Self-Learning Visual Path Recognition

Phillip Chan and Gordon Wyeth
The University of Queensland, Department of Electrical and Electronic Engineering
St Lucia, Queensland 4072
Australia

Abstract

In this paper the feasibility of equipping a mobile robot with the ability to learn a path, in real time, using principles based on insect-based navigation skills and a biologically plausible neural network model inspired by the "Conjunction of Local Features Network" (CLF network) through both real-world and controlled environment experiments is presented. Results shown for experiments on a prototype LEGO robot indicate that memory-based navigation using the proposed navigation network is suitable for simplified environments.

1 Introduction

Until recently most approaches to robot visual navigation were based on symbolic representation of the world in terms of known structural information. One of the most fundamental problems with the symbolic paradigm is that a high degree of prior knowledge of the robot's environment must be known, thus reducing the structural complexity of the environment for which the robot can be operated in. The dependency on structural information makes this approach unsuitable for exploratory navigation tasks, which directly affects its ability to operate in a real world scenario.

Today, some roboticists have turned their attention towards a memory-based approach [Owen et al 95] and [Rao & Fuentes]. The fundamental principle of memory-based navigation is the process of sensory-motor coordination. It can be described as a form of sequence learning, where the robot navigates by remembering what it sensed along a learned path. Although experiments with insects [von Frisch 71] & [Srinivasan et al 95] have shown the biological plausibility of this approach, little work has yet been done using vision system.

The goal of this research is to demonstrate a biological plausibility memory-based visual navigation network based on biological principles derived from the study of insect navigation (mainly bees). This visual navigation system will be demonstrated on a real-time, visually guided mobile robot performing the task of path recognition in a controlled environment. The robot does not navigate its way within its environment using *known* landmarks or pre-defined structural information of its environment but rather *learns* a sequence of visual impressions of its journey that it is later able to recall.

The results presented in this paper are obtained from preliminary experiments conducted. The goal of this research is to create a robot with a foraging behaviour, which will set off in search of a pre-defined type of object (food) from its home location (hive), using its vision and sonar sensors for obstacle avoidance. Once a source of objects is located, the robot collects the object deposit and returns it to its home location. Navigation for the robot's homing is guided solely by visual and orientation information. The robot will continue to return to the located source using learned vision and compass information until the food supply is exhausted. Having exhausted the located source, the robot will return to its exploration mode and set off in search of another potential supply.

The robot will be completely autonomous, with its camera, encoder on each of its wheels for path integration, electronics and battery on board a wheel based chassis. A network (which will be presented in this paper) based on the Conjunctions of localized features (CLF) network [Edelman & Poggio] will be used as its control architecture due to its ability to perform unsupervised learning, allowing the robot to learn visual landmarks as it travels from its hive to the food source.

The experiments presented in this paper were conducted using the prototype robot shown in Figure 1. The chassis of the robot is constructed using Lego, motion control is achieved using Lego Interface card, visual input is provided using a Video Blaster SE100 card and a grayscale CCD camera. There are no obstacle avoidance sensors onboard.

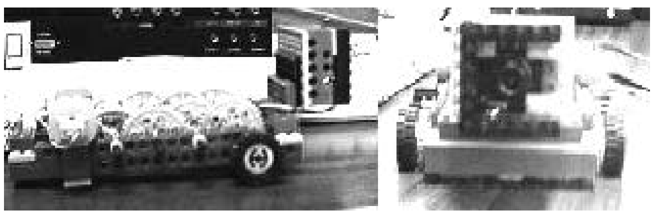


Figure 1. Prototype Lego Robot

2 Insect Navigation Principles

Many hypotheses have been proposed and experiments performed on insect navigation ([von Frish 71], [Srinivasan 95] & [Collett 96]). They each illustrated the amazing abilities of foraging bees during flight, but they failed to explain insects robust navigation ability as a whole. However, if the underlying principle of these navigation strategies demonstrated by honeybees can be learned, one would see immediate implications in the implementation of mobile robots. Following this line of thought, the following list provides a guide of the better known theories.

