Avoiding Obstacles Using a Connectionist Network *

A. Silva† P. Menezes, J. Dias
Departamento de Engenharia Electrotécnica
Instituto de Sistemas e Robótica
Pinhal de Marrocos, 3030 Coimbra, Portugal
{arlindo, paulo, jorge}@isr.uc.pt

Abstract

In this article, visual data obtained by a binocular active vision system is integrated, together with ultrasonic range measurements, in the development of a obstacle detection and avoidance system based on a connectionist grid. The traditional notion of probabilistic occupation grid is extended through the use of a three-layer structure of connectionist networks which allows the integration of several sensorial modalities (in this case ultrasonic sensor readings and stereo vision information) in a probabilistic environment representation. The connectionist nature of the network also allows us to deal with obstacle avoidance by using a mechanism similar to potential field over a discrete set of the robot's configuration space with each grid node representing a possible configuration. The value in each grid node gives us a measure of the configuration occupancy probability and can also be used to guide the robot to a predefined goal configuration simulating a simple gradient descending technique. Finally we present experimental results obtained with the implementation of the above method in a mobile platform which also provides the support for the sensing devices described throughout the article.

1 Problem domain

In this article, visual data obtained by a binocular active vision system is integrated, together with ultrasonic range measurements, in the development of an obstacle detection and avoidance system based on a connectionist grid. This grid based framework allows the integration of two different sensing modalities (vision and sonar), in a way that incoming sensorial data can be mutually enhanced and validated. This approach assumes that, for a mobile robot moving in a cluttered and noisy environment, where obstacle presence and location are unknown, several sensing modalities can be cooperatively used to obtain a more realistic perception of the universe around the robot.

This article proposes a connectionist structure for the integration of different sensing modalities which is an extension of the occupancy grids first described by [11]. The same extension deals with obstacle avoidance by building, in real-time, a potential field over the occupancy grid [9, 6, 7].

Each grid node maps a configuration in a discrete subset of the robot's configuration space. Detected obstacles result in sets of restricted configurations. Using the grid data, a potential field is interactively computed for each mapped configuration. This computation is done in real-time, during the robot's motion, so the potential field changes as new data is integrated into the grid. Sensorial integration is supported by a set of operations which allow incoming sensorial data to be mutually enhanced an validated. These operations are described in the following sections. This structure is the base of a sensorial framework used to implement innovative sensing and control strategies, using the integrated sensorial information. These control strategies are based on a connectionist network which, using the information from the occupancy grid, allows the planning of the robot's motion in a reactive way. The motion is planned in the sense that the computed potential field can be used to search a path to the goal (or a local minimum), and is reactive since information recently gathered will change the potential field and, consequently, the path [1, 9, 8, 7].

A feed-forward artificial neural network is used to navigate the robot through its environment, avoiding restricted configurations corresponding to unexpected obstacles.

All our experimental work is being done around a mobile platform, which provides the test-bed for our experiments. This mobile platform is also the support for the sensing devices described throughout the article.

*This work was partially financed by NATO Scientific Affairs Division on the framework of the NATO Science for Stability Program, PO-ROBOT project.

†Working under grant PRAXIS XXI BM/7255/95 from JNICT-Junta Nacional de Investigação Científica e Tecnológica

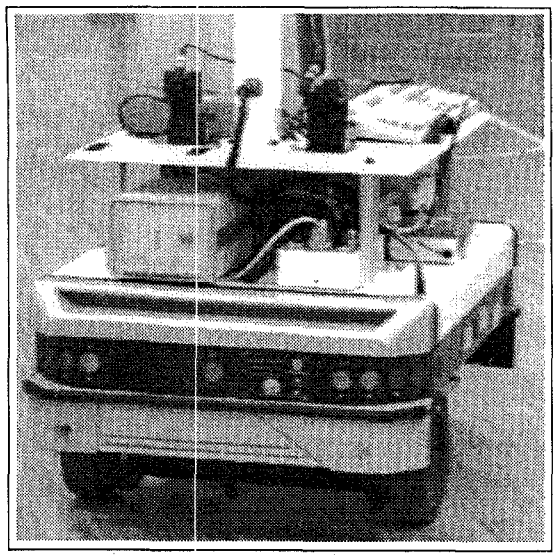

Figure 1: The mobile platform with different sensors: active vision system, sonar sensors, odometry and inertial sensors.

