MOBILE ROBOT LOCALIZATION USING RANGE AND VISION

J. Neira, J. D. Tardós
Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza
c/María de Luna, 3, E-50015 Zaragoza, Spain
email : {jneira, jdtardos}@mcps.unizar.es

J. Horn, G. Schmidt
Technische Universität München, Lehrstuhl für Steuerungs-und Regelungstechnik
D-80290 München, Germany
email: {horn@bull., gs@}lsr.e-technik.tu-muenchen.de

In this paper we present the 2D version of the *Symmetries and Perturbation model* (SPmodel), a probabilistic representation model and an EKF integration mechanism for uncertain geometric information that is suitable for sensor fusion and integration in multisensor systems. We apply the SPmodel to the problem of location estimation in mobile robotics, experimenting with the mobile robot MACROBE. We have chosen two types of sensorial information whose complementary nature allows a mobile robot to localize itself precisely in a known environment: range images and intensity images obtained from a laser sensor. By measuring distance to objects, range basically provides *position* information. On the other hand, the intensity image provided by the laser sensor (equivalent to a monocular vision system) can be used to detect vertical edges, corresponding to corners, frames, etc., which mainly provide *angular* information. Results of these experiments show that fusing simple and computationally inexpensive sensorial information can allow a mobile robot to precisely locate itself. They also demonstrate the generality of the proposed fusion and integration mechanism.

1 Introduction

Mobile robots cannot rely solely on dead-reckoning to determine their location because dead-reckoning errors are cumulative. For this reason, mobile robots must be equipped with external sensors that obtain information from the environment to help the robot determine its location more accurately. There has been considerable work in this direction[1,2,3,4,5]. Most mobile robots rely on using one type of sensor or on sensing one type of feature from the environment. This may be sufficient in many cases, and yet there are situations in which relying on one sensor or one type of feature may be insufficient to precisely locate the robot. For example, the MACROBE robot is equipped with a range sensor capable of obtaining information of the scene in front of the robot that in[5] is used to determine the location of walls in its field of view. The detection of a wall allows the robot to reduce uncertainty in the direction normal to the wall. This has proven to be quite effective in most cases. However, when the sensor only detects walls in front of him, uncertainty is only reduced in the direcion of motion. In this situation, being able to obtain sensor information from a different sensor, or of different nature, would help the robot to locate itself more precisely.

This work has been motivated by these ideas. We show that multisensor fusion in mobile robotics allows to use more simple and inexpensive sensor processing, and at the same time makes the robotic system more robust. For multisensor fusion to be possible, a model that allows to describe geometric information of diverse nature and fuse it appropriately is necessary. Sections 2 and 3 of this paper describe an uncertainty representation and fusion model, the SPmodel [6], that has been developed for this purpose. We make use of it in the experimental framework of the MACROBE project. Information provided by a laser sensor is used both to determine the location of walls in front of the robot, and by processing the intensity image, to determine the location of vertical edges that can correspond to corners, as well as door and window frames. The process of extracting environmental information and fusing it to locate the MACROBE, as well as experimental results are described in section 4. Finally in section 5, the main conclusions derived from experimenting with the SPmodel and the MACROBE are drawn.

2 Uncertain Geometry: the SPmodel

Most of the aspects in multisensor fusion in mobile robots are related to the concept of *location*. One needs to determine whether a point belongs to an edge, whether two edges are collinear, what is the distance between a point and an edge, between two points, etc. Choosing an appropriate representation for the location of geometric entities of diverse nature is fundamental to adequately answer these questions. It must be taken into account that sensors render *uncertain* geometric information. There are two fundamental aspects of geometric uncertainty: *partiallity* and *imprecision*. Partiallity refers to the degrees of freedom (d.o.f.) associated to different geometric entities, and how they determine the location of other entities related to them. Imprecision refers to the accuracy in the estimation of the location of geometric entities.

