Hough-transform.

The other re-localisation processes search the best transformation $T = (\Delta x, \Delta y, \Delta\theta)^T$ of the local grid into the global grid. Three matching procedures are described to find the best transformation T and its covariance $\boldsymbol{C}_T$:

- Matching local segments directly against the global occupancy grid, described in 3.2,
- Matching global segments directly against the local occupancy grid described in 3.3 and
- Matching directly the two occupancy grids as described in section 3.4.

After we have found such a transformation T and its covariance $\boldsymbol{C}_T$ we apply a Kalman filter to correct the robot position as described below in section 4.2.

3.1 Matching Local Segments against Global Segments

The first re-localisation process that we tested is based on extracting and matching segments from the local and global occupancy grids. Segments represent the limits to free-space and are obtained with the extended Hough-transform. As each segment is extracted from the local grid it is matched matched against segments obtained from the global grid using a form of Mahalanobis distance.

Let S_{lo} represent a segment from the local grid, and let S_{gl} represent a segment from the global grid. These segments are represented by the following parameters as defined above:

$$S_{lo} : (\theta_{lo}; \sigma^2_{\theta\, lo}; C_{lo}; \sigma^2_{C\, lo}; x_{lo}; y_{lo}; h_{lo})$$
$$S_{gl} : (\theta_{gl}; \sigma^2_{\theta\, gl}; C_{gl}; \sigma^2_{C\, gl}; x_{gl}; y_{gl}; h_{lgl})$$

Matching is a process of comparing each pair of segments to detect similarity in orientation (2), colinearity (3) and overlap (4). The difference in orientation, squared, must be less than the sum of the variances in orientation.

$$(\theta_{gl} - \theta_{lo})^2 \le \sigma^2_{\theta\, lo} + \sigma^2_{\theta\, gl} \qquad (2)$$

If the segments have similar orientation, then their colinearity can be tested by testing the perpendicular distance from the origin.

$$(C_{lo} - C_{gl})^2 \le \sigma^2_{C\, lo} + \sigma^2_{C\, gl} \qquad (3)$$

Finally, we can test for overlap by comparing the Euclidean distance between the center points to the sum of the half-lengths.

$$(x_{gl} - x_{lo})^2 + (y_{gl} - y_{gl})^2 \le (h_{gl} + h_{lo})^2 \qquad (4)$$

Segments which pass these three tests are ranked based on a sum of differences normalized by covariances. The segment with the smallest difference is selected as a match.

$$Diff = \text{Error!}^2 \cdot \sigma_\theta \text{Error!}^2_{lo} + \sigma_\theta \text{Error!}^2_{gl}) + \frac{(C_{lo} - C_{gl})^2}{\sigma^2_{Clo} + \sigma^2_{Cgl}} + \frac{(x_{gl} - x_{lo})^2 + (y_{gl} - y_{gl})^2}{(h_{gl} + h_{lo})^2}$$

To the segment with the highest probability that has passed the whole comparison test we apply two independent Kalman filters, as described in section 4.1.

3.2 Matching Local Segments against the Global Grid

In the second process for position estimation, the global grid is matched with local segments obtained from the local grid. The process starts with finding line segments in the local grid, continues by calculating the best transformation of the segments into the global grid

and ends by applying the Kalman filter which is described in 4.2.

For matching a local segment directly against the global grid, we produce a mask of the segment. This mask contains the probabilities of the local grid cells lying on this line segment. This mask is transformed into the global grid and correlated with the global grid cells lying under this mask. The value of that correlation increases when the cells are of the same state (both have a probability to be occupied or both have a probability to be free). On the other hand the value decreases when two cells have different states (one cell has a probability to be occupied and the other to be free). The value of that correlation is the basis for the estimation of the quality of the transformation. By searching for the best transformation of the segment mask into the global grid we find the best transformation between the two grids. With the estimated position of the robot we have also an estimation of the transformation between the two grids which we are using to find the best transformation.

The calculation of the covariance $\boldsymbol{C}_T$ is based on the values of the correlations around the best transformation. The distribution of the correlation values are approximated by a Gaussian function. Thus, we can calculate the covariance $\boldsymbol{C}_T$ of the transformation T by analysing the distribution of the correlation values.