2 The framework to register and combine sensorial information

Suppose a mobile robot (or agent) moving in a Euclidean 2D space W, called the robot's workspace. Information about the presence, position or form of any obstacles in the workplace is unknown to the agent.

If we attach a fixed frame F_W to the workspace W and introduce a robot centered frame F_A, we can define the robot configuration q' as being the position (x, y) and the orientation θ of F_A in relation to F_W or $q' = (x, y, \theta)$. C is the set of all possible robot configurations and is called the configuration space. If we discretize the axes of any frame defined in C, we get a discrete set of configurations D, with each configuration $q = (k_x\delta_x, k_y\delta_y, k_\theta\delta_\theta)$ with δ_x, δ_y and δ_θ representing each axe discretization step and k_x, k_y and k_θ belonging to $\mathcal{Z}$.

We can define an occupation grid G in the configuration space C as a subspace of D, with every configuration $q = (i, j)$. Every configuration in an occupation grid G has associated an occupation probability $p(q)$. A configuration with a $p(q)$ above a certain threshold is said to be occupied, i.e. the robot cannot assume that configuration due to the presence of an obstacle.

To use several modalities of sensorial information in a cooperative way, a framework capable of enabling the effective integration of incoming sensorial flows in an unified representation, must be defined.

Our approach to the development of such a framework is based on a three level grid based representation of the robot's configuration space. In all of this levels, the notion of occupancy grid as a subspace of the discrete configuration space plays a central roll.

- **First level: Memory -** The first level is a global occupation grid G (figure 2) of configurations, which slices the the robot's C configuration space by discretization of F_W. This way, each configuration in G maps a square of the robot's work space W. To each configuration q is associated an emptiness factor $v(q)$, which indicates the probability of that configuration not being occupied by an obstacle, and an occupancy factor $p(q)$, indicating the probability of q being occupied by an obstacle. G is used as a long term memory for the robot, as it will be seen.

- **Second level: Agent's Grid -** The second level is a local grid of configurations L (figure 2), centered on the robot's referential F_A. Each configuration in L also has $v(q)$ and $p(q)$ values attached to it. This values will be changed by incoming sensorial information and used to compute the occupation state $\Phi(q)$ of each configuration. $\Phi(q)$ is calculated iteratively, with the use of a connectionist structure similar to a cellular automata. This is explained in more detail in section 4. The final result is similar to a potential field computed over L, which can be used to allow the robot to avoid obstacles by following the negative gradient of the field.

  The grid G is used as a long term memory, by updating it with the $v(q)$ and $p(q)$ values of the configurations of L as the robot moves. If odometry is used to keep track of the robot's displacement, this operation is affected by odometry errors, which increase over time. To avoid these problems, a time mark is stored together with the values of $v(q)$ and $p(q)$.

- **Third level: Sensor Grid -** The last level is where sensor modeling and data integration is done. For each sensor S two functions are defined. $P(s, q)$ returns the occupancy value in configuration q belonging to a subset of L where the readings of S are mapped, given a reading s from the sensor. $V(s, q)$ returns the emptiness value in configuration q belonging to a subset of L where the readings of S are mapped, given a reading s from the sensor. The values $p(q)$ and $v(q)$ are integrated in L by adding the new values to the current ones. In the following section we present examples for two sensorial modalities: range data from ultrasonic sensors and obstacle detection obtained by a binocular active vision system.