Usually, a different set of parameters is used to represent the location of different geometric features. For example, one can represent the location of a point in 2D by a vector (x, y), where x and y represent the cartesian coordinates of the point with respect to a base reference. The location of an edge can be represented by a pair (ϱ, θ), where ϱ represents the perpendicular distance of the edge to the origin of the base reference, and θ its orientation with respect to the x axis of this reference. Since the parameters used to represent the location of diverse geometric entities differ, as we consider more geometric entities the determination of geometric relations between them becomes complex. In some representations the same set of parameters is used to express the location of any geometric feature, for example using a triplet (x, y, ϕ). This has the disadvantage that not all geometric features require the same amount of parameters to completely determine their location. This representation, being overparameterized, causes problems to the fusion mechanism (covariance matrices will be singular).

In this work, a new representation model, the SPmodel, is used. It is a probabilistic model to represent uncertain geometric information that avoids these inconveniences. In the following, the 2D version of the SPmodel is presented, and a description of how it deals with both aspects of geometric uncertainty is given.

2.1 Partiallity

Different geometric features have different d.o.f. associated to their location. For example, the location of a robot in 2D is determined by three d.o.f., while the location of a point only by two. Normally, this leads to establishing a different set of parameters to describe the location of different geometric entities, giving rise to the unnecessary complications described above. In the SPmodel, a reference E is associated to the location of any type of geometric feature. The location of this reference with respect to a base reference is given by a transformation t_{WE} composed by two cartesian coordinates and an angle:

$$\mathbf{x}_{WE} = (x,\ y,\ \phi)^T \quad \text{where} \quad t_{WE} = \text{Trans}(x,\ y) \cdot \text{Rot}(z,\ \phi)$$

The composition of two location vectors is denoted by $\oplus$, and the inversion of location vectors, as well as the composition with the inverse are denoted by $\ominus$. Thus, given $\mathbf{x}_{AB} = (x_1,\ y_1,\ \phi_1)^T$ and $\mathbf{x}_{BC} = (x_2,\ y_2,\ \phi_2)^T$, their composition is calculated as follows:

$$\begin{aligned}
\mathbf{x}_{AC} &= \mathbf{x}_{AB} \oplus \mathbf{x}_{BC} \\
&= (x_1 + x_2 \cos\phi_1 - y_2 \sin\phi_1, y_1 + x_2 \sin\phi_1 + y_2 \cos\phi_1, \phi_1 + \phi_2)^T
\end{aligned}$$

Similarly,

$$\ominus\mathbf{x}_{AB} \;=\; (-x_1 \cos\phi_1 - y_1 \sin\phi_1, x_1 \sin\phi_1 - y_1 \cos\phi_1, -\phi_1)^T$$

The d.o.f. that determine the location of a geometric entity are related to its *symmetries of continuous motion.* The symmetries of a geometric entity

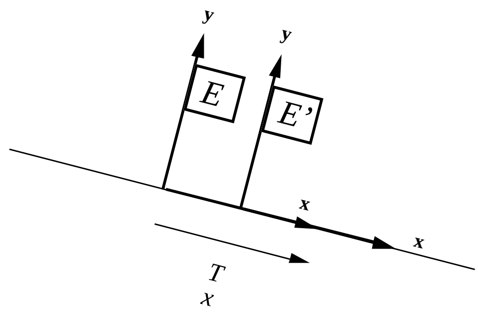

Figure 1: Representation of a 2D edge.

E are defined as the set $\mathcal{S}_E$ of transformations that preserve the element. For example, the symmetries of an infinite edge are the set of continuous translations (T_x) along the edge (fig. 1). We represent the set of symmetries using a *row selection matrix* B_E, denominated *binding matrix of the feature*.

The binding matrix of a geometric entity allows us to express one of the fundamental geometric concepts, *coincidence*. Given two geometric entities of the same type, whose location is represented by A and B respectively, their locations coincide if:

$$B_A \mathbf{x}_{AB} = 0 \tag{1}$$

where B_A denotes the binding matrix of both geometric entities.

