Dynamical Interactions between Learning, Visual Attention, and Behavior:
An Experiment with a Vision-Based Mobile Robot

(Published in Fourth European Conference on Artificial Life, pp.309–317, MIT Press. 1997)

Jun Tani, Jun Yamamoto and Hiro Nishi
Sony Computer Science Laboratory Inc.
Takanawa Muse Building, 3-14-13 Higashi-gotanda,
Shinagawa-ku,Tokyo, 141 JAPAN
email: tani@csl.sony.co.jp

Abstract

We investigate how a vision-based robot can learn an analogical model of the environment dynamically through its behavior. We propose a cognitive architecture consisting of multiple neural network modules. The recurrent neural network (RNN) learns the sequence of events encountered incrementally as episodic memories so that the RNN can make prediction based on such sequences in the future. The visual module has two task processes to execute, namely object recognition and wall-following. Attention between these two tasks is switched by means of the topdown prediction made by the RNN. The effect of the topdown prediction to the vision processes is modulated dynamically using the measurement of learning status of the RNN. We have conducted experiments involving learning both static and dynamic environments using a real vision-based mobile robot. It was shown that the robot adapts to the environment in the course of dynamical interactions between its learning, attention and behavioral functions. We show an interpretation of the results from the view of Matsuno's *the internal observer*.

1 Introduction

We speculate that cognitive robots may need to have internal descriptions or analogical models of the world so that they can simulate mentally their own behavioral sequences. In addressing the issues of the description, it is, however, crucial to consider how such a description can be grounded in the physical world and how the mental processes manipulating the description can be situated in the behavioral context[8].

Recently, the dynamical systems approach has been actively studied in the domain of adaptive behavior [3, 19]. We have hypothesized that its language may best represent cognitive aspects of robots and may provide insight into the above problems of the description. Our previous work[22, 21, 23] concerning robot navigation learning showed that an analogical model of the environment can be successfully embedded in the internal dynamical structure of a neural network model through the learning process, and that mental processes, such as look-ahead prediction or planning, can be situated naturally in the behavioral context as coherence is achieved between the internal and the environmental dynamics. The dynamical systems approach enables robots to attain descriptions which are intrinsic to their behavior.

Our experiments, however, were still limited in their scope. Firstly, due to the simplicity of the robot itself and of its environment, the complexity of their interactions was quite limited and therefore the robot's behavioral became highly deterministic and predictable. (The mobile robot had only a simple sensing device consisting of a laser range finder and two motors on the left and right wheels.) Secondly, our experiments were successful only in the case of learning of static environments. The actual learning was conducted in an off-line manner. These limitations may obscure the essential problems of robot cognition. We speculate that the very problems of cognitions commence in the moment when a robot attempts to interact with an unknown environment and tries to extract a certain structure of the behavioral causalities hidden in the non-deterministic sequences from its interaction experiences.

This paper introduces our new project in which we investigate the above problems. We built a new robot for which the primary sensory input was visual images by a video camera. The robot has to control camera orientation, both horizontally and vertically, in addition to maneuvering of its wheels. During navigation, the robot moves avoiding collisions with obstacle walls and simultaneously tries to recognize passing objects using the vi-

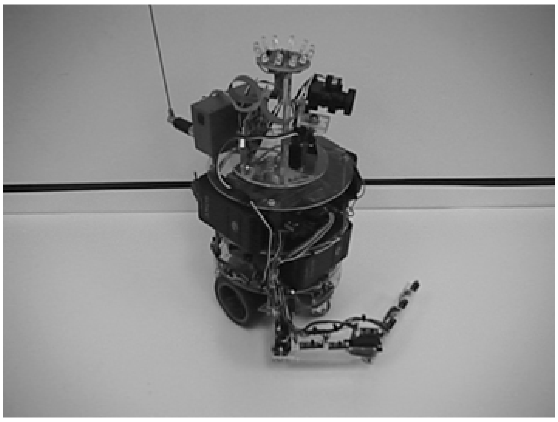

Figure 1: The vision-based mobile robot used in the experiments.