The interaction between the three levels can be summarized as follows: as the robot moves new data is acquired by its sensors. This readings are mapped by

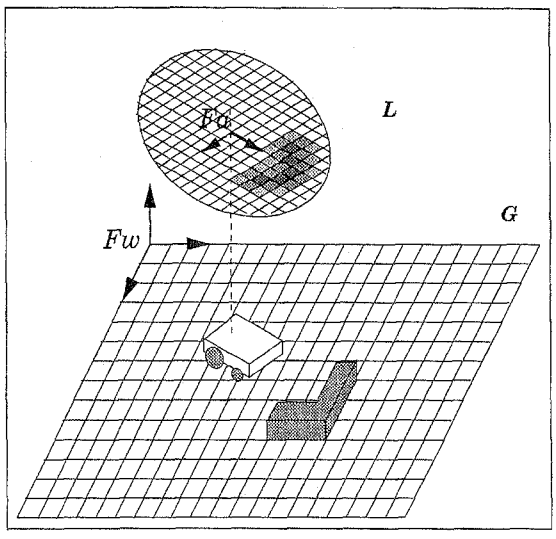

Figure 2: Global and local configuration grids.

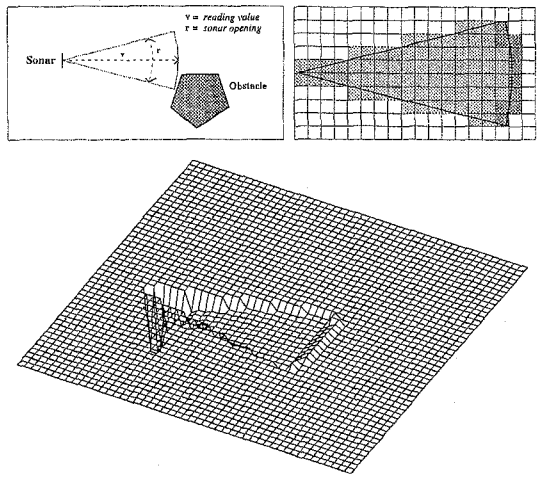

Figure 3: Model for an ultrasonic sensor and equivalent grid representation for a set of readings.

the third level occupancy functions in the relevant subsets of L, and the new $p(q)$ and $v(q)$ values are fused with existing ones. The values of $p(q)$ and $v(q)$ are also stored in G, for later use, if necessary, and labeled with a time stamp.

The movement of the robot also makes some configurations in the local grid L to loose their information, as if the obstacles had moved away. These configurations get their $p(q)$ and $v(q)$ (if they exist) from the nearest configuration in G. The time elapsed since the values were saved is used to simulate the decrease of confidence on the sensorial information, due to obstacle displacement or in result of odometry errors accumulation.

3 Sensing modalities

Besides odometry, two main sensing modalities are used in the process of environment mapping. Those are ultrasonic range data and information acquired through a binocular active vision system.

3.1 Ultrasonic sensors range measurements

The range measurements are obtained from a ring of 24 ultrasonic sensors present in our mobile platform. This kind of sensors, being cheap and easy to use, frequently present erroneous data, caused by multiple reflections and sonar scattering. However, this information can be validated through the use of multiple readings [4], or by integrating other modalities of sensorial information.

For the ultrasonic sensors a simple model is used: a reading is represented by a pie-slice where the arc points have the same probability of belonging to an obstacle, being that probability inversely proportional to the distance measurement. This will give the values of $P(q)$ for the configurations intersected by the arc. $P(q)$ will be 0 in every other configuration.

A point inside the pie-slice is considered to have a probability of not belonging to an obstacle, with that probability being the same in module as if a reading with its cone passing in the point had been obtained by the sensor in the same position. This will give the $V(q)$ values for the configurations inside the pie-slice. $V(q)$ will be 0 in every other configuration. An illustration of this model is summarized in figure 3.

Using this model we can speed up the updating if this grid-based representation with very good experimental results and avoiding the usual gaussian representations - see figure 7.

