Visual Navigation in an open environment without map

P. Gaussier(*), C. Joulain(*), S. Zrehen, J.P. Banquet(**), A. Revel
ENSEA ETIS, 6 Av du Ponceau, 95014 Cergy Pontoise Cedex, France
(**) CREARE Université Paris VI, France
(*)first co-authors
e-mail: gaussier or joulain@ensea.fr

Abstract

In this paper we describe how a mobile robot controled only by visual information can retrieve a particular goal location in an open environment. Our model does not need a precise map nor to learn all the possible positions in the environment. The system is a neural architecture inspired from neurobiological studies using the recognition of visual patterns called landmarks. The robot merges those visual information and their azimuth to build a plastic representation of its location. This representation is used to learn the best movement to reach the goal. A simple and fast on line learning of a few places located near the goal allows the robot to reach the goal from anywhere in its neighborhood. The system uses only an egocentric representation of the robot environment and presents very high generalization capabilities. We describe an efficient implementation tested on our robot in two real indoor environments. We show the limitations of the model and its possible extensions to create autononous robots only guided by visual information.

1 Introduction

Most of the existing navigation systems use allocentric referentials like cartesian map representations or at least cartesian coordinates to locate the important objects they are computing about (the robot, the goal, the areas to avoid...). In well structured environments, the navigation problem consits in finding the best route to go from one place to another. The robot movements can be limited to simple orders such as go ahead, turn right,... because the robot can recognize a wall or a T junction (maze problem). These planification systems suppose the places are already learned and usually use improved versions of the A* algorithm to find the best route to reach a goal. These systems need an important engineering work to choose the information, to recalibrate the robot position, to check the robot current state, or to wait for the recognition of the next state when a reactive planning mechanism is used. In the case of a real autonomous navigation, if the robot forgets to learn a place or learns several times the same physical place, it becomes unable to navigate correctly (cut or infinite loop in the graph of its cognitive map - [3]).

In a less structured environment, when the robot does not move in corridors but must evoluate in a room or in any other "open" environment, potential field techniques [1] can be used. For each location the strenght of the attraction of the goal on the robot is computed. It implies at least to store the goal location and the robot location in a cartesian referential frame (need to compute precise trigonometrical computations). Unfortunately, odometry currently used to measure distances is not precise in a long run and must be recalibrated by other sources of information such as particular visual patterns called landmarks [4]. Thus in both structured and "open" environments, the actual main problem in realizing really autonomous mobile robots is linked to the problem of finding learning criteria such as how to choose the learned positions and how to regulate the learning level [5]. The way information is represented seems to be crucial to reduce the algorithm complexity. Indeed, if the robot had to learn each position in the environment before being able to navigate correctly, the learning time would be huge. Moreover, the robot would be unable to perform topological generalization.

In this paper, we show experimental results of a navigation model proposed in a previous paper [10]. It is a neural architecture named PerAc (Perception Action) based on animals navigation models which do not require a precise map of the environment to navigate. This model has been implemented on a mobile robot named Prometheus. It only uses a small number of panoramic views to decide which movement to perform in order to reach a learned position in an open indoor environment. In the first part of the paper, we briefly summarize the principles of our neural network model and show simulation results. In the second part, we present an efficient and low cost implementation (in terms of memory load and computation time) of the algorithm on a real mobile robot. Finally, we discuss experimental results and propose different improvements of this first implementation.

2 Landmark-based navigation

More and more systems take into account the fact that animals mainly use landmark information and navigate directly from 2D perceived images. This approach reduces their algorithmic complexity and increases their robustness. The PerAc architecture is a neural computation architecture proposed to solve a wide variety of control problems requiring learning capabilities (by opposition to adaptation [1] capabilities). It consists in an action level (a hardwired pathway able to play the role of a reflex mechanism) and a perception level trying to recognize particular situations and to associate them with the correct action through an associative or a reinforcement learning rule. The perception level allows the robot to react to a situation even if it is too complex to allow the action pathway to propose an answer (the goal is not in sight for instance) by generalization of previously learned situation (the landmark configuration in the navigation task). Moreover, if the action pathway induces a negative reward, the links between the recognition of the perceived situation and the current robot action can be inhibited and a link with an action avoiding a negative reward can be learned. At the beginning of the exploration phase, we suppose