3.3 Matching the Local Grid against Global Segments

The principle of the third re-localisation process is the same as of the second re-localisation process which was described in section 3.2. The only difference is that we compare segments the global grid directly to the local grid. If we have found the best transformation of a global segment into the local grid, we apply a Kalman filter, as described below in section 4.2.

3.4 Matching the Local Grid against the Global Grid

The fourth re-localisation process also searches for the transformation of the local grid into the global grid. In this case, we directly use the two occupancy grids. It is possible to correlate the entire local grid with the global grid. However, we have decided to use only a part of the local grid and to find the best transformation of this part into the global grid. One reason for that selection is the data reduction. Perhaps more importantly, we note that the free regions of the environment are regions without information and thus not suitable for the re-localisation process. Thus, with this approach we produce a mask of the local grid, which contains the grid cells with a probability to be occupied above a certain threshold. This mask is used in the same manner as the segment masks to find the best transformation between the two grids as described in section 3.2. After finding the best transformation T and the covariance $\boldsymbol{C}_T$ of this transformation we apply the Kalman filter described in 4.2.

4 Updating the Estimated Position of the Robot

Because the parameters of the global grid are absolute and those of the local grid are relative to the robot, the uncertainty of the local grid relatively to the global grid is the same as the uncertainty of the robot position $\hat{P}_r = (\hat{x}_r, \hat{y}_r, \hat{\theta}_r)^T$. By correcting the position of the local grid, we can correct the robot position. The following section presents the Kalman filters for the first re-localisation process which is based on segments obtained from the two occupancy grids as described in section 3.1. Section 4.2 discusses the Kalman filter for the re-localisation process that were described in sections 3.2, 3.3 and 3.4.

For the updating of the estimated position of the robot we use a Kalman filter [1], [5], [6]. Here we want introduce only the notions of the Kalman filter for the linear and discrete case that will be used in the following:

$X^*(t)$ is the predicted state vector at the instant t. It is based on projecting forward from the estimated state at time $T-\Delta t$. The evolution of the state vector is predicted by applying a process model represented by a matric φ disturbed by a driving term $\hat{U}(t)$ and a unpredictable disturbance $\hat{V}(t)$, to the state vector which was estimated for the time $t-\Delta t$, $\hat{X}(t-\Delta t)$:

$$X^*(t) := \varphi\, \hat{X}(t-\Delta T) + \hat{U}(t-\Delta T) + \hat{V}(t-\Delta T) \quad (5)$$

The driving term permits other processes to contribute information about the evolution of the system. In our system it is set to zero. The process noise, $\hat{V}(t)$, accounts for bias of the process model and can be estimated or set to zero. The covariance of the process noise, $\hat{\boldsymbol{Q}}_x(t)$, provides a means to account for the imprecision of the process model, including higher order derivatives. This shows up in the prediction of the covariance of the state vector:

$$\boldsymbol{C}_x^*(t) := \varphi\, \hat{\boldsymbol{C}}_x(t-\Delta T)\, \varphi^T + \hat{\boldsymbol{Q}}_x(t-\Delta T) \quad (6)$$

The observation process is described by a "sensor model" modelled by a matrix ${}^Y\boldsymbol{H}_X$. This matrix predicts an observation vector, $Y^*(t)$ from the currently predicted state vector $X^*(t)$. The prediction is estimated to be corrupted by a bias, $W(t)$, with uncertainty given by a covariance R(t).

$$Y^*(t) := {}^Y_X\boldsymbol{H}\; X^*(t) + \hat{W}(t) \quad (7)$$

$$\boldsymbol{C}_{Y^*}(t) := {}^Y_X\boldsymbol{H}\;\; \boldsymbol{C}_x^*(t)\;\; {}^Y_X\boldsymbol{H}^T + \hat{\boldsymbol{R}}_x(t) \quad (8)$$

While the estimated bias is often set to zero, the covariance of the bias serves to account for unmodelled errors in the sensor model.