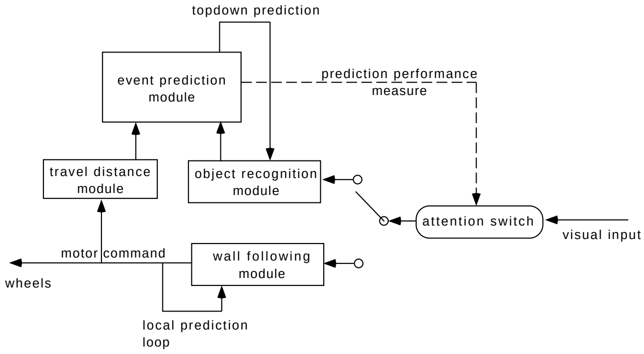

Figure 2: The proposed architecture consisting of multiple modules.

sion. This task is not so simple considering the range of its visual field is physically limited and its recognition process is limited by its real time requirements. The robot has to switch its visual attention dynamically from the wall it is following to the objects it is trying to recognize. The dynamics of its visual attention are observed to affect enormously the ways the robot interacts with its environment.

Our robot attempts to learn incrementally what it experiences and the visual attention dynamics are adapted in real time. A neural network (NN) model consisting of multiple modules learns to categorize the visual images of objects and also learns to predict the sequence of events, such as encountering the objects or the corners of the walls, while the robot moves around the environment. Regarding visual attention, the timing of the attention switching between wall following and object recognition are modulated based on the performance of the NN predictions. Complexity arises when dynamics of the neural learning and the adaptation of the visual attention as well as robot's behavior interact each other. Modulation of the attention dynamics affects the behavior of the robot, which results in further neural learning based on the newly obtained experiences. This learning causes an alteration in the performance of the neural network, which results in further modulations of the attention dynamics.

In the following, we will describe our models and show preliminary results from our experiments.

2 The Robot and Its learning Task

Fig 1. shows our vision-based robot. The robot maneuvers by modulating the rotational speed of its two wheels. The video camera, which is mounted on top of the body, captures color images; the range of its visual field is 60 degrees horizontally and 40 degrees vertically. The camera head can rotate 150 degrees horizontally and 70 degrees vertically. 16 touch sensors are mounted around the body. In conjunction with the torque measured by the wheel motors, these touch sensors can detect collisions with obstacles.

The task of the robot is to learn an analogical model of its environment through its travel. When the robot navigates by following a wall in its environment, it will eventually detect an object or a corner in the wall. The robot learns the sequences of what it sees and how far it travels between one corner and the next corner. Once the robot learns the sequence of such events, it becomes able to predict coming events.

The landmark-based navigation approach has been studied by many other researchers [15, 14]. In those investigations, it was proposed that the topological map of the environment can be represented in the form of a finite state machine (FSM). However, only a few investigations studied qualitatively how the robots behave if the detection of landmarks is nondeterministic or how the learning processes evolve if the environment dynamically changes. (Yamauchi and Beer discussed these problems in their formulation using the so-called Adaptive Place Network [26].) We would like to discuss these problems qualitatively by using the dynamical systems approach.

3 Models

3.1 Overview of the model

Fig 2. shows a schematic diagram of our model. The visual image flows into the attention switch module where its flow is switched either to the wall following module or to the object recognition module based on the visual attention dynamics. The wall following module detects the edge of the nearby wall on the left-hand side of the robot and generates the motor commands for the wheels in order for the robot to follow the wall. This module has a local prediction loop so that it can predict how the perceived edge of the nearby wall changes as the robot moves even when the flow of the visual image is interrupted for some seconds due to the attention switch. The motor commands are also sent to the travel distance module for integration with respect to time in order to determine the travel vector from one corner to the next corner. When the object recognition module identifies an object, it sends a categorical signal to the event prediction module. The event prediction module functions when the robot passes a corner. The module receives a travel vector from the previous encountered corner or a categorical identification of an object which the robot found during its travel from the previous corner. The event prediction module then predicts the next event. The prediction of the next object to be encountered is fed back to the object recognition module in the form of the top-down signal. The recognition of an object involves cooperative dynamics between the bottom-up and top-down processes. A measurement of the prediction performance is sent to the attention switch module in order to modulate its dynamics.

3.2 The visual processes

As we have described briefly in the previous section, the robot has to switch its attention between two visual tasks: wall edge following and object recognition. These two tasks are alternated between during the travel.