3.2 Obstacle detection by binocular disparity

Vision (figure 4) is a powerful sensing modality which can be used, not only in low level tasks, like the ones of obstacle detection and avoidance presented here, but also in higher level problems, e.g. object identification and simulation of visual behaviors [2, 3, 5]. In this article we describe a technique to complement and validate range information obtained by ultrasonic sensors, by using stereoscopic vision data. To do this, we developed a simple algorithm for obstacle detection, using stereoscopic information. Its simplicity makes it useful for mobile robotics, where the need for real time implementation asks for algorithms with moderated computational demands.

The solution found allows us to detect obstacles in the field of view of the stereoscopic setup, both above and below the robot's motion plane. The algorithm is based on the assumption: the images from the ground in front of the robot, are from a plane parallel to the

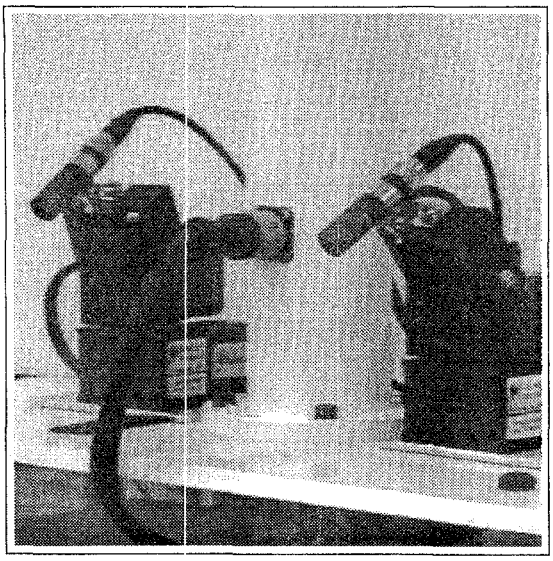

Figure 4: Detailed view of the stereoscopic setup.

robot's trajectory. Since each point on ground lies in the same plane, we can establish an affine transformation which relates correspondent points between the two images. For any image pair with points not in the ground plane, the transformation will fail. This means that any obstacle in the field of view, above or bellow the ground plane, will result in a disparity region when the two images are matched using the transformation mentioned above. When the robot is moving, any relevant disparities between image pairs are identified as an obstacle.

The mathematics underlying this principle are very simple. A real world point with (X, Y, Z) coordinates is related to its image representation, with coordinates (u, v), by a 3×4 transformation matrix A'as expressed by:

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} a'_{1,1} & a'_{1,2} & a'_{1,3} & a'_{1,4} \\ a'_{2,1} & a'_{2,2} & a'_{2,3} & a'_{2,4} \\ a'_{3,1} & a'_{3,2} & a'_{3,3} & a'_{3,4} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

From our particular setup we have that $Z = \gamma$, since we want all points to lie in the ground plane. Expression 1 can be simplified into 2:

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = A_{3\times3} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (2)$$

Since we have two images from the same scene, one from the right camera and another from the left camera, we can establish the relation between the two:

$$A_r^{-1} \begin{bmatrix} s_r u_r \\ s_r v_r \\ s_r \end{bmatrix} = A_l^{-1} \begin{bmatrix} s_l u_l \\ s_l v_l \\ s_l \end{bmatrix} = \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (3)$$

And between matching points in the left and right image:

$$\begin{bmatrix} s_r u_r \\ s_r v_r \\ s_r \end{bmatrix} = A_r A_l^{-1} \begin{bmatrix} s_l u_l \\ s_l v_l \\ s_l \end{bmatrix} = F_{3\times3} \begin{bmatrix} s_l u_l \\ s_l v_l \\ s_l \end{bmatrix} = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \alpha_{1,3} \\ \alpha_{2,1} & \alpha_{2,2} & \alpha_{2,3} \\ \alpha_{3,1} & \alpha_{3,2} & \alpha_{3,3} \end{bmatrix} \begin{bmatrix} s_l u_l \\ s_l v_l \\ s_l \end{bmatrix} \quad (4)$$