Example 2.1: *Determining whether two points coincide*
Consider two points A and B, whose relative location is given by:

$$\mathbf{x}_{AB} = (x, y, \phi)^T \;\; ; \;\; B_A = B_B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

According to eq. (1) we have:

$$B_A \mathbf{x}_{AB} = (x,\, y)^T = 0$$

This result expresses the fact that the two points coincide if the relative position of their associated references is zero, *regardless of what their relative orientation is.*

◇

To express coincidence between different types of geometric elements, we use the *binding matrix of a pairing.* In the case of two geometric entities of different type, whose location is represented by A and B respectively, one of the following equations expresses whether their locations coincide (up to symmetries):

$$\begin{array}{ll} B_{AB}\mathbf{x}_{AB} = 0 & \text{Direct Constraint} \\ B_{BA}\mathbf{x}_{BA} = 0 & \text{Inverse Constraint} \end{array} \tag{2}$$

where B_{AB} and B_{BA} denote the binding matrix of the pairing.

Example 2.2: *Determining whether a point belongs to an edge*
Consider a point P and an edge E, whose relative location is given by:

$$\mathbf{x}_{EP} = (x, y, \phi)^T \ ; \ B_P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \ ; \ B_E = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} ; B_{EP} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$$

In this case eq. (2) gives:

$$B_{EP}\mathbf{x}_{EP} = y = 0$$

That is, the point belongs to the edge if its position in the direction of the y axis of the edge reference equals zero. ◇

2.2 Imprecision

Sensors give imprecise information, thus one only obtains an estimate of the location of a given geometric element. Most classical models of imprecision belong to one of two categories: *set-based* models and *probabilistic* models. For several reasons, we favor the use of probabilistic models[7,8]. From the *appropriateness* point of view, its seems less apparent that sensors give measurements with uniform distributions of error, as set-based models suggest. From the *practicality* point of view, in set-based models the correlation between position and orientation errors is seldom considered and complex to consider. In the SPmodel, the estimate of the location of a given entity E is denoted by $\hat{\mathbf{x}}_{WE}$, and the error associated to this estimate is expressed using a *differential location vector* $\mathbf{d}_E$, relative to the reference associated to the element, so that the true location of E is given by (fig. 2):

$$\mathbf{x}_{WE} = \hat{\mathbf{x}}_{WE} \oplus \mathbf{d}_E \tag{3}$$

Since the d.o.f. of $\mathbf{d}_E$ corresponding to the symmetries of continuous motion contain no location information, we assign 0 to their corresponding

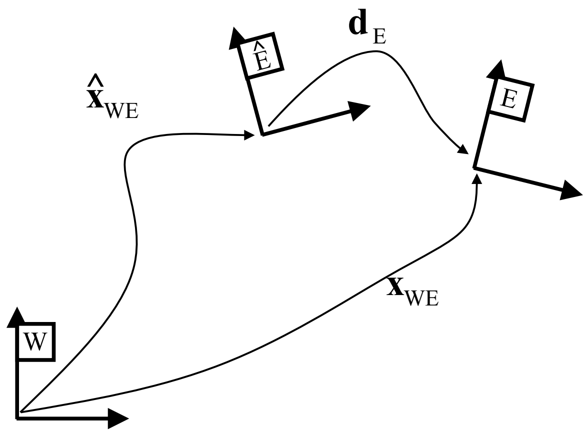

Figure 2: Uncertain location of E in the SPmodel.

values. We call *perturbation vector* a vector $\mathbf{p}_E$ formed by the non-null elements of $\mathbf{d}_E$. These two vectors are related by the binding matrix B_E:

$$\mathbf{d}_E = B_E^T \mathbf{p}_E \quad ; \quad \mathbf{p}_E = B_E \mathbf{d}_E$$

The information associated to the estimated location of a geometric element E is represented by a quadruple $\mathbf{L}_{WE} = (\hat{\mathbf{x}}_{WE}, \hat{\mathbf{p}}_E, C_E, B_E)$, where:

$$\mathbf{x}_{WE} = \hat{\mathbf{x}}_{WE} \oplus B_E^T \mathbf{p}_E \; ; \; \hat{\mathbf{p}}_E = E[\mathbf{p}_E] \; ; \; C_E = Cov(\mathbf{p}_E)$$

Note that the error associated to a location is expressed relative to the feature reference E and not to the base reference W. In this way the value of the covariance is not magnified by the distance of the feature to the base reference. This guarantees that covariance values have a clear interpretation. The use of the binding matrix also makes the representation non-overparameterized. In [8], details on how fundamental operations with the SPmodel are performed can be found.

