Journal of Intelligent and Robotic Systems **22:** 129–141, 1998.

Multisensor Fusion: An Autonomous Mobile Robot

F. MATÍA and A. JIMÉNEZ
DISAM – Universidad Politécnica de Madrid, José Gutiérrez Abascal 2, E-28006 Madrid, Spain; e-mail: matia@disam.upm.es

(Received: 5 February 1997; accepted in final form: 10 November 1997)

Abstract. A conventional autonomous mobile robot is introduced. The main idea is the integration of many conventional and sophisticated sensor fusion techniques, introduced by several authors in recent years. We show the actual possibility of integrating all these techniques together, rather than analyzing implementation details. The topics of multisensor fusion, observation integration and sensor coordination are widely used throuhout the article. The final goal is to demonstrate the validity of both mathematical and artificial intelligence techniques in guaranteeing vehicle survival in a dynamic environment, while the robot carries out a specific task. We review conventional techniques for the management of uncertainty while we describe an implementation of a mobile robot which combines on-line heterogeneous sensors in its navigation and localisation tasks.

Key words: integration, sensor fusion, map building, localisation, artificial vision, reactive navigation, mobile robotics, autonomous systems.

1. Introduction

During the last decade, the use of intelligent control has increased in robotics. Fuzzy logic, neural networks, genetic algorithms, and expert systems have been used in control and planning tasks, especially in the field of mobile robotics. Nowadays the main goal is to build autonomous systems. But uncertainty about a robot's location and information from the environment makes it very difficult to achieve complete autonomy in real applications, without auxiliary techniques that also have a sound theoretical foundation. The full integration of both kind of techniques, artificial intelligence (AI) and mathematics, is not easy.

A mobile robot may be considered as an intelligent autonomous system (IAS) in the sense that:

- The complete navigation system resides on an on-board computer, and the vehicle is completely wireless (autonomy).
- The robot has some kind of reasoning capability which allows it to make its own decisions, and to appropriately select, fuse and integrate heterogeneous sensor data (intelligence).

The main task of our robot is to reach a goal, following a path. But the intelligence of such an IAS could be reduced to planning and control if uncertainty were not present in the sensors or in the environment. If we restrict planning to the decomposition of a mission into elemental tasks, or to the planning of a path between two points, given a map of the environment, there is no uncertainty to take into account. On the other hand, the control system calculates the appropiate velocity commands to follow the reference path, and uncertainty is present in the odometric and navigation sensors.

Without precise sensor data, control objectives will never be reached. To cope with uncertainty modelling in mobile robotics, well-known state estimation (localisation) and navigation algorithms exist (Kalman filter,probability theory, Dempster–Shafer theory, decision theory, fuzzy logic, etc.).

The contribution of this work isthe fusion of heterogeneous sensors in a concrete mobile robot (Matia et al. [6]) which uses all its sensorial capability to reduce uncertainty while it navigates, avoiding obstacles, integrating new observations to refresh environment maps, and coordinating vision systems to improve location estimation. Advanced robot control techniques are complemented with AI in the control module, using fuzzy logic and/or neural networks.

2. Integration of Sensors and Problem Solving Tasks

The following list represents the use of each sensor in our mobile platform:

1. Sonars:
   - Occupancy grid maps;
   - Reactive control.
2. Color vision:
   - Localisation.
3. Active vision:
   - Geometric maps;
   - Reactive control;
   - Localisation.

Sonars are used for map building and reactive navigation. A color camera is used for vehicle localisation, and a b/w camera coupled with an infrared laser is used for map building, reactive navigation and localisation.

Furthermore, the algorithms used in the three previous tasks (map building, navigation and localisation) depend on the kind of sensor we use. For example, occupancy grid maps differ from geometric maps in both their building algorithms and their use. Localization with the color camera is static (the vehicle is completly stopped), while localisation with the b/w camera is dynamic. Finally, several navigation controllers use sonars, while others use both sonars and the active vision system. In this other list we show how several planning algorithms are needed to integrate several tasks which use different kind of sensors:

1. Path planning:
   - Occupancy grid maps;
   - Geometric maps.

2. Navigation planning:
   – Reactive control with sonars;
   – Reactive control with active vision.
3. Localisation planning:
   – State estimation with color vision;
   – State estimation with active vivion.