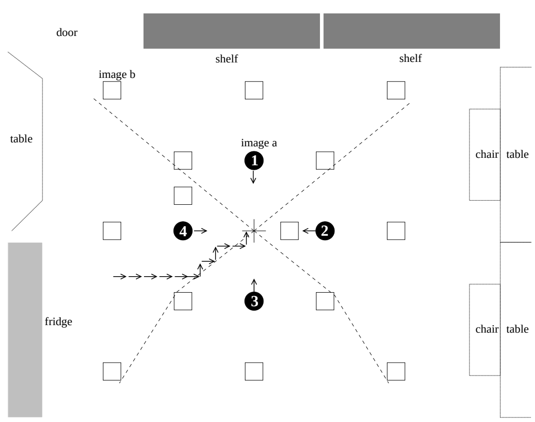

Figure 1: Four views (circles) are learned (i.e. associated with an elementary movement), others are associated as a function of the similarity between learned and tested places (boxes). As we can see, if the robot learn to reach the cross from each learned view, it can reach the cross from all the views. The set of arrows represents a possible path. The views are in a $1.2m \times 1.2m$ area, the learned views are at 30 cm from the center. The scale is not respected for the position of the different furniture (in fact they are about 1.5 m from the center)

Prometheus moves randomly, looking for something interesting. When it finds its goal, it moves around it in order to explore various positions in its proximity. At these places, it learns both the landmark configuration (represented by a set of local views and their angles) and the direction leading to the goal. Later, when the robot wants to find the goal, it considers the information of the "place cells" (i.e. the cells which react to a specific set of local views associated to their azimuth) and moves in the direction associated to the most activated "place cell" (competitive mechanism) to reach the goal [2]. Thus at each step, the distance to the target is reduced (Fig. 1) and the robot returns inevitably to the learned position. A complete description of a neural implementation of the learning process can be found in [10]. The PerAc architecture for place learning realizes an approximation of a potential field function without the cost of learning what to do from each position in the environment.

When the goal is in sight (goal recognition), a neuron corresponding to its angular position relative to the robot's facing position is activated in the Target Azimuth group (we suppose that the robot has previously learned what the goal looks like [12]). A shifting mechanism activates a neuron in the Direction of Movement Proposal (DMP) group by adding an angle corresponding to the angle between the robot and the north direction. The inverse shifting mechanism is applied to the output of the Direction of Movement group, by substracting the same angle. Thus the robot movement is correct whatever the robot orientation is (conservation of the sensory topology in the whole neural network architecture).

First to recognize a place, the robot must be able to isolate a local view (focus of attention mechanism that will be presented later). Next, the information about the landmarks recognition and the associated angle are merged to produce a unifyed representation that can be easily learned and matched with previously learned representation. That merging is represented by the matrix product of the information corresponding to "what" (Landmark) and "where" (Azimuth) the objects are (Fig. 2 Landmarks Azimuths group). A time integration process allows to suppress the sequential aspect of the scene exploration (spatio-temporal merging). With this representation, there is no need to "recognize" specifically what the landmarks are (a fridge, a chair...), it is only important to distinguish them and to know their angular position. Even if a landmark is missing (for instance if the fridge is removed), because the "image" of the scene (Landmarks Azimuths) is noisy, the other landmarks can allow a good recognition (we have shown that several landmarks can be removed, hidden or displaced without disturbing the global recognition of the scene [13]). This "plastic" merging by opposition to the static recognition of a multisensor config-

[1] Learning is something that can be performed quickly and involves a variation in the structure of the control architecture whereas the adaptation consists in a slower variation of the structure parameter. Adaptation can induce learning if there is a non linear modification of the adapted parameters that induce a non linear variation of the system response.