The difference between the observed sensor reading, Y(t) and the predicted sensor reading, $Y^*(t)$ is called the "innovation". This is the information that we obtain by matching the local grid to the global grid. Part of the power of the Kalman filter is that it permits us to recursively update the estimated state with the innovation, providing a weighting to account for the relative imprecisions of the innovation and the predicted state. This weighting is provided by the Kalman Gain matrix:

$$\boldsymbol{K}(t) := \boldsymbol{C}_X^*(t) \; {}_X^Y\boldsymbol{H}^T \left[{}_X^Y\boldsymbol{H} \, \boldsymbol{C}_Y^*(t) + \boldsymbol{C}_Y(t) \right]^{-1} \qquad (9)$$

This gain matrix is then used to update the estimated state and its uncertainty:

$$\hat{X}(t) := X^*(t) + \boldsymbol{K}(t)\,[Y(t) - Y^*(t)] \qquad (10)$$

$$\boldsymbol{C}^{\wedge}(t) := \boldsymbol{C}^*(t) - \boldsymbol{K}(t) \; {}_X^Y\boldsymbol{H} \, \boldsymbol{C}^*(t) \qquad (11)$$

4.1 Kalman Filter for the First Re-localisation Process

The first re-localisation process is based on local and global segments obtained from the two occupancy grids. Each match of a local segment with a global segment provides a constraint on the position of the robot $\hat{P}_r = (\hat{x}_r, \hat{y}_r, \hat{\theta}_r)^T$ and its uncertainty $\mathbf{C}_P$ which may be treated as an innovation. Any one segment contains an ambiguity concerning its end-points. The end-points may be due to an intersection with another surface, or may simply be the limit of the sensed region. To account for this, we model the observation process as measuring the perpendicular distance from the origin for the segment, c. Each segment extracted from the global grid is transformed to the robot centered coordinates frame from the world coordinated frame by a homogeneous coordinate matrix composed from the estimated position of the robot. The center point and orientation are multiplied by this matrix.

$$\begin{bmatrix} x_{lo}^* \\ y_{lo}^* \\ \theta_{lo}^* \end{bmatrix} = \begin{bmatrix} Cos(-\theta_r) & Sin(-\theta_r) & -x_r \\ -Sin(-\theta_r) & Cos(-\theta_r) & -y_r \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{gl} \\ y_{gl} \\ \theta_{gl} \end{bmatrix}$$

Since the innovation comes only from the perpendicular position and orientation, $[c, \theta]^T$, the sensor model must be completed by a transformation defined by the orientation of the segment, θ.

$$Y^* = \begin{bmatrix} c^*(t) \\ \theta^*(t) \end{bmatrix} = \begin{bmatrix} -Cos(\theta) & -Sin(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{lo}^* \\ y_{lo}^* \\ \theta_{lo}^* \end{bmatrix}$$

thus the sensor model is the composition of these two transformations

$${}_X^Y\boldsymbol{H} = \begin{bmatrix} -Cos(\theta) & -Sin(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Cos(-\theta_r) & Sin(-\theta_r) & -x_r \\ -Sin(-\theta_r) & Cos(-\theta_r) & -y_r \\ 0 & 0 & 1 \end{bmatrix}$$

The local model segment is also transformed to a vector of the form $[c, \theta]^T$ by:

$$Y = \begin{bmatrix} c \\ \theta \end{bmatrix} = \begin{bmatrix} -Cos(\theta) & -Sin(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{lo} \\ y_{lo} \\ \theta_{lo} \end{bmatrix}$$

Having identified a correspondence of a pair of segments from the local and global model, the difference between the predicted and observed segments is applied using equations (8) through (11).

4.2 Kalman for the second, third and fourth Re-localisation Process

In sections 3.2, 3.3 and 3.4 we have introduce matching procedures which find the best transformation T of the local grid into the global grid. In this section we discuss the Kalman filter which is based on this transformation T and which will correct the robot position $\hat{P}_r = (\hat{x}_r, \hat{y}_r, \hat{\theta}_r)^T$.

As state vector $\hat{X}(t)$ of this Kalman filter we use the whole robot position $\hat{P}_r$. The transformation T provides directly the observation. Therefore, the sensor model is the identity matrix of dimension three.

$${}_X^Y\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The prediction of the state vector and its uncertainty is the estimated position of the robot provided by the vehicle controller $\hat{X}(t) := \hat{P}_r = (\hat{x}_r, \hat{y}_r, \hat{\theta}_r)^T$, $\boldsymbol{C}^{\wedge}(t) := \boldsymbol{C}^{\wedge}_p$. The local grid is transformed to the global reference using the estimated position so that the innovation is provided by the difference in position and orientation at which the grids are found to provide the best match.