3. The Mobile Robot and its Environment

Figure 1 shows a mobile robot. The mobile platform is a Robuter vehicle from the French company Robosoft. The perception system is composed of: (i) a ring of 24 sonars used for reactive navigation and cell map generation, (ii) an infrared laser diode and an infrared camera situated in front of the robot, used for control, localisation and geometric map building, and (iii) a CCD color camera with pan and tilt movements situated on the top of the robot, used for localisation.

The environment is a 2D room with known fixed obstacles, fixed beacons with known 3D localisation, and possible unknown obstacles.

3.1. ENVIRONMENT MODELS

Two different environment models are used: an occupancy grid and a geometric model. In the first one, the 2D environment is divided into a grid with an occupancy value: 0 for empty and 1 for occupied. A variant of this model is quadtrees, in which the initial space is recursively divided (if necessary) into four equal parts until all objects fill a cell.

In a 2D geometric model, each object is represented by a set of segments (Ayache and Faugeras [1]). Each line is represented by two parameters: a distance d and an angle α. The path planning module of our mobile robot uses this geometric model with two different algorithms: Voronoi diagrams and visibility graphs. The planner decides when to use each algorithm. While visibility graphs always generate the shortest path, Voronoi diagrams give better results in environments with high obstacle density.

3.2. SENSOR MODELS

The information given by each sensor is modeled to compare the real measurements with the estimates. This will allow later improvement of sensor information. Since all sensor model parameters are more or less estimated, a probabilistic feature must be added to the sensor model to represent the certainty of the measure (Durrant-White [4]).

The **odometric model** corresponds with the kinematic model of the mobile platform:

$$\mathbf{x}(k+1) = \mathbf{f}\big(k, \mathbf{x}(k), \mathbf{u}(k)\big) + \mathbf{v}(k), \text{ where } \mathbf{x}^{\mathrm{T}}(k) = \big[x(k), y(k), \theta(k)\big]$$

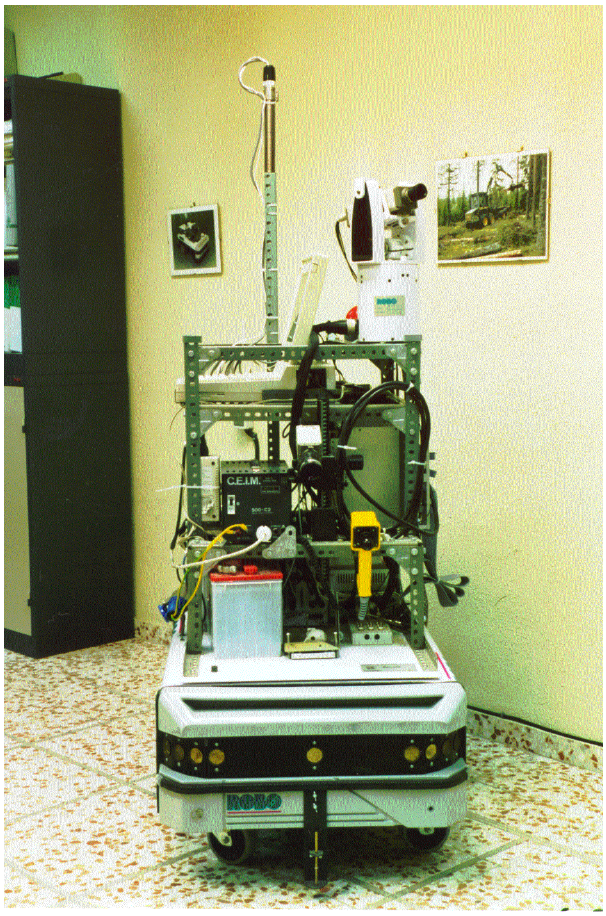

Figure 1. The mobile platform.

is the robot location (oriented position) at instant k, $\mathbf{f}$ is the kinematic model, $\mathbf{u}(k)$ is the velocity command vector, and $\mathbf{v}(k)$ is the model noise vector (uncertainty is included here).