3 Multisensor Fusion: the SPfilter

In [7], the SPmodel is used to establish a general integration mechanism that allows to obtain a *suboptimal estimation of location* for objects or features from a set of partial and uncertain sensorial observations. The estimation of the location of an object or feature from a set of geometric observations is nonlinear, due to the existence of orientation terms, and can be solved using the *extended Kalman filter* or the *extended information filter*[9,10]. In this paper we use the 2D version of the SPmodel along with the specialized version of the EKF for the SPmodel, the SPfilter, to estimate the location of a mobile robot from a set of partial and imprecise observations of features in the robot's environment.

The extended information filter is formulated as follows: let $\mathbf{x}$ be the state vector whose value is to be estimated, and let there be n independent and possibly partial observations $\mathbf{y}_k$ of $\mathbf{x}$, where $k \in \{1, \ldots, n\}$, affected by white Gaussian noise:

$$\hat{\mathbf{y}}_k = \mathbf{y}_k + \mathbf{u}_k \;\; ; \;\; \mathbf{u}_k \sim N(0, S_k)$$

Let each observation $\mathbf{y}_k$ be related to $\mathbf{x}$ by an implicit nonlinear function of the form $\mathbf{f}_k(\mathbf{x}, \mathbf{y}_k) = 0$. We use a first order approximation of $\mathbf{f}_k$:

$$\mathbf{f}_k(\mathbf{x}, \mathbf{y}_k) \simeq \mathbf{h}_k + H_k(\mathbf{x} - \hat{\mathbf{x}}) + G_k(\mathbf{y}_k - \hat{\mathbf{y}}_k)$$

where:

$$\mathbf{h}_k = \mathbf{f}_k(\hat{\mathbf{x}}, \hat{\mathbf{y}}_k) \;\; ; \;\; H_k = \left.\frac{\partial \mathbf{f}_k}{\partial \mathbf{x}}\right|_{(\hat{\mathbf{x}}, \hat{\mathbf{y}}_k)} \;\; ; \;\; G_k = \left.\frac{\partial \mathbf{f}_k}{\partial \mathbf{y}}\right|_{(\hat{\mathbf{x}}, \hat{\mathbf{y}}_k)} \tag{4}$$

The estimate $\hat{\mathbf{x}}_n$ of the state vector and its covariance P_n after integrating the n measurements are:

$$\hat{\mathbf{x}}_n = P_n M_n \;\; ; \;\; P_n^{-1} = \sum_{k=1}^{n} F_k \;\; ; \;\; M_n = -\sum_{k=1}^{n} N_k \tag{5}$$

where:

$$F_k = H_k^T (G_k S_k G_k^T)^{-1} H_k \;\; ; \;\; N_k = H_k^T (G_k S_k G_k^T)^{-1} \mathbf{h}_k \tag{6}$$

This is the nonrecursive formulation of the information filter, which is equivalent to a least squares estimation in batch mode: integrating a block of n measurements at the same time. Formulations of the recursive information filter and Kalman filter can be found in [7]. This scheme is applicable to a wide range of estimation problems; in this work we apply it to three different estimation problems in mobile robotics: the estimation of the location of a wall from laser points, the estimation of the location of the robot from wall observations, and the estimation of the robot location from intensity image edge observations. Next, the third problem is described.