[2] Those "place cells" are called like this as a reference to the biological place cells recorded in rat hippocampus[11].

uration seems to be performed by a brain region called the hippocampus and involved in the memorization and navigation processes (processes that we study in the frame of a neurobiological project [2]). The place rep-

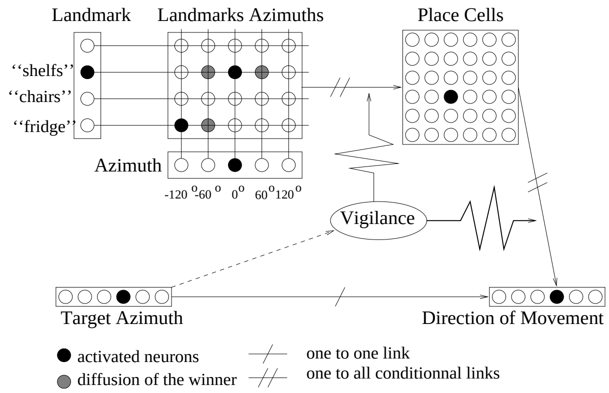

Figure 2: The navigation neural network. Target azimuth and Direction of movement are WTA. Landmarks azimuths emphasizes merging of visual and motor flow.

resentation (Landmarks Azimuths) is learned by the Place Cells group of neurons. If the robot recognizes the goal, it moves in that direction (reflex link between Target Azimuth and Direction of Movement on Fig. 2). Otherwise, learned association between Place Cells and Direction of Movement allows an efficient generalization to the whole environment. Consequently, this process creates a basin of attraction in which the robot always moves in order to come closer to the goal. Moreover, it has been mathematically proved that there was no local minimum induced by the competition between the action neurons within the domain bounded by the set of landmarks [10]. To simulate visual navigation in

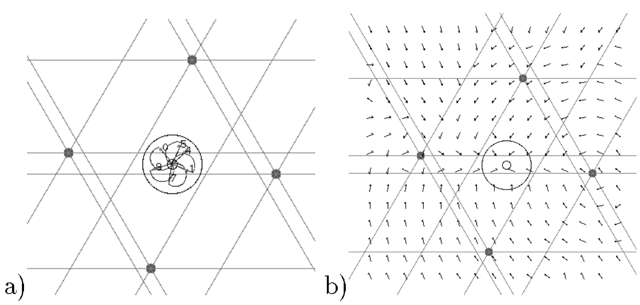

Figure 3: Simulated environment with complex landmarks. a) Each black circle represent a landmark which present 6 differents aspects according to the observer point of view. The small numbers represent the places where a panoramic view has been learned by the robot (the drawing between those circles represent the robot trajectory during learning). b) Vector field representing the robot movement direction when using the east landmark as angle origin.

a real environment, we have supposed that landmarks do not have the same aspect according to the robot location (see Fig. 3). The simulation results are even better than in the case of cylindrical landmarks. The robot is always able to join its goal and moreover the place recognition is more robust because the variation of place cells activity is more significant from one location to the other (the competition mechanism can then be less precise). If no absolute direction is available, a landmark can be used as orientation reference but then the generalization capabilities to long distances is reduced. However, the place cells responses remain correct if the robot stays in the area surrounded by the landmarks.

3 An efficient implementation

In order to test our model we have implemented the algorithm in C.

3.1 Sensors and environment

The visual input comes from a 384×288 grayscale CCD camera. Its field of view is about 70 degrees. To build the panoramic view a servo-motor is used to pan the camera. The robot takes 24 images with a 7.5 degrees rotation between each capture. The central vertical band of each image (30×288 pixels) is merged to constitute the global panoramic view (the central band is only used because the camera distorts the image sides). A 1066×288 pixels panorama is obtained. Its field of view is about 270 degrees. It is not a complet 360 degrees image but it is sufficient for our application. As you can see on figures 4,5,11 the images merger is not perfect but it works (which shows the robustness of our system). The test environments are our every-day