Sonars are usually distributed along a ring (see Figure 2). The **sonars** model follows:

$$p\big(z(k) = z \mid l\big) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(z-l)^2}{2\sigma^2}\right), \tag{1}$$

where p is the probability density function (so uncertainty is also considered), $z(k)$ is the sensor measure, l is the distance of the closest object and σ is a parameter to be determined empirically. In the case of the occupancy grid model, uncertainty is refreshed using Bayes' theorem (Elfes [5]):

$$P\big(s_i(k+1) \mid z(k+1)\big) = \frac{p(z(k+1) \mid s_i(k))P(s_i(k) \mid z(k))}{\sum_{s_i} p(z(k+1) \mid s_i(k))P(s_i(k) \mid z(k))}, \tag{2}$$

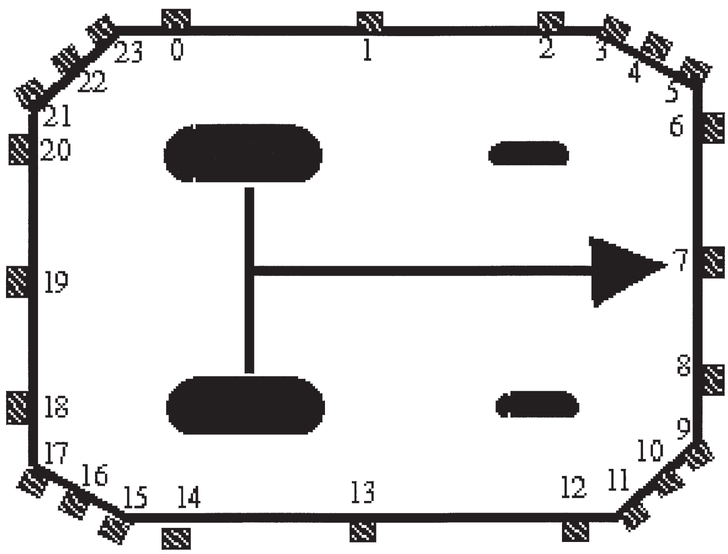

Figure 2. Sonar ring.

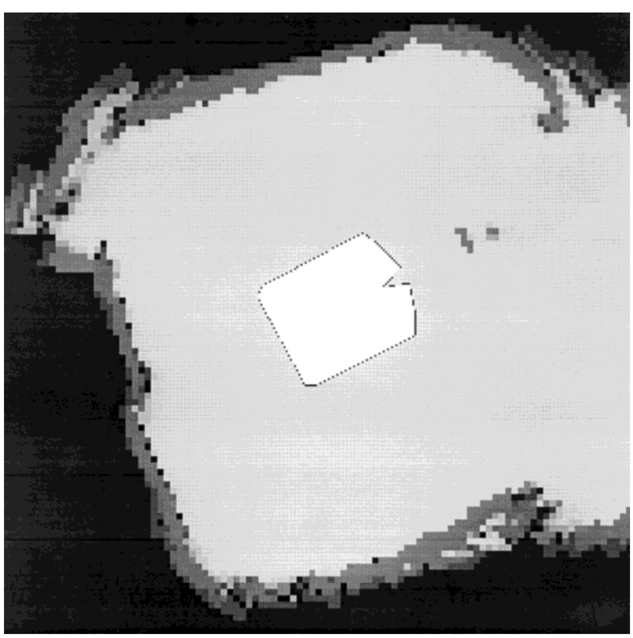

Figure 3. Probabilistic map.