- **Path integration or Dead-reckoning:** Path integration or dead-reckoning [Dyer 96] & [Collett 96], this process uses a directional reference (either an external compass reference or an inertial reference) provided by vestibular or somatosensory feedback to maintain a continuously updated egocentric representation of the animal's position relative to its home.

- **Snapshot Model:** The snapshot model [Cartwright & Collett 83], where the direction of flight is calculated from a process which involves the comparison of the snapshot image of the goal and the current retina image. Cartwright & Collett from their experiments indicated that bees memorize a nearly unprocessed retinal image of the goal location, this image is believed to contain no explicit information about the distance of the objects on the retina. The comparison of the snapshot and retina image is only performed when the orientation of the insect at the time of the snapshot, matches that of the present orientation.

- **Image Matching:** Image matching [Collett 96], one method which is believed to help bees to pinpoint their final approach to its goal is called image matching, where a previously stored snapshot of the goal is compared with the current retina image to align the insect before touch down. During this image matching process, bees are reported to circulate in a figure of eight aligned in a preferred orientation.

Path integration, the snapshot model and image matching hypotheses individually provide explanations to different aspects of honey bee's navigation abilities, however, they fail to answer in regard to insect navigation as a whole, such as their robustness in path recognition as demonstrated by [Collett 96]. The combination of these hypotheses however, seems to provide the best yet explanation to insect's robust navigation skill. In both models, insects appear to guide themselves by following the paths that produce the best match between current and remembered landmarks, visual impression or some homing vectors.

Following this line of thought, the following model is proposed. On the robot's first outward journey, it records a sequence of images throughout its journey and the orientation in which each image is taken. Subsequent homing is performed through a process similar to that of dead-reckoning, except that only orientation information is stored, with no distance information. Matching of current image to stored sequence is done when the robot has orientated itself to that of the recorded orientation, (preferred viewing orientation [Collett 96]). Once the robot's current location is matched, the orientation at which the match occurred becomes the new heading direction for the robot. For the return path, the heading direction is in the opposite direction of the matching orientation. This process repeats until the robot reaches the vicinity of its goal (food source or hive). Once the robot has arrived in the vicinity of its goal, the robot uses the short range beacon signal of the food source to zero in for food collection.

3 Proposed Navigation Network

3.1 System Overview

The input to the visual navigation system shown in figure 2 is a 64x64 gray-scale image from the robot's onboard camera. The input image is analyzed by two processes operating in series. The first one, the *highlighting mechanism* (section 3.2.1) defines the spatial extent of regions containing the most "different" information of the image, followed by the second process, the *resolution reduction mechanism* (section 3.2.2) which reduces the resolution of the 64x64 image into a 10x10 representation map. Although details are lost after this process, the resultant image is able to resist error caused by minor image displacement

The processed image is then passed into the visual navigation neural network (section 3.3), together with the robot's orientation or action input. The purpose of the VNNN is to form a spatial representation of the relation between each unique visual impression of a location of the robot's journey and the orientation or action taken by the robot at the time of the image acquisition.