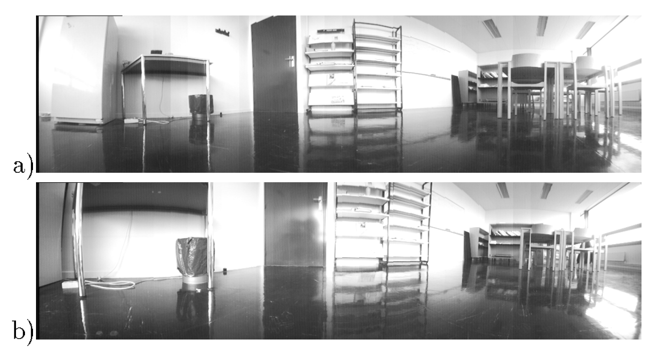

Figure 4: a) and b) Panoramic images constructed by our system referenced in Fig. 1.

working rooms with the chairs, tables, shelfs, workstations, dustbins... We use the environments as they are without any change. In addition, the more complex the environement is, the more numerous landmarks can be, and thus the more efficient our algorithm is.

3.2 Implementation

The first step of the navigation algorithm is to find where the possible landmarks are. All the image rows are averaged and weighted with a larger weight for the

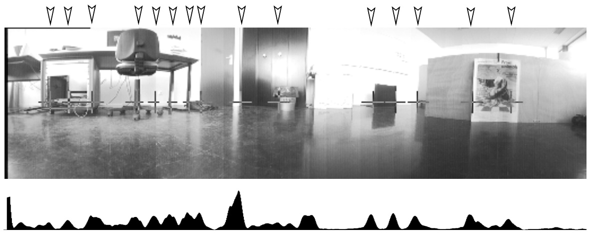

Figure 5: 15 local views stored in a learned image (environment Fig. 8, black circle S), and the absolute value of the derivative.

points near the center of a row. The resulting one dimentional signal is derivated and the local maxima and minima are used to point out what will be the center of local views (an example derivative absolute meaning is shown on Fig. 5). Each panorama projection contains, in average, 20 local maxima. On figure 5, 15 maxima are extracted to be the center of 15 local views. For each selected point, a 32×32 pixels local view is built by averaging the 148×288 pixels of the corresponding panoramic image part. The $\mathbf{y}$ axis is just scaled whereas a logarithmic transformation is used for the $\mathbf{x}$ axis. Then each local view is compared with each learned local view. This comparison is a simple difference between the pixels of the local views. The five best corresponding views are used as landmarks, i.e. their positions in the image are compared with the ones in the learned panorama. The sum of those absolute angles gives us the similarity between panoramas. So the movement associated to the best corresponding panoramic image is performed.

The complexity of the algorithm for one panorama analysis is about 14 millions integer additions and 1 million floating point multiplications. More than 95% of the calculation time is dedicated to the creation of the local views, the rest is spent in views comparison. The total calculation time is less than 1 second on a Pentium 133 and could be easily reduced. In fact, when the robot had learn 10 positions, it performs a movement every 15 seconds. About 13 of those 15 seconds are spent in the acquisition of the panoramic image (camera rotation).

4 Experimental results

The algorithm has been tested in a large number of rooms, some plans can be seen on figures 1 and 8. We show that our algorithm works in large open areas with a very high precision in target reaching. Next, we change the robot head direction and introduce perturbations and show the robot continue to run correctly. New results, not describe there, show that our robot can reach the goal from a $20m^2$ start area in a $7.2m \times 5.4m$ room with obstacles and only 10 learned positions[6]. Simple obstacle avoidance is implemented through a low level reflex mechanism that used the infrared sensors of the robot[9].

4.1 Place cells response

Results on figures 6 and 7 show how the different positions in the environment are far from the learned positions. Figure 6 is a combination of the four responses of the place cells by selecting the maximal response (a gray level is associated to each direction).

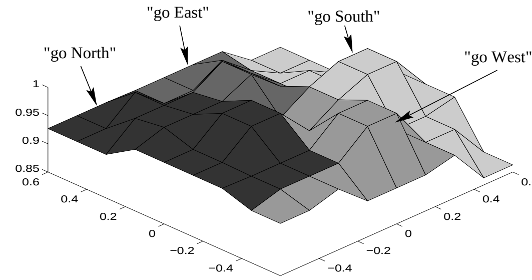

Figure 6: Maximal place cell response in the experimental area (environment figure 1), unit is meter, a gray level is associated to each direction (verification on Fig. 1).