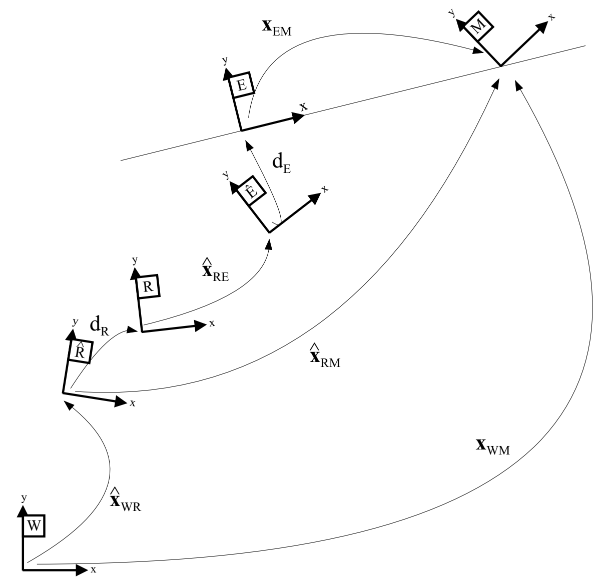

Figure 3: References involved in the integration of a monocular edge to the estimation of the robot location.

3.1 Estimating the location of a mobile robot from monocular 2D edges

Let $\mathbf{L}_{WR} = (\hat{\mathbf{x}}_{WR}, \hat{\mathbf{p}}_R, C_R, I_3)$ be the estimated location of a mobile robot. Let $\mathbf{x}_{WM}$ represent the location of a vertical edge (a 2D point) according to the map. Let $\mathbf{L}_{RE} = (\hat{\mathbf{x}}_{RE}, \hat{\mathbf{p}}_E, C_E, B_E)$ be the estimated location of the edge according to the monocular camera (fig. 3). Using only one image, our observation of the point is a projection line between the point and the camera. Thus, in this case we have:

$$B_M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \;\; ; \;\; B_E = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \;\; ; \;\; B_{EM} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$$

Assuming that edge E corresponds to model point M (fig. 3), we use it to improve the estimation of the robot location by means of the *inverse constraint*:

$$\mathbf{f}_k(\mathbf{x}, \mathbf{y}_k) = B_{EM}\mathbf{x}_{EM} = B_{EM}(\ominus \mathbf{d}_E \ominus \hat{\mathbf{x}}_{RE} \ominus \mathbf{d}_R \oplus \hat{\mathbf{x}}_{WR}) = 0$$

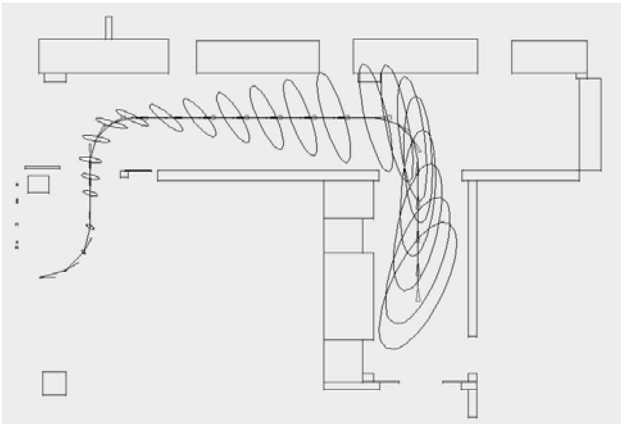

Figure 4: MACROBE path according to odometry. Uncertainty is magnified 100 times.

In this case the state to be estimated is represented by the perturbation vector of R, and the measurement by the perturbation vector of E:

$$\mathbf{x} = \mathbf{d}_R \;\; ; \;\; \mathbf{y}_k = \mathbf{d}_E \;\; ; \;\; \mathbf{f}_k(\mathbf{x}, \mathbf{y}_k) = B_{EM}(\ominus \mathbf{y}_k \ominus \hat{\mathbf{x}}_{RE} \ominus \mathbf{x} \oplus \hat{\mathbf{x}}_{WR}) = 0$$

4 Experiments

We have used the mobile robot MACROBE to experiment with the SPmodel and the SPfilter. In the experimental setup, MACROBE is programmed to follow a path through a narrow corridor (fig. 4). At 24 different steps of the robot path, the laser sensor takes an image, a 41×321 array of environmental points. In [5], this information is processed to segment walls out of these points. The locations of the sensed walls are compared to those of a map, and using an exact solution, the position of the robot is determined. Results of subsequent single-image localization and dead-reckoning data are fused using an EKF. In this work we fuse both the range information and the intensity information as is detailed next.