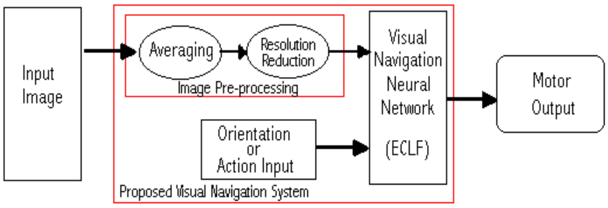


Figure 2. Navigation System Overview

3.2 Image Preprocessing

3.2.1 Highlighting Operation

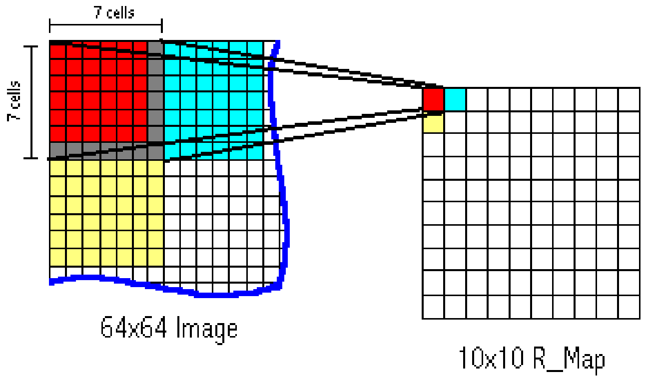
The first level of pre-processing uses a similarity detection algorithm (similar to edge detection except it is more resistant to error under different lighting conditions). The Similarity detection highlights areas from the CCD camera image that are interesting, different or unexpected. The value of each pixel e is calculated by comparing its value to the average of its surround pixels. This operation is defined as:

$$p(x,y) = \begin{cases} 1 & if\ \frac{1}{9}\sum_{x-1}^{x+1}\sum_{y-1}^{y+1} p(x,y) < T \\ 0 & otherwise \end{cases}$$

where T is a predefined threshold, and p(x, y) is the pixel value at (x, y).

3.2.2 Resolution Reduction Operation

The resulting image is then passed to the second level pre-processing which maps the 64x64 image into a 10x10 representation map. This mapping is done by applying 100 7x7 frames over the 64x64 image overlapping each other as demonstrated in figure 3. The region that each frame enclosed is then averaged and compared with a preset threshold, if the average of a region is greater than that

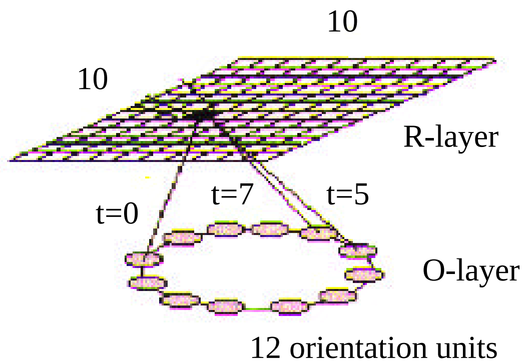


threshold then the corresponding cell in the 10x10 map is selected forming the 10x10 R-layer.

Figure 3. Resolution Reduction Mechanism

3.3 Navigation Neural Network

At the heart of the visual navigation system is the image sequence learning mechanism, a biologically plausible model based on the *Conjunction of Localized Features* (CLF) [Edelman & Poggio 89]. The structure of the network consists of two layers shown in figure 4. The first layer is the Representation layer (R-layer), this layer is made up of 10x10 units representing the pre-processed 64x64 pixels image from an onboard CCD camera. Every unit in the R-layer is mapped to all units in the second 12 units



Orientation layer (O-layer).

Figure 4. Structure of the Extended CLF network

The R-layer consists of 100 units representing the visual impression of any given location in the robot's journey. In the same way, the O-layer consists of 12 units representing the orientation of the robot to a given reference point. This can either be a geographic, magnetic or solar reference point. This layer will be driven by a compass during learning.

3.3.1 L-connections

The R-units in the representation layer are connected among themselves by lateral (L) connections, whose initial strength is zero. The L-connection forms the representation of a single visual impression of an individual location along a route. When an image is applied to the network, a set of R-units will become active. The lateral connections among any two activated R-units, b and d is updated by:

$$\Delta W_{bd} = \Delta W_{bd} + \left(\frac{W^{\max} - W_{bd}}{2} \right)$$

where W^{max} is the predefined maximum connection weight.