Ground plane calibration - Relation 4 defines, up to a scale factor, the relation between points in the left and right image. To obtain this relation, given by matrix F, we must do a calibration step which we call *ground plane calibration*. If make $\alpha_{3,3} = 1$ in equation 4 and manipulate the equations we can obtain 5:

$$\begin{bmatrix} u_l & 0 \\ v_l & 0 \\ 1 & 0 \\ 0 & u_l \\ 0 & v_l \\ 0 & 1 \\ -u_l u_r & -u_l v_r \\ -v_l u_r & -v_l v_r \end{bmatrix}^T \begin{bmatrix} \alpha_{1,1} \\ \alpha_{1,2} \\ \alpha_{1,3} \\ \alpha_{2,1} \\ \alpha_{2,2} \\ \alpha_{2,3} \\ \alpha_{3,1} \\ \alpha_{3,2} \end{bmatrix} = \begin{bmatrix} u_r \\ v_r \end{bmatrix} \quad (5)$$

This relation is a set of linear equations relating the parameters $\alpha_{i,j}$ with the coordinates of correspondent points in the left and right images. Since we have more unknowns than equations, we need a set of correspondent point pairs, at least 8, in order to make possible the computation of all $\alpha_{i,j}$.

To obtain better results, it is advisable the use of a larger set of points and a minimization criterion, such as the least-squares criterion. Equation 5 can then be solved by using the Single Value Decomposition (SVD) method.

Results obtained using the described method are presented in figure 5.

Obstacle registration - To make the obtained results useful for integration with the ultrasonic readings, a further step must be taken: a simple calibration is done in the $Z = \gamma$ plane, relating points with image coordinates (u, v) with points with (X, Y, γ) coordinates in the real world. Again SVD is used to

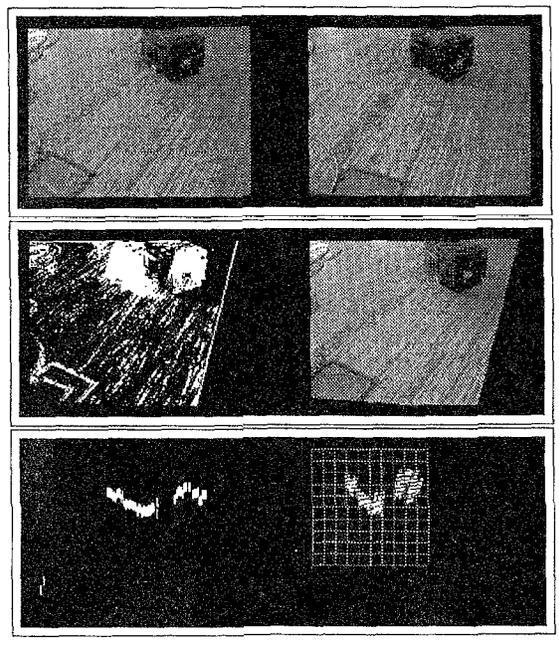

Figure 5: From stereo images to range information in the $Z = \gamma$ plane.

guarantee the calibration robustness. After this calibration step, we can compute the distance from the robot to a detected obstacle by finding the coordinates in the $Z = \gamma$ plane of image points where disparity starts to be significant. All the process of going from stereo images to range information, including disparity images, is illustrated in figure 5. The first image pair is a simple stereoscopic view of the ground plane in front of the robot. An obstacle (box) can be seen in both pictures. The left picture in the second pair shows the discrepancy between both images obtained through image transformation after ground plane calibration. Points not in the ground plane are shown in white and there is also some noise present. In the last image pair it can be seen, in the left picture, the obstacle ground boundaries after the noise was removed, and, in the right picture the boundaries are registered in a cellular grid, using the calibration information obtained in the obstacle registration phase.