4.1 Fusing range information

The range information given by the laser sensor is fused to estimate the robot location in a numerically equivalent way as it is done in [5]. Each point is

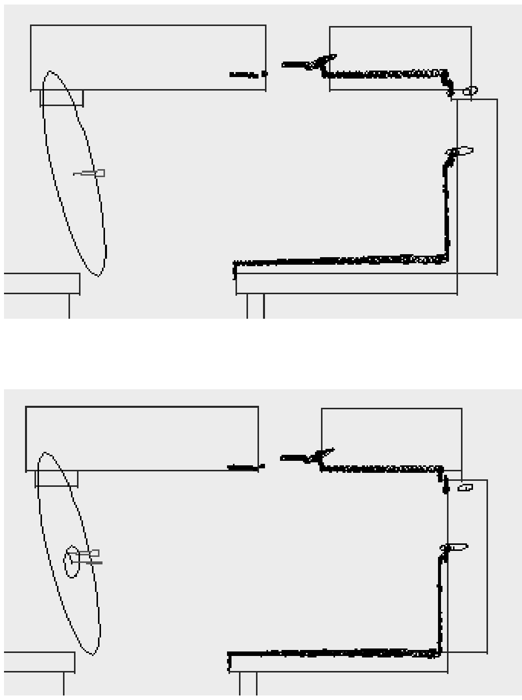

Figure 5: Range observations before and after integration.

associated to a map wall (if not considered spurious) and its contribution to the robot location estimation is calculated using the SPfilter. Figure 5 shows the observed points at a robot location (upper image), and their predicted location after reestimating the robot location (lower image). Note the change in robot position, orientation, as well as the reduction of uncertainty.

4.2 Fusing intensity information

The laser sensor provides not only the distance to each pixel of the array but also gives intensity information of each point. In this work we also process the intensity image of the laser sensor to extract a set of vertical edges that can correspond to corners, and also to door or window frames. This information is contrasted with the *a priori* map of the environment, composed of 2D segments corresponding to walls and 2D points corresponding to vertical edges. The

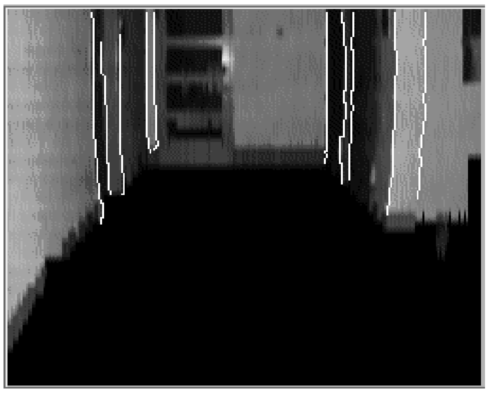

Figure 6: Vertical edges extrated from intensity image.

nearest model feature to each observation is chosen as candidate for pairing, and in a first step, all observations having only one candidate whose location can be considered compatible are fused in the estimation of the robot location (we perform a hypothesis test on their relative location). This process limits the possibilities of accepting an incorrect match, and it is repeated until none of the remaining observations has a suitable candidate.

The quality of the intensity image is rather low (fig. 6). There is little contrast and considerable distortion, especially on the lower part of the image. We use the VISTA software for edge extraction[11]. The contrast and brightness of the laser image is adjusted, and the image is scaled to limit the effect of distortion. The resulting image is processed using Canny's edge extractor [12], and vertical edges are selected.

Figure 7 shows the projection of the observed vertical edges at a robot location (upper image), and their predicted location after reestimating the robot location (lower image). In general, uncertainty reduction is less apparent than in the case of integrating range points because in this case we integrate less observations to estimate the robot location.