To allow each R-unit to encode the spatial structure of a specific view / visual impression, responding selectively to that view only, a unit activation strength A_a is assigned to each R-unit and is calculated using its lateral connections by:

$$A_a = \sum_{i=0}^{100} W_{ai} a_i$$

where a_i is the activation of R-unit i and a_i = 1 if R-unit i is activated else a_i = 0.

3.3.2 V-connections

Whereas the L-connection forms representation of the visual impression of locations, the V-connections form the associations between an individual location (activated R-units) and the robot's orientation (activated O-unit) at the time when the image was acquired.

The initial strength of the "vertical" (V) connection between an R-unit and an O-unit is zero. Hebbian relaxation rule where weights are updated by the correlation between input and output activities (A_a, A_{ij}), that is the activities on both ends of the link, is used to update the V-connections from the R-layer to the active O-unit. The change in connection strength V_{ab} from R-unit a to O-unit b=(i,j) is given by:

$$\Delta V_{ab} = \min\{a\, V_{ab} A_a . A_{ij}, V^{\max} - V_{ab}\} . \frac{V^{\max} - V_{ab}}{V^{\max}}$$

where A_{ij} is the activation of the R-unit (i,j) after WTA, V^{max} is an upper bound on a connection strength and α is a parameter controlling the rate of convergence.

3.4 Learning

During learning (as illustrated in figure 2), a sequence of visual impressions of the robot's surrounding is captured and used as inputs to the network. Winners on the R-layer are then identified using pre-processing mechanisms mentioned in section 3.2. The weight vector and thresholds of V-connection of the winning unit is then updated according to the rule above, followed by the adjustment of the L-connection among the R-layer's units. At the same time the winning R-units are identified, a unit on the O-layer will be firing according to the compass input. The connection weight of the two active units on the R and O layers are then associated as described above, forming an association of the captured visual impression and the orientation of the robot at the time of acquisition of the snapshot.

3.5 Recalling

During testing (as illustrated in figure 2), the learning of the network ceases. At any given time, the visual stimulus of the robot's current location will trigger some R-units to fire more strongly than others, and through the V-connection, an O-unit will be triggered. If the robot's current orientation matches that of the activated O-unit, these R-units are selected as the representation of the current location. The robot will continue to move in its current direction until the acquisition of the next snapshot where this process is repeated.

On the other hand, if the robot's current orientation does not match that of the activated O-unit from the winner of the current visual stimulus, then the current activated R-unit is dismissed and the same comparison process repeats on a different R-unit with the next highest activation value. If a match is made then the robot precedes as before. However if no match is made, the robot will classify the current visual input as unknown, and it will rotate at that point obtaining new visual input of that location from different point of view, this behavior imitates the "Turn Back and Look" behavior described in [Lehrer 93]. The robot will continue to wander around until a match is found.

4 Experiment Setup

Lego robot shown in Figure 1 was constructed for this experiment. An onboard miniature gray-scale CCD camera connected to a Video Blaster SE100 capturing card, provides visual input to the host computer where all images and networks processing is performed. The movement control signals are generated in the host computer and sent to the robot via a Lego controller unit shown in figure 5.

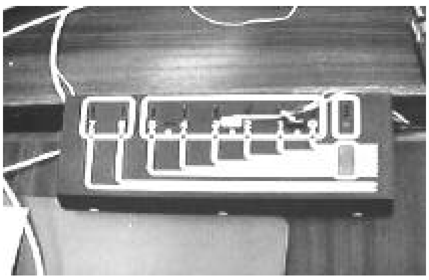


Figure 5. Lego controller Unit

5 Results

5.1 Experiment 1: Manual Path Recognition

In this experiment the robot is manually moved through the arena shown in figure 6 during training, it uses the top of the arena as its fix reference point (North), during training, the robot's orientation is updated manually in relation to this reference point. Figure 7 shows the robot and its network's output as it travels through the arena after the initial training. The motion of the robot as shown is by incremental position of equal spacing to its learned path. The result of this experiment demonstrated the network's ability to recognize a learned path where each location along the path can be identified by its distinctive visual impression.