First, the camera head turns maximally to the left and focuses on the edge between the wall and the floor. The camera head then turns gradually to the forward direction, following the perceived edge line as foveated in the center of the visual field. The measured trajectories of the head's rotation in the horizontal and vertical directions $(\theta_h(i), \theta_v(i))$ represent the shape of the wall edge. This single movement of the camera head from the extreme left to the forward direction takes about 2 seconds. The current motor commands for the wheels wh^t are determined by a pre-determined mapping with respect to $(\theta_h(i), \theta_v(i))^t$. This mapping function is tailored to ensure that the trajectory the robot travels is smooth and avoids collisions with the walls. Since the relative location of the wall gradually changes as the robot moves, it is necessary to predict how the shape of the edge changes as a function of maneuvering. This is necessary because the visual attention can be switched to the other task for a relatively long period. The prediction is done using a simple forward model [12] implemented on a three-layered perceptron-type NN. A trajectory sampled at time t: $(\theta_h(i), \theta_v(i))^t$ is fed into the input of the forward model in addition to the motor commands for the wheels wh^t. The output: $(\theta_h(i), \theta_v(i))^{t+1}$ is the predicted shape at time $t+1$. Although it is mathematically true that the robot can predict a long time ahead through the recursive usage of the forward model, in practice the accuracy of the prediction decays substantially a few seconds into the future. It is important to note that there is a high risk of collision if the robot travels for more than several seconds relying on this prediction. During the wall following task, corners are detected by means of identifying the shape of the wall edge in addition to the rotation differential between the left and right wheels.

After the camera head turns to the forward direction, it then turns gradually to the right, searching for objects. In our experimental setup, objects are painted with colored patterns; floors and walls are painted grey and white. The search for objects is conducted using the color information. Many researchers [1, 5, 10] in the AI or robotics fields have worked on biologically inspired systems of visual routines and visual searches using color information. We have utilized ideas from their research. In the visual search process, a region consisting of a certain number of color pixels in the visual field "pops-up" [2]. Then, the center of the "pop-up" region is foveated– i.e. the camera head moves so that the region is relocated in the center of the visual field. Van Essen [6] proposed a model of dynamic routing between an attended region in retina and the visual cortical field which is modeled roughly using an associative memory. We used this idea. The attended region of the color image is routed up to the Hopfield [9] type associative memory network where memories of objects are stored as they were learned in the object-centered framework. The Hopfield network consists of 10x10x3 neurons corresponding to the color image of 10x10 pixels. In the routing process, the image of the attended region is scaled so as to match with this pixel size. Three neurons are allocated for each pixel in order to represent its color information. The color of each pixel is categorized into one of three categories in the Hue-Saturation space and only its corresponding neuron is activated. There is an array of winner-take-all neurons which is connected bi-directionariry with the Hopfield network (see Fig 3.). The neurons also receive top-down prediction input from the event prediction module. The strength of this top-down prediction is modulated based on the performance of the prediction module. The recognition proceeds dynamically involving cooperation between these two networks as they receive both the bottom-up visual signals and the top-down prediction signals. The final winner of winner-take-all neurons represents the identified category of the visual image. (The combination of winner-take-all neurons and associative memories has also been studied in the so-called PATON architecture by Omori [18] using a simple numerical analysis.) The dynamics of a neuron in the Hopfield network is given by:

$$\begin{aligned} u_i(t+1) &= \gamma \cdot u_i(t) + k_1 \sum_n w_{i,j}{}^H a_j(t) \qquad (1) \\ &\quad + k_2 \sum_n w_{i,k}{}^w a_k(t) + k_3 \cdot in_i \end{aligned}$$

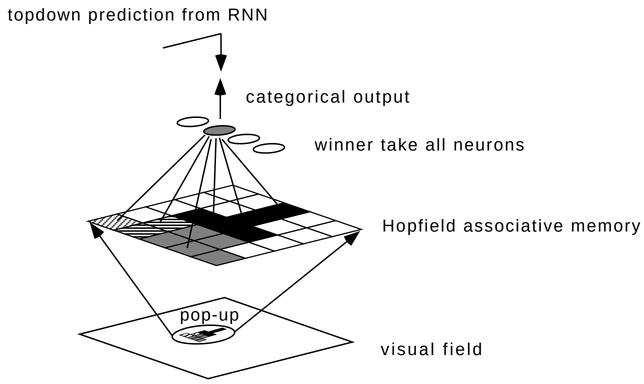

Figure 3: The object recognition module consisting of the visual field, the Hopfield associative memory and the winner-take-all neurons.