5 Updating the Global Grid

The updating process of the global grid integrates the data of the local grid into the global grid. It is indispensable that the updating process takes into account the uncertainty of the local grid position. Because the local grid is centered on the robot, this uncertainty is the

uncertainty of the robot's estimated position and orientation. We remark that the correction of the estimated position of the robot is very important for the updating process particular during exploration of unknown environment.

The updating process starts with the convolution of the local grid with the uncertainty of the robot position to produce a grid which we call conv-grid. . Finally we apply the following updating rule :

global-grid(x, y) =

$$\begin{cases} \text{conv-grid(x, y)} & \text{if global-grid(x, y) = unknown} \\ \frac{1}{2}(\text{conv-grid(x, y) + global-grid(x, y)}) & \text{otherwise} \end{cases}$$

In this formula global-grid(x, y) is the global grid with the absolute parameters x and y. conv-grid(x; y) is the local grid convoluted with the uncertainty of the robot position and transformed to the absolute parameters.

6 Experimental Results

We have incorporated the perception and the re-localisation procedures described in this paper into the overall system of our mobile robot platform. Figure 2 shows an example of one of our experiments taken with the robot in a hall-way. Figure 2a shows the raw ultrasonic range data projected to external coordinates around the robot. Figure 2b shows the local grid and the edge segments extracted from this grid. Figure 2c shows the robot with its uncertainty in estimated position within the global grid, including the edge segments extracted from the global grid. Figure 2d shows the local grid superimposed on the global grid at the position and orientation of best correspondence.

After the a series of experiments we have found that the most stable position estimation results are obtained by matching segments to segments (first technique) or grids to grids (fourth technique). The results are comparable or more accurate than the ones we obtain with previous work using a parametric model of segments extracted directly from the sensory data [5], with the exception that the grid can also support obstacle avoidance of very small obstacles.

Matching a local grid to a global grid gives a transformation covariance which is relatively large, perpendicular to segments and too small in the direction of segments. Thus when following a wall, this technique can falsely constrain the position uncertainty of the robot, resulting in an inability to match when other features are encountered. For this reason, it is preferable to match segments extracted from the grids and to update the estimated position using an extended Kalman filter.

7 Conclusion

In this paper we have described an investigation into the use of occupancy grids for position estimation for a mobile robot. We have compared four techniques for matching a global occupancy grid to a local occupancy grid. We have derived a simple the Kalman filter equation for the case of directly matching a local grid to a global grid. We have also derived an extended Kalman filter for the case where we match segments extracted from the two grids. Our first experiments show that the most reliable position estimation is obtained by matching segments extracted from the local and global grids and using the match as a two dimensional "innovation" vector with an extended Kalman filter.

Bibliography

[1] M. Babarrere, P. P. Krief, and B. Gimonet. Le Filtrage et ses Applications. Cepaclus Edition, 1978.

[2] D. H. Ballard and C. M. Brown. Computer Vision. Englewood Cliffs, NJ: Prentice Hall, 1982.

[3] J. Borenstein and Y. Kore, "Real-time obstacle avoidance for fast mobile robots in cluttered environments", In IEEE International Conference on Robotics and Automation, pages 572-577, 1990.

[4] J. Borenstein and Y. Koren, "The vector field histogram: fast obstacle avoidance for mobile robots" IEEE Transactions on Robotics and Automation, 7(3): 278-288, June 1991.

[5] J. L. Crowley. "World modelling and position estimation for a mobile robot using ultrasonic ranging". In IEEE International Conference on Robotics and Automation, pages 674-680, 1989.

[6] J. L. Crowley, P. Stelmaszyk, T. Skordas, and P. Puget. Measurement and integration of 3-d structures by tracking edge lines. International Journal of Computer Vision, 8(1): 29-52, 1992.

[7] A. Elfes. Using occupancy grids for mobile robot perception and navigation. IEEE Computer, pages 46-57, June 1989.

[8] H. P. Moravec and A. Elfes. "High resolution maps from angle sonar". In IEEE International Conference on Robotics and Automation, pages 116-121, March 1985.

[9] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. AI Magazine, 9(2): 61-74, 1988.

[10] F. Wallner, T. C. Lueth, and F. Langinieux. "Fast local path planning for a mobile robot", In ICARCV'92 Second International Conference on Automation, Robotics and Computer Vision, September 1992.

[11] Y. Zhang and R. E. Webber. "On combining the hough transform and occupancy grid methods for

detection of moving objects" In IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2155- 2160, July 1992.