Further experiments have been carried out using similar visual impressions along the journey in an attempt to confuse the robot. The result of these experiments shows the navigation network's ability to overcome this problem using the method described in section 3.5.

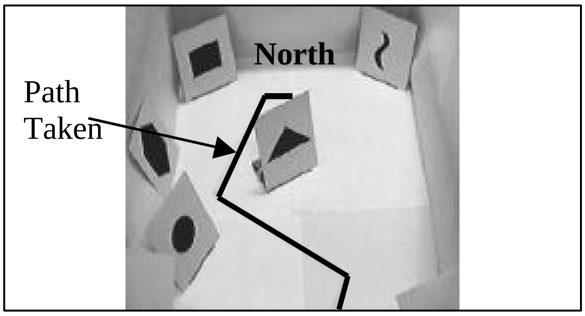


Figure 6. Experiment 1 Setup

5.2 Experiment 2: Automated Path Recognition

The goal of this experiment is to evaluate the network's ability to guide the robot autonomously on a learned path by observing the robot's behavior during testing. Due to the nature of the testing environment and setup, no suitable orientation sensor can be used, therefore, in this experiment actions of the robot are used as inputs instead of its orientation to a global reference point. Hence instead of forming a map in the sensor space, the navigation network will form a mapping in the action space. The path taken by the robot during training is shown in figure 8 where the robot is to move forward for about 50 centimeter then turn at about 85° to its left and precede forward until destination. Figure 8 shows the results of the testing run. Notice the point at which a change of action occurred (Point X in figure 8) where the robot started to turn to its left, this point was correctly mapped into the network and executed during testing, however, the robot started to move forward again before the turn was completed