$$a_i(t+1) \quad = \quad sigmoid(u_i(t+1)/T)$$

Here, u_i and a_i are the internal state and the activation state of the i-th neuron, respectively, γ is a decay parameter, the ${w_{i,j}}^H$ are the intra-connective weights of the Hopfield network, the ${w_{i,j}}^w$ are the inter-connective weights of the winner-take-all neurons and in_i is the input from the visual field. The dynamics of the winner-take-all neurons are given by:

$$\begin{aligned} u_i(t+1) &= \gamma \cdot u_i(t) + h_1 \sum_n {w_{i,j}}^W a_j(t) \qquad (2) \\ &\quad + h_2 \sum_n {w_{i,k}}^h a_k(t) + \eta \cdot pred_i \\ a_i(t+1) &= sigmoid(u_i(t+1)/T) \end{aligned}$$

Here, the ${w_{i,j}}^W$ are the intra-connective weights of the winner-take-all neurons, the ${w_{i,k}}^h$ are the inter-connective weights from the Hopfield network, $pred_i$ is the input from the top-down prediction and η is a parameter to regulate the strength of this top-down prediction.

The learning takes place after each recognition process (i.e. after the network dynamics are terminated). The intra-connections of the Hopfield network are updated based on Hebb's rule implemented with a constant decay mechanism. The decay is necessary to prevent the weights from diverging in the process of incremental learning. The learning rule is:

$$\Delta w_{i,j} = -\zeta \cdot w_{i,j} + \epsilon(a_i - 0.5)(a_j - 0.5) \cdot 4.0 \qquad (3)$$

Here, ζ is a decay parameter. For updating the inter-connective weights between the Hopfield network and the winner-take-all neurons, only the winning neuron is set as being activated to 1.0; the others are set as being deactivated to 0.0. Following this process, the same Hebb's learning rule is applied.

3.3 Attention and self-referential processes

The problem with the visual attention arises because the visual process is resource-bounded in both time and space [1]. If our robot spends a longer time in recognizing objects, there is a high risk of collision. On the other hand, if the robot spends a shorter time on the recognition process, the identification results may be in error as the answer is required before the NN dynamics converge. Clearly, a good strategy for determining the timing of the attention switch is required. The time required for convergence depends on the learning status of the NN modules. In the early stages of learning, the attractor of the Hopfield network is shallow and the top-down prediction is inaccurate. Therefore, the Hopfield network dynamics take a long time to converge. In such cases, they likely oscillate because contradictions between the top-down prediction and the bottom-up signals. On the other hand, when the learning converges, the top-down prediction and the bottom-up signals agree quite well, which cause the Hopfield network dynamics to converges rapidly. For this reason, the performance of the prediction module is monitored so that the current learning status can be used to determine when to terminate the iteration of the network dynamics and also to evaluate the validity of the topdown prediction. Here, $steps^{max}$, the maximum steps allowed for iterations of the Hopfield network, is defined by

$$steps^{max} = l_0 + l_1 \cdot error^{pred} \qquad (4)$$

where $error^{pred}$ is the current prediction error measure (PEM) of the prediction module (an average of the prediction error among the previous 5 predictions); l_0 and l_1 are constants. In addition to this adaptation strategy, η is defined by

$$\eta = \eta_0 \cdot (1.0 - error^{pred}) \qquad (5)$$

where η_0 is a constant. This equation implies that the validity of the top-down prediction increased as the predictability by means of learning is improved.

What we have proposed here is a modeling of self-referential processes in which the robot can be aware of the validity of its own mental processes which is fed back to the attention processes in an unconscious way. These models of the adaptation process of visual attention are also based on the ideas of Koch and Crick [4] in the physiologicical side. They have hypothesized that the neurons in the visual cortical areas whose responses are changed by attentions are the ones that receive inputs from the prefrontal cortex.

3.4 Prediction by recurrent neural net

The event prediction module is implemented with a standard recurrent neural network (RNN) [19, 11] as shown

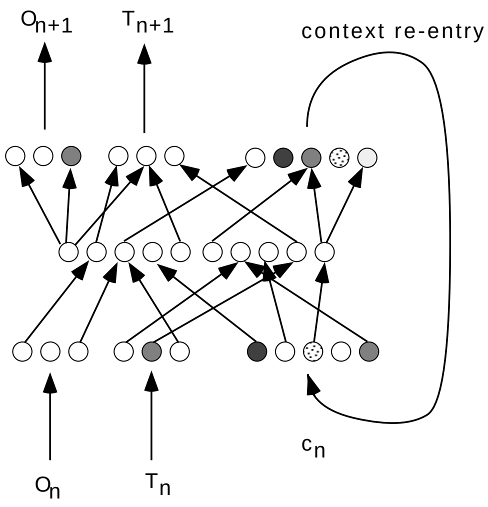

Figure 4: The RNN implemented for the event prediction module.