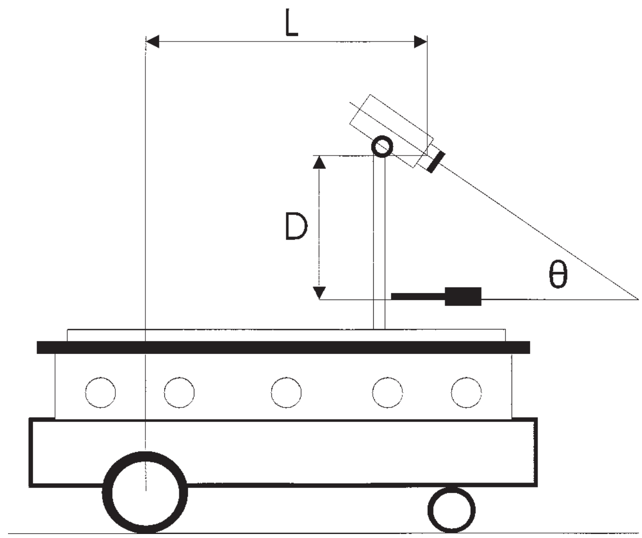

Figure 4. Camera-infrared laser. Side view.

where $s_i(k)$ is the state (occupied or free) of cell i at instant k and P states for probability. An example of a sonar map, using a 24 sonar ring, is shown in Figure 3.

The set **camera-infrared laser** supplies a set of infrared light points in the image frame $\mathbf{x_I}(k)$, which may be transformed into the robot coordinate frame $\mathbf{x_R}(k)$. The noiseless equations follow:

$$x_R(k) = x_I(k) + D\frac{1 + (y_I(k)/((1 + \mathrm{g}r^2(k))f))\tan\theta}{\tan\theta - y_I(k)/((1 + \mathrm{g}r^2(k)))f} + L_X, \tag{3}$$

$$y_R(k) = y_I(k) + \frac{x_R(k)x_I(k)}{(1 + \mathrm{g}r^2(k))f\cos\theta} + L_Y, \tag{4}$$

where L_X and L_Y are the horizontal coordinates of the camara, f is the focal distance, D is its height, θ is the camera angle, g is the image distortion factor, and $r^2(k) = x_I^2(k) + y_I^2(k)$ (see Figure 4).

The robot coordinate frame must be transformed later into the origin coordinate frame $\mathbf{x_O}(k)$. Again the noiseless equation follows:

$$\mathbf{x_O}(k) = \begin{bmatrix} x(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} \cos\theta(k) & \sin\theta(k) \\ -\sin\theta(k) & \cos\theta(k) \end{bmatrix} \mathbf{x_R}(k). \tag{5}$$

From all the light points, a segment extraction is carried out (see Figure 5), so a set of measured segments (d, α) is obtained from the model $\mathbf{z}(k) = \mathbf{h}(k, \mathbf{x}(k)) + \mathbf{w}(k)$, where $\mathbf{z}^{\mathrm{T}}(k) = [d(k), \alpha(k)]$ is the segment position at instant k, $\mathbf{h}$ is the sensor model, $\mathbf{x}(k)$ is the robot location, and $\mathbf{w}(k)$ is the model noise vector (measurement uncertainty).

The **CCD color camera** is used to measure the 3D position $\mathbf{x_B}(k)$ of several artificial beacons. Again the beacon corrdinates are transformed into the origin

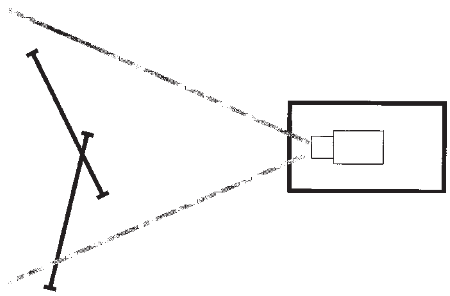

Figure 5. Camera-infrared laser. Upper view.

coordinate frame through an equation similar to (5). The sensor model also has the form $\mathbf{z}(k) = \mathbf{h}(k, \mathbf{x}(k)) + \mathbf{w}(k)$. Special attention must be paid to the fact that the three perception systems must be previously calibrated to minimize uncertainty in the model parameters. Sonars may be calibrated for large or small ranges, and camera parameters must be estimated to make proper use of the equations shown in this section.