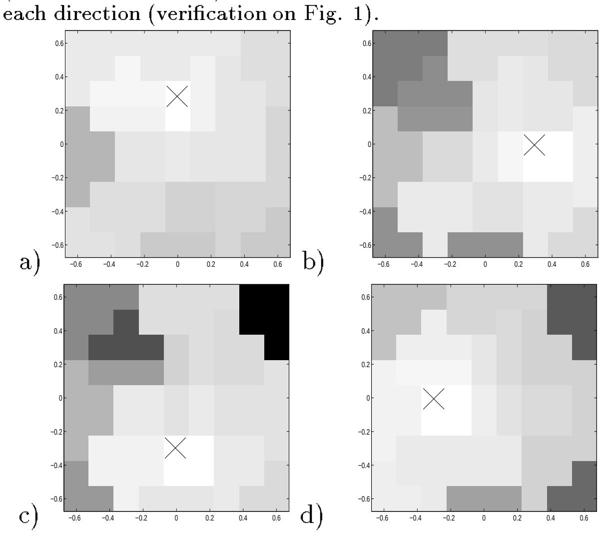

Figure 7: Response of each place cell when covering a $1.2m \times 1.2m$ square (environment Fig. 1), white color represents high place recognition. The cross represents the learned view position, the goal position is 0, 0.

Another environment is used (Fig. 8) to show precisely the variation of the place cells activities of two arbitrary locations. On figure 9, the responses of two place cells when the robot is moved on line D are shown. Those learned positions are the place "i" in figure 10 for figure 9 a) and the place "ii" in figure 10 for figure 9 b).The maximum activity of the place cell is always obtained for the learned location and it monotonously decrease on a large distance. These results emphasize the fact that even with large distances (comparing to the robot size and to the environment size), the robot is able to perform an action in order to come closer to the goal.

4.2 Movement precision

Figure 10 shows a real path taken by the robot to reach the goal. We can see that the closest the robot is from the goal is less than 2cm (note that the width of each

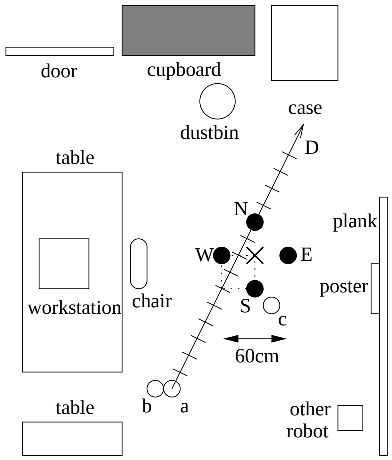

Figure 8: A second room for the experiment, the white circles and line D represent several tested positions. Four views are learned (black circles). The dotted rectangle corresponds to figure 10 area, the cross represents the goal.

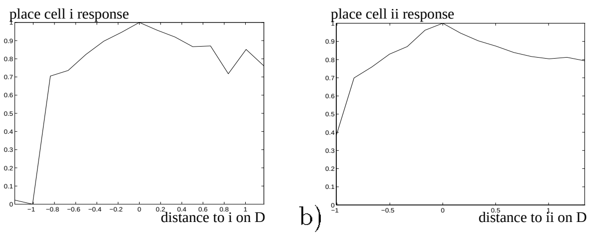

Figure 9: Response of a place cell (learned view) according to the distance to i and ii in meters(environment Fig. 8, 10).

move is about 4.5 cm in this experiment). In fact, the smallest distance is less than half the width of each single movement (about 2cm). In theory, with landmarks at distance d, the precision p representing a 1 pixel shift in the image, is: $p = tan(1/4) \cdot d$. It comes from the field of view being about 270 degrees, the **x** axis made of 1066 pixels that each pixel represents about 1/4 degree. Thus, with landmarks at 1.5m, the maximum precision is 7mm and with landmarks at 15m it would be 7cm. The starting point of the robot can be very remote from the goal. The robot is still able to reach the goal with great precision.