in Fig 4. The RNN may correspond to the prefrontal cortex, as a number of studies have suggested that the prefrontal cortex performs the function of a working memory or of planning events (see Ref. [7].) The RNN receives input from two different sensory sources. One is a visual image of colored objects; the other is the travel vector from one corner to the next corner. This part of the modeling is based on the well-known fact about "where and what pathways" [24] of visual processing in the human brain. The RNN does not receive direct sensory images of those, but receives categorical signals for them instead. The visual image is categorized by the combination of the Hopfield network and the winner-take-all neurons, as we have described above. The travel vector is categorized using the standard Kohonen network [13] in the travel distance module. The travel vector is entered into the Kohonen network, after which the winner neuron represents the category of the travel vector. The Kohonen net is self-organized in an on-line manner. The output of the RNN is the prediction of the categorical signals for the two sensory sources. In the figure, T_n represents the travel vector category and O_n represents the object category. The RNN process in this figure is one in which the travel vector from the previous corner to the current corner is identified as being in the second category, resulting in a prediction that an object of the third category will be encountered at the next. We employ Jordan's idea of context re-entry which enables the network to represent the internal memory [12]. The current context input c_n (a vector) is a copy of the previous context output: by this means the context units remember the previous internal state. The navigation problem is an example of a so-called "hidden state problem" in that a given sensory input does not always correspond to a unique situation or position of the robot. Therefore, the current situation or position is identifiable, not by the current sensory input, but by the memory of the sensory sequences stored during travel. Such a memory structure is self-organized through the learning process. The context self-organized in these units is likely to have a rather distributed fuzzy representation. The RNN used in our experiment has 9 input nodes, 9 output nodes, 25 context nodes and 25 hidden units.

3.5 Incremental learning and consolidation process

It is difficult for RNNs to learn the received information incrementally. It is generally observed that the contents of the current memory are severely damaged if the RNN attempts to learn a new teaching sequence. One way to avoid this problem is to save all the past teaching data in a database. When new data is received, it is added to the former date in the database, and all the data is then used to re-train the network. Although this procedure may work well, it is not biologically plausible.

Observations in biology show that some animals and humans may use the hippocampus for a temporary storage of episodic memories [20]. Some theories of memory consolidation postulate that the episodic memories stored in the hippocampus are transferred into some regions of the neocortical systems during sleep. Recent experiments [25] on the hippocampal place cells of rats show evidence that those cells reinstate the information acquired during daytime active behavior. McClelland [17] further assumes that the hippocampus is involved in the reinstatement of the neocortical patterns in long term memory and that the hippocampus plays a teaching role in training the neocortical systems.

We apply these hypotheses to our model of RNN learning. In our system, an experienced sequence of events, which may correspond to a temporary episodic memory, are stored in the hippocampal database. In the consolidation process, the RNN which corresponds to the prefrontal cortex rehearses the stored memory patterns. The rehearsal can be done by recursively activating the RNN using the closed feedback loop from the outputs of the sensory prediction to the sensory inputs. The generated sequential patterns are sent to the hippocampal database. The RNN can be trained using both the rehearsed sequential patterns and the newly experienced ones. In our experiment, the robot stores up to 14 steps of previously encountered events in the hippocampal database. In the consolidation process, the RNN rehearses for 14 iterations to generate a sequence, then 28 steps of the sequential patterns in total are used to re-train the RNN. The re-training of the RNN is conducted by updating the connective weights obtained in the previous training.