4. State Estimation

The control module must keep an accurate estimate of the mobile robot's location at each moment. This state estimate $\hat{\mathbf{x}}(k)$ is obtained using the extended Kalman filter. The algorithm works by integrating all sensor readings into a more precise measure which allows the robot's state to be predicted.
Three levels of state estimation are available:

- The odometry system, which gives an initial but unaccurate estimate of the location based on the last position and the encoders' readings.
- The set camera-infrared laser, which uses a map of the environment and compares it with the light point measurements.
- The CCD color camera, which uses the known locations of several artificial beacons and compares them with the information obtained from the actual image.

The incomming sensor readings are integrated (Crowley [3]) as follows. While the robot is moving, following a path or avoiding an obstacle, the incoming state from the odometry system $\hat{\mathbf{x}}(k \mid k)$ is improved on-line by integrating the laser measurements. The extended Kalman filter has four steps in each cycle:

1. **Prediction** of new measurements:

$$\hat{\mathbf{x}}(k+1 \mid k) = \mathbf{f}\big(k, \hat{\mathbf{x}}(k \mid k), \mathbf{u}(k)\big),$$

$$P(k+1 \mid k) = \mathbf{F}_{\hat{\mathbf{x}}}(k) P(k \mid k) \mathbf{F}_{\hat{\mathbf{x}}}^{\mathrm{T}}(k),$$

$$\hat{\mathbf{z}}(k+1 \mid k) = \mathbf{h}\big(k+1, \hat{\mathbf{x}}(k+1 \mid k)\big),$$

where $\mathbf{F}_{\hat{\mathbf{x}}}$ is the Jacobian matrix of $\mathbf{f}$, and P is the covariance matrix of $\hat{\mathbf{x}}$.

2. **Observation** of new measurements: $\mathbf{z}(k+1)$.
3. **Matching** of predicted and observed measurements:

$$\nu(k+1) = \mathbf{z}(k+1) - \mathbf{z}(k+1 \mid k),$$
$$\mathbf{S}(k+1) = \mathbf{H_x}(k+1)P(k+1 \mid k)\mathbf{H_x^T}(k+1),$$

where ν is the measurement innovation and $\mathbf{S}$ is its covariance matrix.

4. **State estimation**:

$$\mathbf{W}(k+1) = P(k+1 \mid k)\mathbf{H_x^T}\mathbf{S}^{-1}(k+1),$$
$$\hat{\mathbf{x}}(k+1 \mid k+1) = \hat{\mathbf{x}}(k+1 \mid k) + \mathbf{W}(k+1)\nu(k+1),$$
$$\mathbf{P}(k+1 \mid k+1) = \mathbf{P}(k+1 \mid k) - \mathbf{W}(k+1)\mathbf{S}(k+1)\mathbf{W}^{\mathrm{T}}(k+1),$$

where $\mathbf{W}(k+1)$ is the Kalman filter gain.

In the case of the set camera-infrared laser, the estimated measurements $\hat{\mathbf{z}}(k+1 \mid k)$ are the segments from the static known map, while the real measurements $\mathbf{z}(k+1)$ are the segments extracted from the real world with the laser. If the difference between both, $\nu(k+1)$, is less than a fixed value (applying the Mahalanobis distance), we can use the Kalman filter to obtain the new position $\hat{\mathbf{x}}(k+1 \mid k+1)$. If not, this means that a new object has been detected in front of the vehicle, so we may use probabilistic techniques (Bayesian) to add these new segments to the map (see Section 6).

In the case of the color camera, artificial vision techniques are used. From time to time (when the uncertainty $\mathbf{P}(k \mid k)$ is high enough) the vehicle searches for landmarks, with known coordinates in the 3D environment. These beacons may be artificial (see Figure 6) or natural (e.g., a window).