$P(q)$ and $V(q)$ for this kind of sensor are defined with base on the image's number of pixels. We can make $P(q)$ proportional to the number of pixels with high disparity in the area of the observation window mapped by q. Configurations where no significant disparity was found have $P(q) = 0$. $V(q)$ is constant (with a value representing our belief in the sensor) in configurations with no significant disparity and is 0 in any configuration where any pixel with high disparity is found.

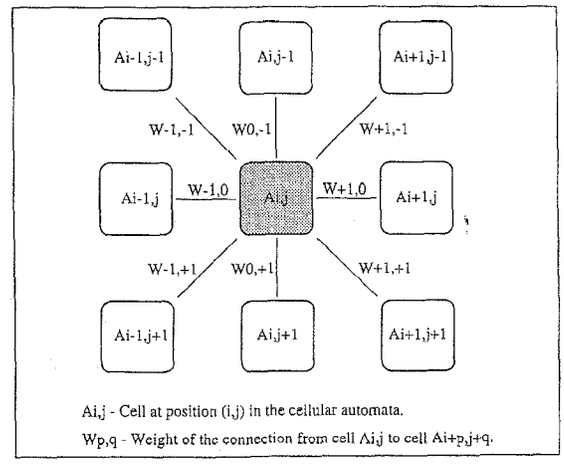

Figure 6: Connections between cells in our cellular automata.

4 Computing the occupation state

We now have $p(q)$ and $v(q)$ values for each configuration in L, being constantly updated with the readings coming from the sensors. How can we use this values to compute an occupation state for each configuration, true enough to allow safe obstacle avoidance? This is done with the computation of a potential field over L. A connectionist structure, similar to a cellular automata, is defined, with each cell representing a configuration in L. This structure can be pictured as a cellular lattice, where each cell is connected to its nearest neighbors (figure 6) and has an output value $\Phi(q,t)$ which represents the occupation state of q at instant t. The cells' values are updated iteratively, and depend only on the values $p(q)$ and $v(q)$ of each cell (configuration) and its neighbors, and the connections between them, accordingly with the automata dynamic rules:

$$\begin{cases} \Phi(q_{i,j},0) &= 0 \\ \Phi(q_{i,j},t) &= m \quad \text{if} p(q_{i,j}) \leq v(q_{i,j}) \\ \Phi(q_{i,j},t) &= P_{i,j} \quad \text{if} p(q_{i,j}) > v(q_{i,j}) \end{cases} \tag{6}$$

$$m = \text{MAX} \left\{ \begin{array}{l} \alpha^{\sqrt{2}} \times \Phi(q_{i+1,j-1},t-1), \\ \alpha^{\sqrt{2}} \times \Phi(q_{i-1,j-1},t-1), \\ \alpha^{\sqrt{2}} \times \Phi(q_{i+1,j+1},t-1), \\ \alpha^{\sqrt{2}} \times \Phi(q_{i-1,j+1},t-1), \\ \alpha \times \Phi(q_{i+1,j},t-1), \\ \alpha \times \Phi(q_{i-1,j},t-1), \\ \alpha \times \Phi(q_{i,j+1},t-1), \\ \alpha \times \Phi(q_{i,j-1},t-1) \end{array} \right\} \tag{7}$$

with α being a parameter chosen to control the obstacles' radius of influence and $\Phi(q_{i,j},t)$ the occupancy value of cell $A_{i,j}$, which maps configuration $q_{i,j}$ at time instant t. As $p(q_{i,j})$ and $v(q_{i,j})$ are modified by

incoming sensorial information, $\Phi(q_{i,j}, t)$ will depend not only on its neighbors values but also on the information about the environment provided by the sensing devices.