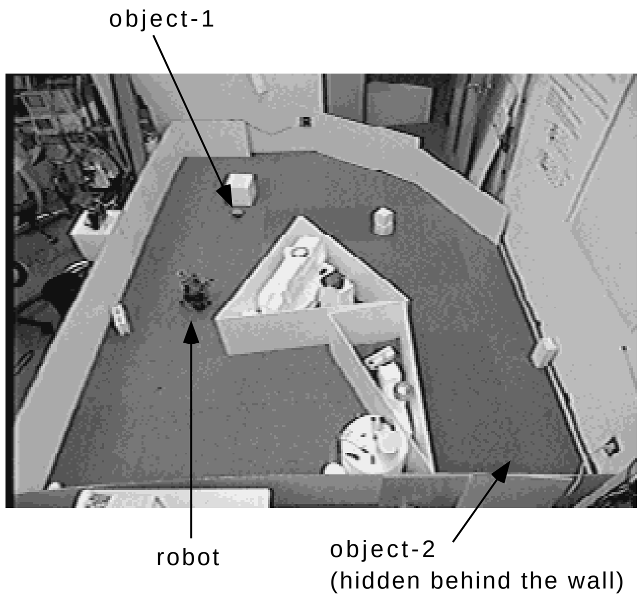

Figure 5: Workspace adopted for the robot learning experiments.

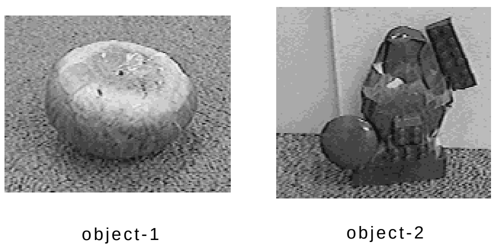

Figure 6: The two objects placed in the workspace which are painted with different color patterns.

4 Experiments

The learning experiments were conducted in the workspace shown in Fig 5., in which two different shapes of objects were located (see Fig 6).

The experiments were conducted in two successive phases; these were learning in the original environment followed by learning in a modified environment.

4.1 Adaptation to the original environment

We will now describe the results for the case of learning in the original environment. Fig 7. shows the observed history of the prediction error measurement (PEM) at each event step in the learning phase. The learning of the RNN is initiated after the RNN experienced 14 steps of the event sequence. During the first period of learning, the PEM gradually decreases. The PEM almost converges in the second period of learning which starts at the 28th step.

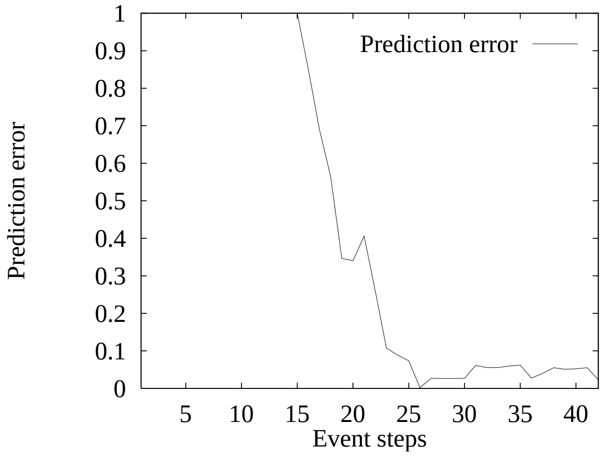

Figure 7: The history of the prediction error as measured for learning in the original environment.

Fig 8 shows the prediction sequence, its actual outcome and the associated activation pattern of context units for each step. The steps proceed upwardly in this figure. The number shown to left denotes the event step; the two adjacent rows show the prediction of the sensory category, where the upper row represents the five categories of the visual image and the lower row the four categories of the travel vector. Black squares represent activated categories and their strength is indicated by their size. The next two rows to the right indicate the actual sensory categorical inputs. The upper and the lower rows represent the visual image and the travel vector, respectively. The large square to the right shows the activation pattern of the 25 context units. Fig 8 (a) shows the sequence obtained during the first period of the learning and Fig 8. (b) corresponds to the second period of learning. Fig 8. shows that the prediction fails frequently in the earliest stage, from the 15th to the 21st step. Prediction is, however, improved during the second period of learning, as is also seen also in the history of PEM in Fig 7. We observe a stable periodicity of six steps in the sequence of Fig 8 (b), but, we do not observe such a periodicity in the earlier stage of Fig 8 (a). We examined the dynamical structure of the RNN obtained at the end of the learning process. The RNN was activated recursively by closing the open loop from the sensory prediction to the sensory inputs. We are confident that we have identified an attractor of the limit cycling with the periodicity six in the phase space of the RNN.