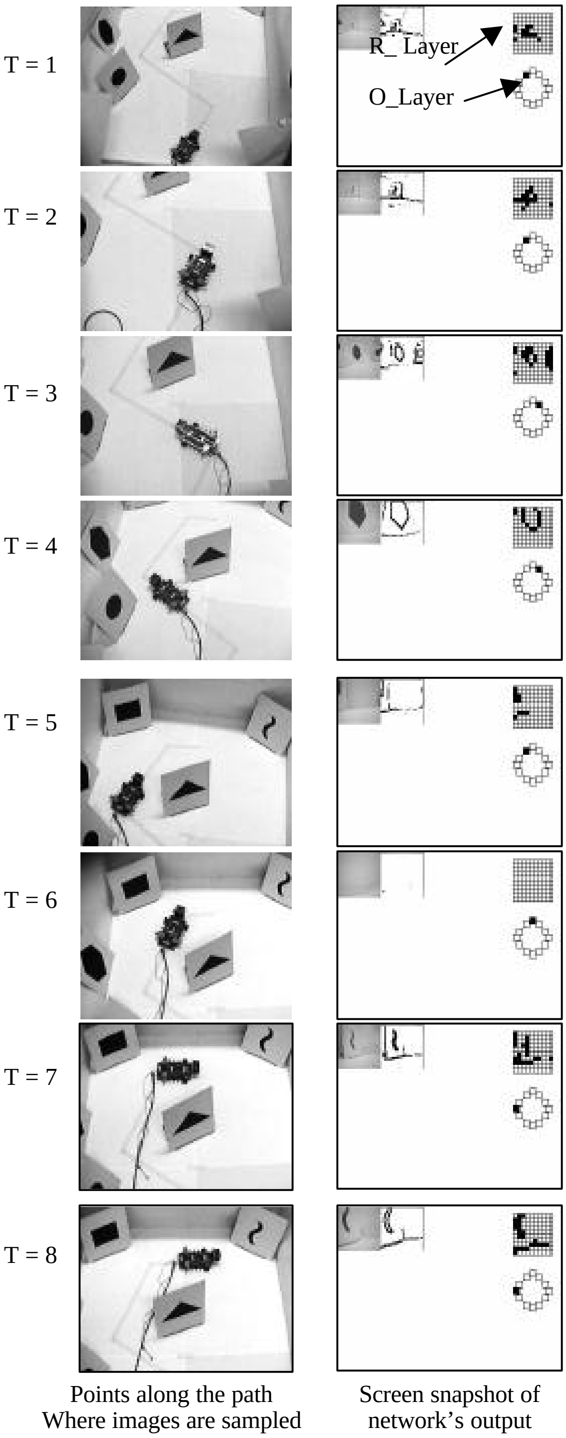


Figure 7. Navigation Network's output as it travels through the learned path. Notice that the reference point used in this experiment is the top-mid point of the arena.

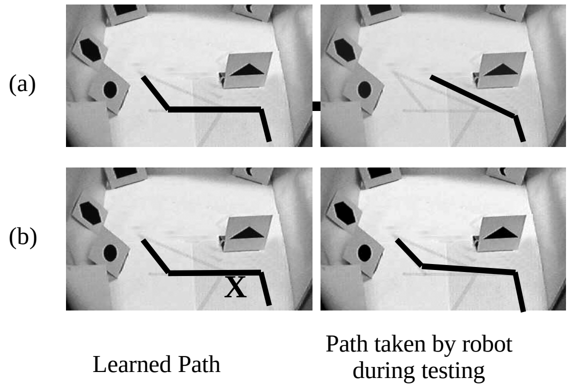


Figure 8. Results obtained from allowing the network to drive the robot after being trained along the training path one time manually. (a) Robot's trajectory after initial training. (b) Robot's trajectory after retraining at point X.

6 Discussion

Up to this point, all testing on the robot was done manually hence removing any error that can be caused by perception - reaction delay that is bound to exist in any control system involving vision. This perception-reaction delay could well contribute to the error in the test run. Another factor that could affect the mapping of the network is the issue of over-learning. Since the robot spends most of its time moving forward and only relatively short period of time in turning, V-connections between R-Units and O-unit[0] (representing forward motion) would naturally be much stronger than that of O-unit[9] (representing left turning motion). Hence there exists a bias towards forward motion. To correct this problem, the robot was placed at point X again and the turning action is retrained manually at that point. At the completion of the retraining, the robot was placed back to the starting point and the test was repeated. This time the network successfully guided the robot to its destination.

Another interesting finding from this experiment is the robustness of the proposed network against displacement. Several test runs were conducted with the robot placed in offset in all directions to its original starting located, even despite of the displacement of the robot from its learned path, it was found that the network was still capable in guiding the robot to its destination successfully.

In future work, it is aimed to resolve the error caused by over-learning and bias for one orientation over another as illustrated in experiment 2. This error arose when two or more locations existed, such that the visual impressions of those locations are too similar for the network to distinguish. The error may be resolved by introducing the notion of uncertainty into the visual impression evaluation process. Whenever an input image is presented to the network, if its uncertainty value is greater than a predefined threshold, then that image is simply discarded and a new image is acquired in a different point.

In response to the success in the preliminary experiments, a fully self-contained, autonomous robot will be built, moving all processing on board and removing the need of the host computer. Equipped with improved navigation network, the robot will be trained and tested on real world environment.

[Veneri, et al 97] presented a visual-based autonomous system capable of memorizing and recalling sensory-motor association. Their robot's behaviours are based on learned associations between its sensory inputs and its motor actions. Perception is divided into two stages. The first one is functional: algorithmic procedures extract in real time visual features such as disparity and local orientation from the input images. The second stage is mnemonic: the features produced by the different functional areas are integrated with motor information and memorized or recalled.