Taking it across all the area mapped in L, this is equivalent to the result of an on-line computation of the repulsive part of a potential field [6, 9], rising in the vicinity of obstacles and decreasing away from them. Besides the repulsive field induced by obstacle presence, the potential field is also affected by the presence of a attractive field generated by the existence of a predefined goal configuration. The potential field U at robot's configuration q at time t is then

$$U(q,t) = U_{att}(q,t) + U_{rep}(q,t) \tag{8}$$

or

$$U(q,t) = \Phi(q,t) + \epsilon dist(q,g) \tag{9}$$

with $dist(q,g)$ the euclidean distance between q and goal configuration g and ϵ a weighting factor.

Avoiding locations where the representation presents high occupancy values, the robot will avoid colliding with obstacles. Descending the potential field gradient the robot will head towards the goal. The result is a very flexible structure, that enables us to integrate in the same 2-D low level representation, information obtained from various sensing devices. The dynamics of the structure itself performs data validation between different sources and accumulated readings. Figure 7 presents $\Phi(q)$ results for a room used during experiments, with two boxes in the middle.

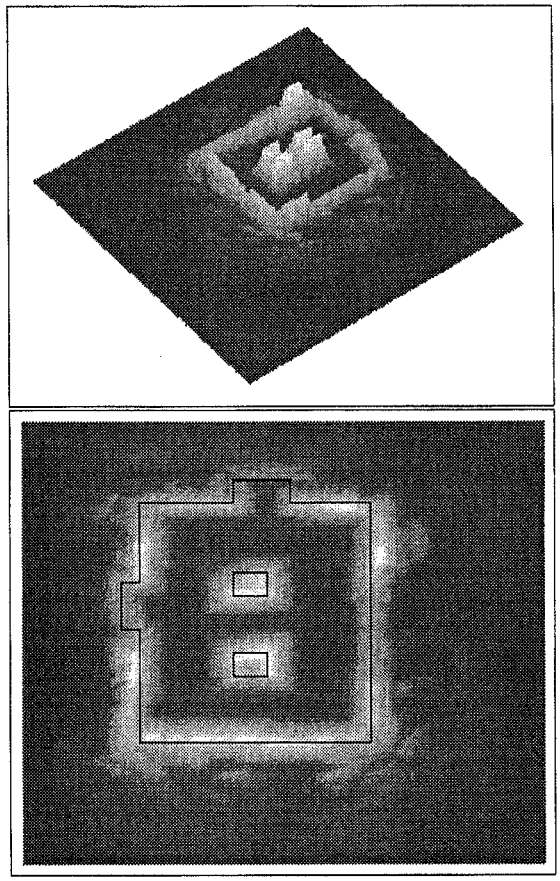

Figure 7: Mapping an experimental room.

5 Controlling the robot

Two main approaches were used to control the robot. In the first one, as suggested above, gradient descending of the potential field U was simulated, in order to navigate the robot to its goal.

In the second approach a three layer feed-forward artificial neural network was trained to navigate the robot, as information about the environment is being gathered and processed in the grid, using that information to safely avoid obstacles and arrive at preseted goals [1].

This neural network type has been often used [8, 9, 7] in mobile robotics to enable the robots to reproduce simple, but intelligent, behaviors. We used it to teach the robot simple heuristics, enabling it to navigate through its environment avoiding the obstacles in its way. It only used the $\Phi(q)$ information and had the present configuration q and the goal configuration g fed to its inputs.

In figure 8 we can see the environmental model in construction while the robot is being navigated by the neural controller.

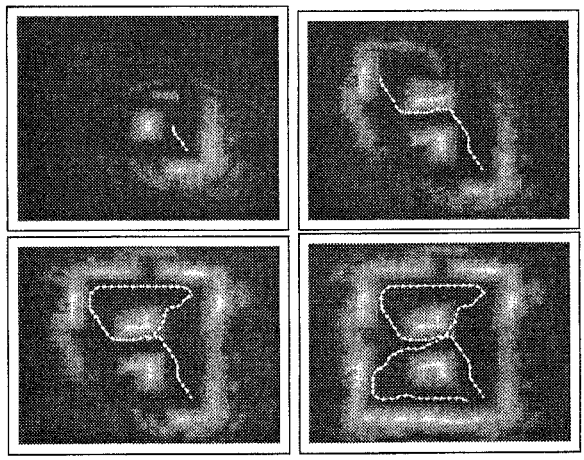

Figure 8: Navigating the robot in a real room.