We will now illustrate how the visual attention dynamics interact with the behavior of the robot. In Fig 9, we compare two trajectories of robot's travel, one from the

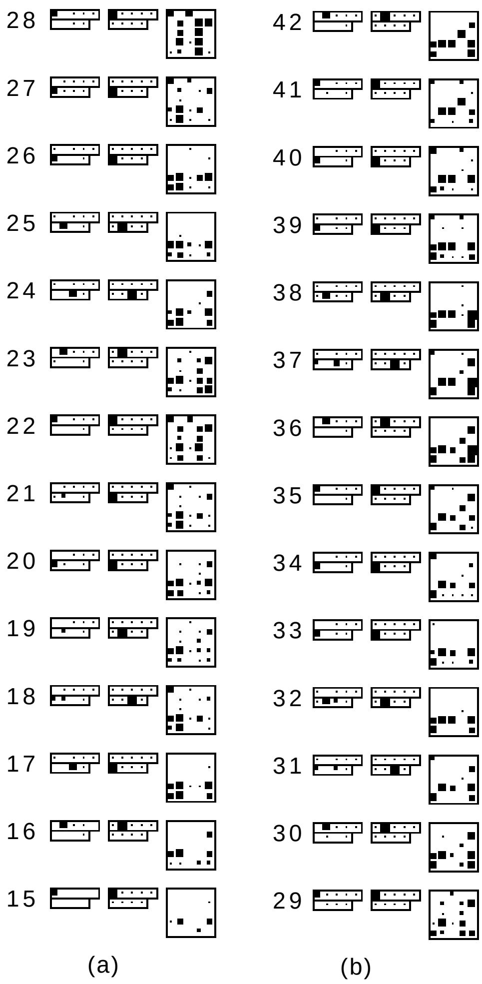

Figure 8: The sequence of prediction, sensory inputs, and context activation pattern of the RNN in the first period of the learning (a) and that of the second period (b).

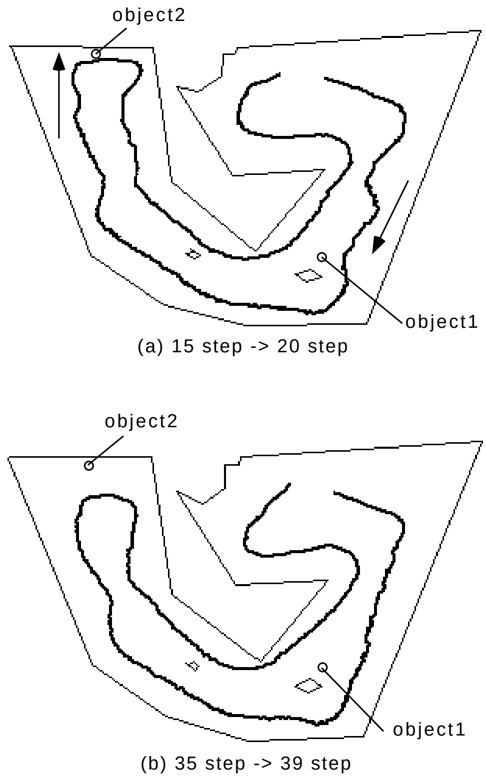

Figure 9: Comparison of the robot trajectories corresponding to two different learning statuses. (a) from the first period of learning shows a more winding trajectory than (b) in the second period.

15th to the 20th step in the first period of learning and the other from 35th to 39th step in the second period of learning; these are shown in Fig 9 (a) and (b) respectively. It is seen that the trajectory winds more in (a) than in (b) especially in the way objects are approached. We infer that the maneuvering of the robot became more unstable because the robot spent a greater time on the visual recognition of objects in the first period of learning due to the higher value of the PEM. Therefore, the robot took a higher risk of the mis-detection of events as its trajectory oscillate. In Fig 9 (a) we note that the robot mis-detected a corner immediately after its recognition of object 2 causing it to take for a while until its prediction to recover. Such a nondeterministic phenomenon in the detection of events affects the RNN's learning. In Fig 8., it is frequently seen that the RNN attempts to predict two categories at the same time. The previous experience of a nondeterministic phenomenon in the sequence of experiences caused the generation of such expressions by the RNN.

When the robot happened to predict the correct sequence for some steps, the PEM as well as the time required for the visual recognition were observed to de-

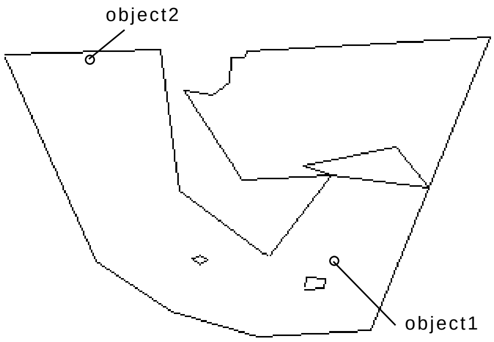

Figure 10: The modified workspace.