The color camera looks towards them, extracts from the image the desired features and measures their position $\mathbf{z}(k+1)$. As in the previous case, by comparing this position with the model $\hat{\mathbf{z}}(k+1 \mid k)$, the estimate of the mobile robot's location is improved.

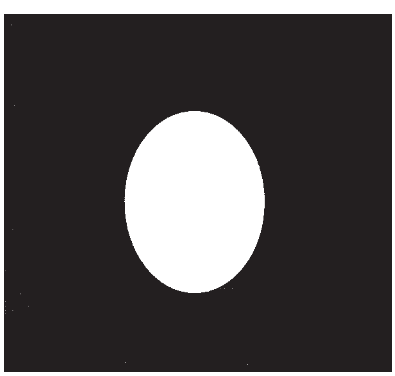

Figure 6. Artificial beacons.

5. Multi-sensor Fusion in Navigation

An intelligent control module resides in the mobile robot for safe navigation. Its main goal is to follow a path with real time obstacle avoidance. Two control paradigms are available to the control supervisor: fuzzy logic for static obstacles and neural networks for multirobot systems. We will discuss the first case, since multirobot systems are not a topic of this article.

Our mobile robot uses the reactive architecture AFREB: Adaptive Fusion of Reactive Behaviors (Moreno et al. [7]). This is composed of two levels:

- The lowest one includes elemental controllers which model primitive behaviours (Brooks [2]) as *follow a path*, *follow obstacle contour* (left or right), and *turn* (left or right).
- A fuzzy decision module which fuses the primitive behaviours originating an emergent behaviour (following a path with obstacle avoidance).

Figure 7 shows the reactive control scheme. The decision module generates a weight $a_i(k)$ for each primitive behaviour i, so they may be fused as follows:

$$\mathbf{u}(k) = \frac{\sum a_i(k)\mathbf{u}_i(k)}{\sum a_i(k)}, \tag{6}$$

where $\mathbf{u}_i(k)$ is the velocity command of the primitive behaviour i, and $\mathbf{u}(k)$ is the final velocity command for the mobile robot. The fusion supervisor must determine the most adequate value for the weights a_i. The fusion rules are as follows:

IF the minimum distance IS medium AND the sensor IS on the right side
THEN the weight of the *follow path* behaviour IS medium
AND the weight of the *right contour following* behaviour IS medium.

In this case, 24 sonars are fused into the regions left, left front, front, right front and right. A second fuzzy decision module which implements heterogeneous sensor fusion of sonar and laser measurements is also available. The rules try to model the following reasoning:

IF the front space IS wide enough (laser)
THEN center the robot (sonars).

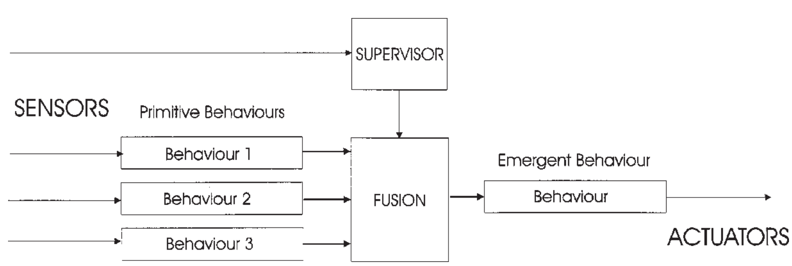

Figure 7. Reactive control architecture.

6. Map Building and Sensor Planning

In Section 3 we suggested the possibility of updating probability grids by integrating sonar information. Now we are going to discuss briefly how to integrate new objects into a geometric map (Zhang and Faugeras [8]).

In Section 4 we said that only new laser measurements which are inside Mahalanobis distance are to be fused into the Kalman filter. But the remaining segments may be used for map building. The steps for map construction on the fly follow:

1. Fusion of light points into segments.
2. Fusion of similar segments.
3. Cut of extra large segments.
4. Dynamic refreshment of segment certainty.
5. Elimination of segments with very low certainty.
6. Fusion of segments with very high certainty into objects.