Computationally, our method is attractive over [Veneri et al 97] for a number of reasons. The visual pre-processing implemented requires very little overhead (as compared to the stereopsis used in Veneri's system), allowing for very fast recognition with ability to resist error caused by minor displacement. The working of Veneri's navigation system relay heavily on the extraction of features from its environment, this means that it must have a small degree of pre-knowledge of its environment, in comparison to our approach where the system does not rely on structural information of its environment but solely on the visual impression of its world. Despite Veneri's success, their system is required to run on six TI-C40 DSP (50MHz) processors, knowing the computational power of a typical insect, surely a simpler solution to memory-based navigation must exist.

7 Conclusions

This paper has presented a low cost (in terms of development cost and development time) real-time navigation network based on insect's navigation principles using a trainable vision system. The results obtained from the above experiments illustrated the feasibility of the proposed paradigm as path learning mechanism. Further more, the proposed navigation network was shown to be robust against displacement error and have a fast learning ability (necessary in real-time operation). Despite the success in the testing of the proposed navigation network, issues such as over-learning are still to be resolved.

References

[Kosaka & Pan 1995] Akio Kosaka, and Juiyao Pan, *Purdue Experiments in Model-Based Vision for Hallway Navigation,* Proceedings of Workshop on Vision for Robots Conference (IROS'95), pp 87-96., 1995.

[Mataric 1990] Maja J. Mataric, *A Distributed Model for Mobile Robot Environment-Learning and Navigation,* MIT Artificial Intelligence Laboratory, Technical Report 1228, 1990.

[Owen & Nehmzow 1996] Carl Owen, Ulrich Nehmzow, *Route Learning in Mobile Robots through Self-Organisation,* 1st Euromicro Workshop on Advanced Mobile Robotics, Kaiserslautern 1996.

[Lambrinos 1995] Dimitrios Lambrinos, *Navigating with an Adaptive Light Compass*, Advances in Artifical Life: Third European Conference on Artificial Life", pp 602-13, 1995.

[Rao & Fuentes 1996] Rajesh P.N. Rao, and Olac Fuentes, Learning Navigational Behaviours using a Predictive Sparse Distributed Memory, Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior, Cape Cod, MA, 1996.

[von Frisch 1971] Karl von Frisch, *Bees : Their Vision, Chemical Sense, and Language,* Cornell University Press, Ithaca and London, 1971.

[Srinivasan et al 1995] M.V. Srinivasan, J.S. Chahl, G. Nagle, K. Weber, S. Venkatesh, and S.W. Zhang, *Low-level vision in insects, and applications to robot navigation*, Computational Intelligence – A Dynamic System Perspective – IEEE press, 1995.

[Veneri et al 1997] N. Veneri, S. Messelodi, and B. Crespi, *A Memory-based approach to sensory-motor coordination*, Machine Vision and Application, 10, pp269-279, 1998.

[Srinivasan 1995] M.V. Srinivasan, *Strategies for Visual Navigation, Target Detection and Camouflage: Inspirations from Insect Vision*, 1995 IEEE International Conference on Neural Networks Proceedings, Vol: 5, pp 2456-60, 1995.

[Collett 1996] T.S. Collett, *Insect Navigation en Route to the Goal : Multiple Strategies for the use of Landmarks*, The Journal of Experimental Biology, 199, pp 227-235, 1996.

[Dyer 1996] F.C. Dyer, *Spatial Memory and Navigation by Honeybees on the Scale of the Foraging Range*, The Journal of Experimental Biology 199, pp 147-154, 1996.

[Cartwright & Collett 1983] B.A. Catwright, and T.S. Collett, *Landmark learning in bees: experiments and models*, Journal of Comp. Physiol. 151, pp 521-543, 1983.

[Edelman & Poggio 1989], *Integrating visual cues for object segmentation and recognition*, Optic News, 15:8-15, May 1989.

[Lehrer 1993] M.Lehrer, *Why do bees turn back and look?*, Journal of Comparative Physiology A 172, pp 549-563, 1993.