4.3 Fusing both range and intensity

Fig. 8 shows the results of this process in the last five steps of the path. The image on the left hand side shows the path and uncertainty evolution relying only on dead-reckoning. The image in the center, left, shows the results when

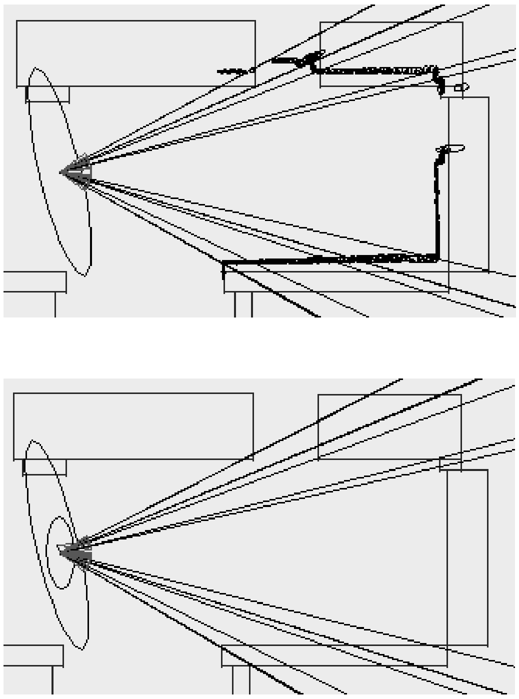

Figure 7: Intensity observations before and after integration.

fusing only the laser wall observations. In this case, only the walls in front of the robot are sensed. We can see that the uncertainty in the location of the robot is only significantly reduced in the direction of motion because the front wall cannot contribute information in the normal direction. The central image, right, shows the path resulting from fusing only the vertical edge observations. In this case, uncertainty reduction occurs in both directions, but since the observation of an edge with a monocular camera only gives information on one d.o.f. (the projection line), this reduction is less significant than in the case of laser walls. Finally, the image on the right hand side shows the path resulting from integrating both laser wall observations and the intensity image vertical edges. It can be seen that the combination of both types of sensor observations leads to a more precise estimation of the location of the robot in all directions.