Figure 8 shows some of the previous steps. On the other hand, examples of sensor planning and coordination may be found in the case of two mobile robots which synchronize their sonars when their position is close, to avoid interferences. But our mobile robot uses sensor planning for the coordination of multiple sensors (sonars and cameras). The two methods for location estimation are to be coordinated:

- The use of structured light.
- The use of artificial vision.

While the robot is moving, a continuous localisation is being carried out with the laser. When the sensor planner detects an excessive increase in the uncertainty

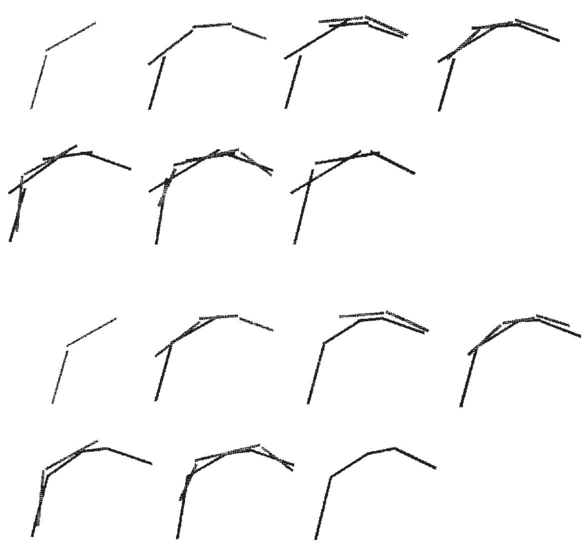

Figure 8. Segment building.

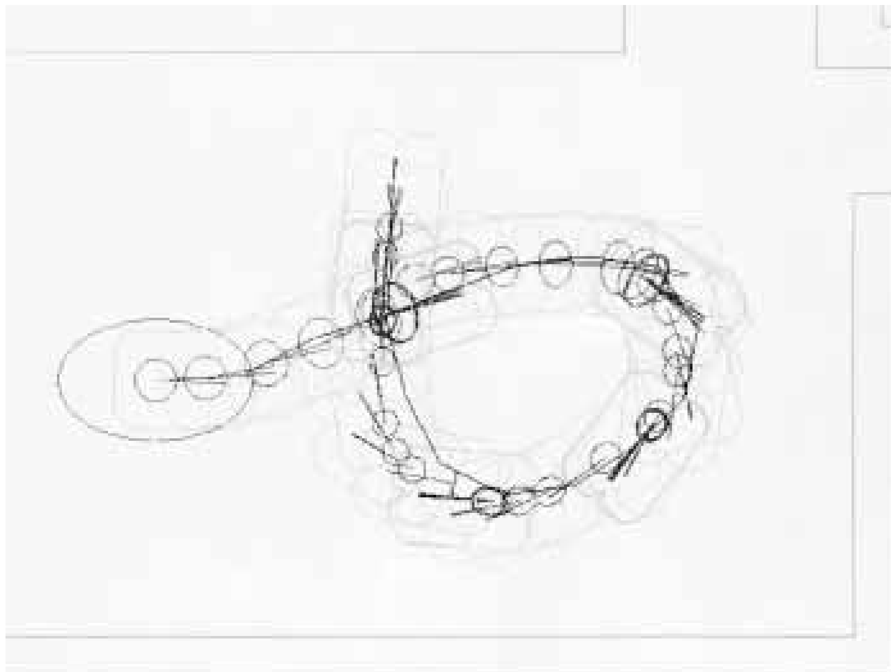

Figure 9. Sensor coordination for map building and localisation.