[12] J. L. Crowley, "Navigation for an Intelligent Mobile Robot", IEEE Journal on Robotics and Automation, Vol 1 (1), March 1985.

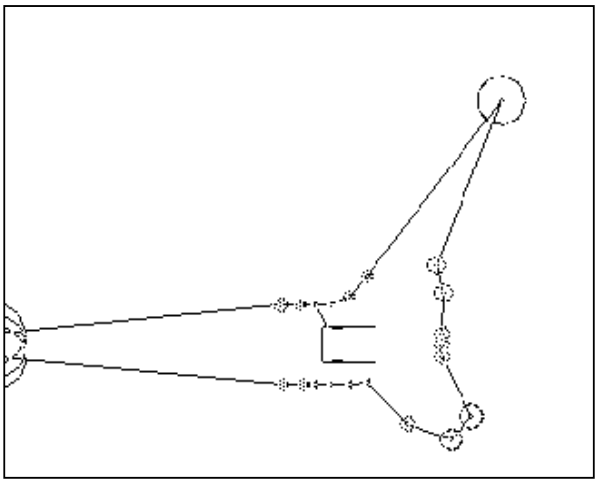

Figure 2a The raw ultrasonic range data projected to external coordinates around the robot.

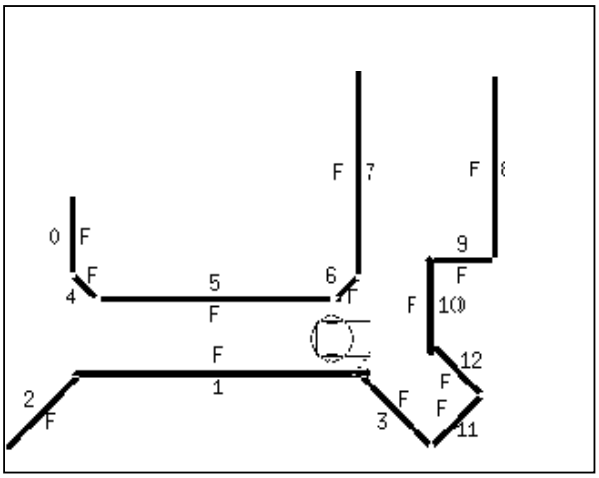

Figure 2c The robot with its uncertainty in estimated position superimposed on the composition of the local grid and the segments extracted from the global grid.

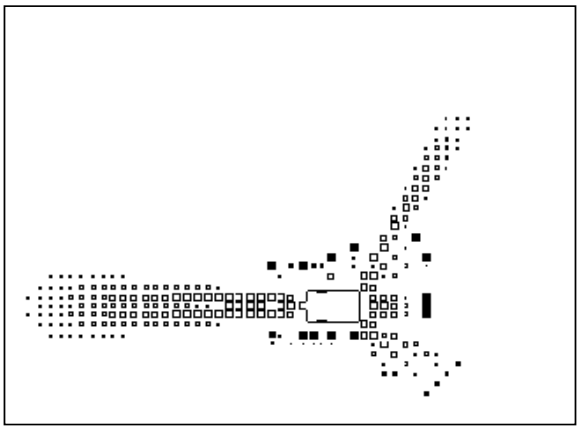

Figure 2b The local grid and the edge segments extracted from this grid.

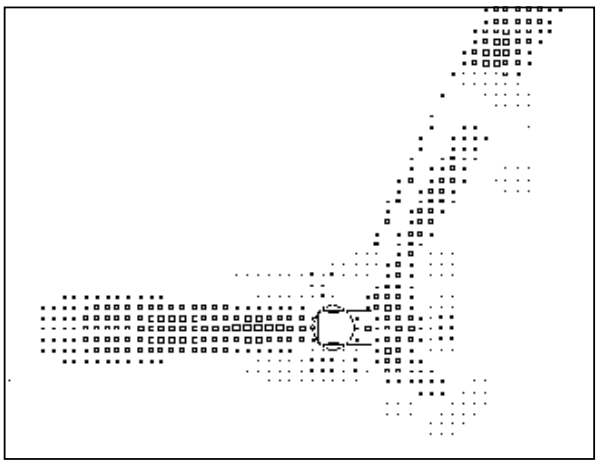

Figure 2d The local grid superimposed on the global grid at the position and orientation of best correspondence.