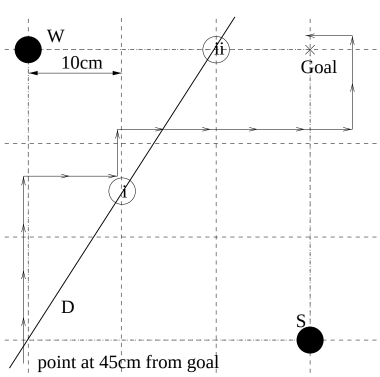

Figure 10: An example of a real path taken by the robot to reach the goal (environment Fig. 8). This area is a zoom of the dotted rectangle in Fig. 8. The arrows positions are those of the center of the camera.

4.3 Change in head direction

Till now the camera was pointed at North at the begining of each mesure. We have tried to change its orientation, the results were thoses expected. All landmarks were shifted in the image by an angle corresponding to the change in orientation. So, with a compass sensor[3] giving an indication of the camera direction, we only have to substract this rotation angle to the angle found between viewed landmarks and learned landmarks. After this shift, the comparison is made with the learned views and the best corresponding movement is performed (according to the body orientation -see section 2-). We have tested several orientations at position S (Fig. 8 and 10), after the shift the activation of corresponding place cells was exactly the same as before up to a 1% error, in spite of the loose of few landmarks in the blind area of the robot.

4.4 Introducing distractors

On figure 11 a), which corresponds to the panorama from position 'c' (Fig. 8), we can see the positions of the five best recognized local views. Thoses views are used as landmarks and the absolute angles with landmarks direction in learned views (black circles Fig. 8), give position S as best matching. So the movement performed is "go North" which reduces the distance to the goal, it is the best of the four possible movements. Then the robot is put back to position 'c' and a perturbation (a person) is introduced in the scene (Fig. 11 b). This perturbation occults one of the landmarks and introduces new possible landmarks. But as you can see on figure 11 b) there is no problem. Instead of the occulted landmark, a new landmark is selected. The system uses the other landmarks and the best matching place is still S, the robot keeps on performing the "go North" movement. This example emphazises the robustness of the system and shows the advantage of using only some local views as landmarks without trying to know what they are.

a)
b)

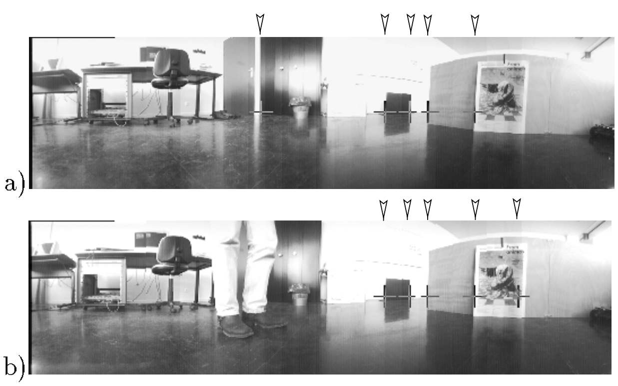

Figure 11: Changement of landmarks when introducing a perturbation (environment Fig. 8, position 'c').

[3] Pewatron sensor model 6070 for example

4.5 Limitation of system

As a first limitation, we must suppose that the visual system can differentiate all the landmarks. We cannot allow the same landmark found twice in the same panoramic view. Otherwise the system would not succeed in knowing which azimut is associated to which landmark. So in the case several landmarks are the same, we suppose the visual system will either use information about the neighborhood or choose a particular landmark to index all the others by reference to it. This implies to learn a sequence and not just to recognize a snapshot. Because sequence learning seems to be one of the major role of the hippocampus in the brain functionning we hope to introduce this kind of feature in the next version of our model. A second limitation results from the decrement of place cells response accuracy as a function of the distance from the goal. For instance, the robot performed a wrong movement when it reaches position 'a' (Fig. 8). Then, the proposed action was "go West" and the robot reached position 'b'. It was a mistake, but in position 'b' all the activities of place cells corresponding to N, E, S, W are less significant than in position 'a'. Our system allows to know if the action reduces the distance to the goal just by looking the evolution of the place cells (Fig. 9). So the robot knows that its movement was wrong. Obviously more efficient trajectories could be obtained if the movements are performed randomly according to their associated neuron activity rather than according to a deterministic WTA mechanism [8]. Moreover, our N.N. can also be used to avoid particular zones or to introduce other goals [9].