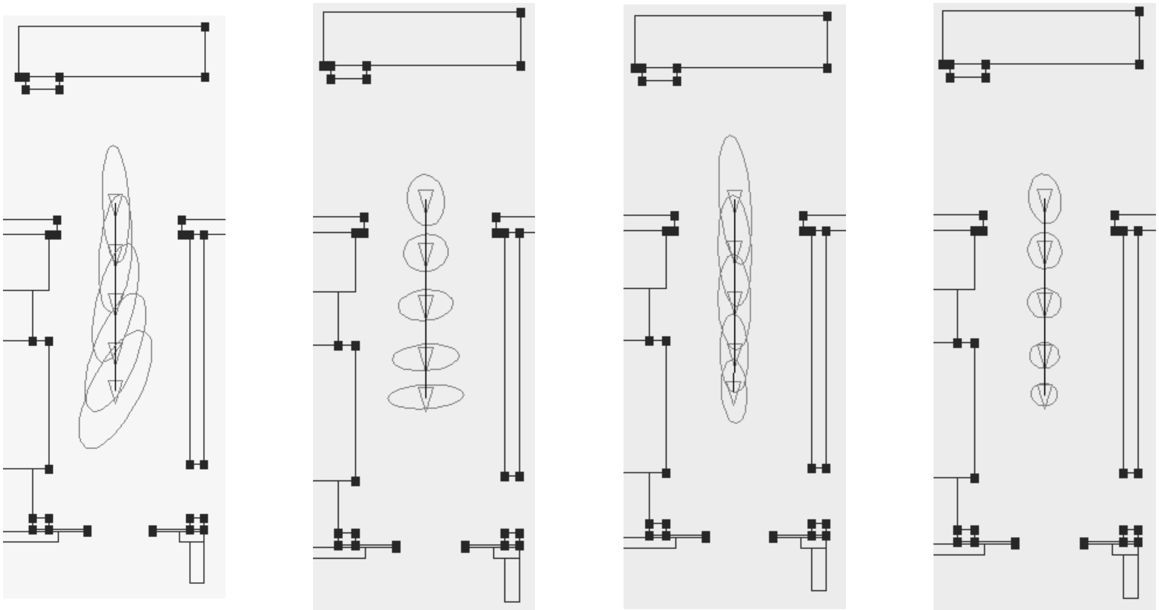

Figure 8: Robot path for four different fusion experiments: *left*, uncertainty evolution of dead-reckoning; *center, left*, integrating 2D walls; *center, right*, integrating 2D vertical edges; *right*, integrating both 2D walls and 2D vertical edges. Uncertainty is magnified 100 times.

5 Conclusions

In this work a model for the representation and fusion of uncertain geometric information is described. It is suitable to be used for location estimation problems in multisensor mobile robots. To prove this, we have applied it to the problem of precisely locating the MACROBE robot by fusing observations of vertical walls as well as vertical edges obtained from a laser sensor mounted on the robot. The results show that this representation and fusion model allows to precisely locate the robot through simple and inexpensive sensor processing.

At present, vision processing is carried out with general vision research software. Our next step is to apply specialized vision processing techniques to allow real-time localization. Furthermore, experiments have been limited to the localization of the robot by comparing all obtained observations with an *a priori* map. Observations that seem not to correspond to any features on the map are simply discarded. For the mobile robot to be confronted to less structured environments, it is necessary that new information is not discarded but confirmed over time and introduced in the map. This map building and updating capability will be the subject of future work.

Acknowledgments

This work was supported in part by spanish CICYT project TAP94-0390, Deutsche Forschungsgemeinschaft project SFB 331, and European Robotics Network ERNET contract CHRX-CT93-038156.

References

1. I. Cox. Blanche – An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle. *IEEE Transactions on Robotics and Automation*, 7:193–204, 1991.
2. J. Leonard and H. Durrant-Whyte. Mobile Robot Localization by Tracking Geometric Beacons. *IEEE Transactions on Robotics and Automation*, 7:376–382, 1991.
3. J. Evans, B. Krishnamurthy, B. Barrows, T. Skewis, and V. Lumelsky. Handling Real–World Motion Planning: A Hospital Transport Robot. *IEEE Control Systems Magazine*, 12:15–20, 1992.
4. A. Kosaka and A. Kak. Fast Vision–Guided Mobile Robot Navigation using Model–Based Reasoning and Prediction of Uncertainties. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, volume 12, pages 2177–2186, Raleigh, North Carolina, 1992.
5. J. Horn and G. Schmidt. Continuous Localization of a Mobile Robot Based on 3D–Laser–Range–Data, Predicted Sensor Images, and Dead–Reckoning. *Robotics and Autonomous Systems, Special Issue "Research on Autonomous Mobile Robots"*, 14:99–118, 1995.
6. J.D. Tardós. Representing partial and uncertain sensorial information using the theory of symmetries. In *IEEE International Conference on Robotics and Automation*, pages 1799–1804, Nice, France, May 1992.
7. J.D. Tardós. *Integración Multisensorial para Reconocimiento y Localización de Objetos en Robótica.* PhD thesis, Dpto. de Ingeniería Eléctrica e Informática, University of Zaragoza, Spain, Febrero 1991.
8. J. Neira. *Geometric Object Recognition in Multisensor Systems.* PhD thesis, Dpto. de Ingeniería Eléctrica e Informática, University of Zaragoza, Spain, September 1993.
9. A. Gelb, editor. *Applied Optimal Estimation.* The M.I.T. Press, Cambridge, Mass., 1974.

10. C.A. Bozzo. *Le Filtrage Optimal et ses Applications aux Problèmes de Poursuite*. Technique et Documentation, Montrouge, France, 1983.
11. A. R. Pope and D. G. Lowe. Vista: A software environment for computer vision research. In *CVPR*, 1994.
12. J.F. Canny. Finding edges and lines in images. Tech Report AI-TR-720, MIT Artificial Intelligence Laboratory, Cambridge, Massachussetts, 1983.