6 Conclusions and future work

One major conclusion of our work is that different sources of sensorial information can be integrated in a common structure to allow the robust modeling of the environment in autonomous robot navigation. Experiments with an isolated source of information shown us that the majority of problems can be overcome by the inclusion of more sensing devices.

In our particular setup it was easy to conclude that, including vision, robustness and flexibility were added, while validation between different sources of information became possible. In the other hand, ultrasonic information is very fast to acquire, while image processing is always problematic to implement in real time. This way, with the use of ultrasonic sensors together with binocular active vision, we can be less demanding in processing power to deal with stereo image pairs.

But to allow successful integration of different sensorial flows a flexible framework must be developed. We believe that our grid based framework is a suitable approach to this task.

At this stage our prospects of future work are mainly focused on the exploration of the richness of possibilities given by the use of binocular active vision in mobile robotics [2, 3, 5]. Our initial setup only gives low level obstacle detection information, but we hope to expand the range of information obtained from stereo image pairs (e.g. 3D information).

In the near future we propose to extend the 2-D model to a 3-D one, again using the possibilities of stereo vision. Acquiring higher level information than simple obstacle presence from the model (e.g. obstacle structure) is another issue being considered.

References

[1] A. Silva, P. Menezes, J. Dias, H. Araújo. "A Connectionist Approach to Obstacle Detection and Avoidance", In *Proceedings of the* 27^{th} *Symposium on Industrial Robots*, Milan, October 1996.

[2] C. Paredes, J. Dias, A. de Almeida. "Detecting Movements Using Fixation", In *Proceedings of the* 2^{nd} *Portuguese Conference on Automation and Control*, Oporto, September 1996.

[3] I. Sousa, J. Dias, H. Araujo. "Normal Optical Flow Based on Log-Polar Images", *International Symposium on Intelligent Robotic Systems*, Lisbon, July 1996.

[4] P. Menezes, J. Dias, H. Araujo. "Low Cost Obstacle Detection and Description", *Fourth International Symposium on Experimental Robotics*, Stanford, June 1995.

[5] J. Dias, C. Paredes, I. Fonseca *et al.* "Simulating Pursuit with Machines", In *Proceedings of the 1995 IEEE Conference on Robotics and Automation*, Japan, 1995.

[6] Christopher I. Connolly and Roderic A. Grupen. "Nonhlonomic Path Planning Using Harmonic Functions", *UMass Amherst Computer Science Department Technical Report*, 1994.

[7] R. Glasius, A. Komoda and S. Gielen. "Neural Network Dynamics for Path Planning and Obstacle Avoidance", In *Neural Networks*, March 1994.

[8] D. Kontoravdis, A. Likas and A.Stafylopatis. "Collision-Free Movement of an Autonomous Vehicle, Using Reinforcement Learning", In *10th European Conference on Artificial Intelligence*, 1992.

[9] Eduard S. Plumer. "Neural Network Structure for Navigation Using Potential Fields", In *Procedings of IEEE Conference on Robotics and Automation*, 1992.

[10] J. Borenstein. "Real-Time Obstacle Avoidance for Fast Mobile Robots", In *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, N5, Septem ber/October, 1989.

[11] A. Elfes. "Sonar-Based Real-World Mapping and Navigation", In *IEEE Journal of Robotics and Automation*, Vol. RA-3, N3, June 1987.

[12] O. Khatib. "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", In *The International Journal of Robotics Research*, Vol. 5, N1, Spring 1986.