5 Conclusion

Our algorithm works correctly even in difficult situations. It supports a lack of landmarks or a misinterpretation of a few of the landmarks. There is no need of a particular number of landmarks (more than 2). To recognize a place, the precision will only grow with the number of landmarks. Our work shows navigation in an unknown environment can be achieved without any complex learning mechanism (only associative learning). The meaning of viewed objects does not need to be really understood by the robot (all the views associated to the same object are not explicitly linked to each other). Our system is intended to be in interaction with its environment. It is just an agent that learns to agree with its environment and its internal motivations. It has no global or complete representation of its world. It "keeps in memory" the link between a particular situation and the action that it has learned to be correct in that situation. Should the universe collapse, the robot's memory would have no more meaning. For instance, a cognitive map involved in high level goal seeking will be simulated in Prometheus with only few neurons in competition (representing a well chosen set of learned places). The decision to learn a new place can be performed with the information about the goal recognition. For instance, if the best recognized view is not correct, the robot can move in a bad direction but then at its new position the global activity of the place cells will decrease (see experimental results). It is easy to build a learning rule that is triggered when the sum of the place cells response decreases [9]. The robot would then find a movement that allows it to go in a direction associated to a global increase of the goal recognition (an efficient reinforcement learning rule is described in [7]). Our future work will consist in testing for real a planification level allowing the robot to pass from one subgoal to another in order to reach a particular goal.

References

[1] R.C. Arkin. Motor schema-based mobile robot navigation. *International Journal of Robotics Research*, 92–112, 1987.

[2] J.P. Banquet, P. Gaussier, J.C. Dreher, C. Joulain, and A. Revel. *Cognitive Science Perspectives on Personality and Emotion*, chapter Space-Time, Order and Hierarchy in Fronto-Hippocamal System: A Neural Basis of Personality. Elsevier Science BV Amsterdam, 1997.

[3] G. Bugmann. *Basic Concepts in Neural Networks : A survey*, chapter 4 :A Connectionist Approach to Spatial Memory and Planning. Perspectives in Neural Networks. Springer, 1997.

[4] R. Chatila. Control architecture for autonomous mobile robots. In P. Gaussier and J.D. Nicoud, editors, *Proc. of Perception to Action*, 254–265, Lausanne, 1994. IEEE Computer Society Press.

[5] M.J. Denham and S.L. McCabe. Biological temporal sequence processing and its application in robot control. In *Control96*, 1266–1271, 1996.

[6] P. Gaussier, C. Joulain, S. Moga, M. Quoy, and A. Revel. Autonomous robot learning: What can we take for free ? In *To appear in International Symposium on Industrial Electronics - ISIE'97*, Gumarães, Portugal, July 1997. IEEE.

[7] P. Gaussier, A. Revel, C. Joulain, and S. Zrehen. Living in a partially structured environment: How to bypass the limitation of classical reinforcement techniques. *to appear in Robotics and Autonomous Systems*, 1997.

[8] P. Gaussier and S. Zrehen. Avoiding the world model trap: an acting robot does not need to be so smart! *Robotics and Computer-Integrated Manufacturing*, 11:279–286, 1994.

[9] P. Gaussier and S. Zrehen. Navigating with an animal brain : a neural network for landmark identification and navigation. In *Proceedings of Intelligent Vehicles*, 399–404, Paris, 1994.

[10] P. Gaussier and S. Zrehen. Perac: A neural architecture to control artificial animals. *Robotics and Autonomous Systems*, 16(2-4):291–320, 1995.

[11] J.O'Keefe and N. Nadel. *The hippocampus as a cognitive map*. Clarendon Press, Oxford, 1978.

[12] C. Joulain, P. Gaussier, and A. Revel. Learning to build categories from perception-action associations. In *To appear in International Conference on Intelligent Robots and Systems - IROS'97*, Grenoble, France, September 1997. IEEE/RSJ.

[13] S. Zrehen. *Elements of Brain Design for Autonomous Agents*. PhD thesis, Swiss Federal Institute of Technology Lausane, 1995.