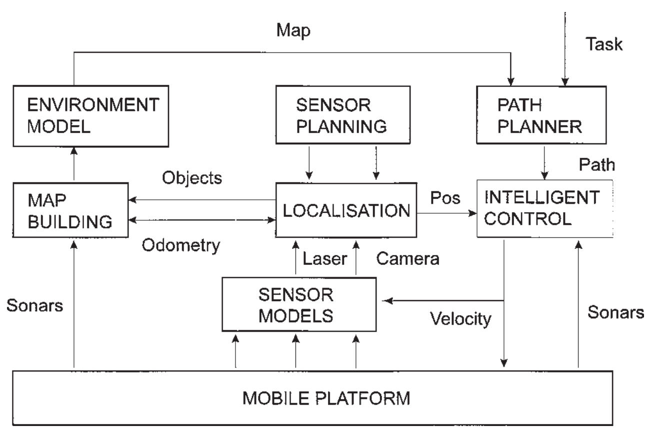

Figure 10. Complete sensor planning and control scheme.

(perhaps no segments are near the front of the robot), it stops the robot and takes advantage of the localisation with the color camera. After uncertainty has decreased, the planner allows the robot to recover its previous movement.

Most of the work inside the planner was to synchronize the only image processing board available, to use both cameras on-line. Figure 9 shows the robot following a path (the elipse represents the position uncertainty) while it uses the laser for localisation and the sonars for obstacle avoidance. From time to time, the localisation with the color camera forces a decrease in the uncertainty. At the same time, new obstacles are being added to the initial map.

The results are more impressive with both localisation methods integrated, rather than using them separately. Figure 10 shows the complete scheme of the control and planning system of the mobile robot.

7. Conclusions

We have demonstrated a mobile robot which takes account of heterogeneous sensor information, and fuses it to achieve a control task. Several features follow:

- Reactive control for path following with obstacle avoidance by means of multisensor fusion.
- State estimation by fusion of odometric and camera observations.
- Dynamic building of environment geometric models, as well as integration of new objects.
- Planning and coordination of continuous and discrete localisation systems.

With these results in mind, we conclude that the mathematical methods described above, appropiately combined with AI techniques, and the fusion of multiple sensor data, allow a conventional IAS to cope with the existent uncertainty in its sensors and in the surounding environment.

Acknowledgements

We thank the engineers Cedric Dassy and Pedro Romero for their work in preparing the demonstration. We specially acknowlewdge the previous experiments in sensor fusion and localisation of Arturo de la Escalera and José María Armingol.

We also acknowledge funding by the Spanish government through the CICYT project TAP96-0600 EVS (*Virtual Platform for Distributed Autonomous Systems Engineering*) and by the European Community through the MOBINET project ERB 4061 PL95-1089 (*Training and Mobility of Researchers Program*).

References

1. Ayache, N. and Faugeras, O. D.: Maintaining representation of the environment of a mobile robot, *IEEE Trans. Robotics Automat.* **5**(6) (1989), 804–819.
2. Brooks, R. A.: A robust layered control system for a mobile robot, *IEEE J. Robotics Automat.* **2**(1) (1986), 14–23.
3. Crowley, J. L.: Mathematical foundations of navigation and perception for an autonomous mobile robot, in: *Reasoning with Uncertainty in Robotics*, Springer, Berlin, 1996, pp. 9–51.
4. Durrant-White, H. F.: *Integration, Coordination and Control of Multi-sensor Robot Systems*, Kluwer Academic Publishers, Dordrecht, 1988.
5. Elfes, A.: Sonar-based real-world mapping and navigation, in: *Autonomous Robot Vehicles*, Springer, Berlin, 1990.

6. Matía, F., Moraleda, E., Jiménez, A., and Puente, E. A.: Autonomous systems and the management of uncertainty, in: *Proc. Fuzzy Logic and the Management of Complexity (FLAMOC)*, Sidney, Australia, 1996, pp. 61–65.
7. Moreno, L., Moraleda, E., Salichs, M. A., Pimentel, J. R., and Escalera, A.: Fuzzy supervisor for behavioral control of autonomous systems, in: *Proc. IECON'93*, Maui, Hawaii, USA, 1993, pp. 258–261.
8. Zhang, Z. and Faugeras, O.: A 3D world model builder with a mobile robot, *Internat. J. Robotics Res.* **11**(4) (1992), 269–285.