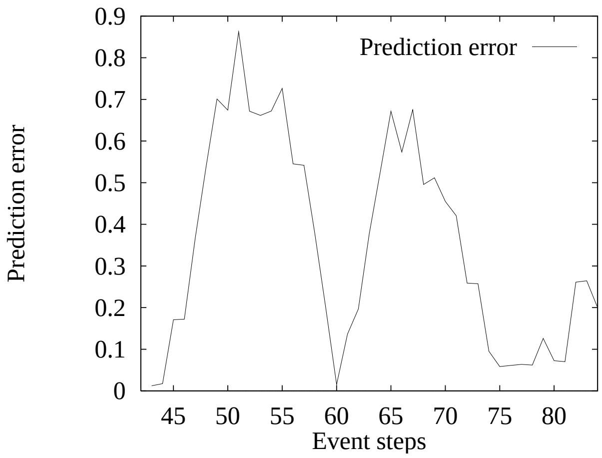

Figure 11: The history of the PEM obtained during learning in the modified environment.

crease. Thereafter a stable regime emerged in which a quasi-coherence was achieved between the dynamics of learning, attention, and behavior. However we speculate that this regime is only marginally stable, as we have observed that the regime could be disrupted by certain catastrophic changes even after a long period of stability. We need to conduct further experiments to investigate more carefully the stability criteria.

4.2 Re-adaptation to a modified environment

After performing the previous experiment, we modified the workspace partially and restarted the robot. The conditions such as the NN weights and the attention parameters were retained from the previous experiment. The geometry of the modified workspace is shown in Fig 10. Fig 11. shows the associated history of the PEM. We observed that the PEM increased when the robot traveled towards the modified region, but the PEM decreased when the robot traveled otherwise in the unmodified region. This PEM increase decayed as the incremental learning was proceeded. We infer that the former memories were preserved to some extent, but their part in contradiction with the modified environment was gradually altered as a result of the new experiences. However, we need to wait for further experiments to be performed before we can confirm that the observed characteristics are more general.

5 Discussion

We observed dynamical interactions taking place between learning, attention, and behavior, which might be one of crucial points needing to be considered when building cognitive robots. It is important to note that when a robot observes the world, such observations inevitably lead to actions of the robot which change the original relation between the robot and the world. This effect was well illustrated in our experiments, which showed that visual attention affects the maneuvering trajectory. According to Matsuno, the observer is included in the internal loop of actions: the observer is an *internal observer* [16]. The internal observer never maintains descriptions as completely static properties, but instead iteratively generates new descriptions, as the interactions proceed between the observer and the environment.

Someone may ask if there exist any phisical entities which correspond to *the internal observer* in animals or animats. Fact is that all there exist are only dynamical structure in which no separable entities of descriptions and observers are seen.

References

[1] P.E. Agre and D. Chapman. Pengi: an implementation of theory of activity. In *Proc. of the Sixth National Conf. of Aritificial Intelligence*, pp. 268–272, 1987.

[2] A.M.Treisman and G.Gelade. A feature integration theory of attention. *Cognitive Psychology*, Vol. 12, pp. 97–136, 1980.

[3] R.D. Beer. A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, Vol. 72, No. 1, pp. 173–215, 1995.

[4] F. Crick and C. Koch. The problem of consciousness. *Scientific American*, Vol. 267, pp. 152–158, 1992.

[5] D.H.Ballard. Animate vision. *Artificial Intelligence*, Vol. 48, pp. 57–86, 1991.

[6] D.C. Van Essen, C.H. Anderson, and B.A. Olshausen. *Dynamic routing strategies in sensory motor, and cognitive processing.* Cambridge, MA: MIT press, 1994.

[7] S.E. Gathercole. Neuropsychology and working memory: a review. *Neuropsychology*, Vol. 8, pp. 494–505, 1994.

[8] S. Harnad. The symbol grounding problem. *Physica D*, Vol. 42, pp. 335–346, 1990.

[9] J.J. Hopfield and D.W. Tank. Neural computation of decision in optimization problems. *Biological Cybernetics*, Vol. 52, pp. 141–152, 1985.

[10] I. Horswill. Visual routines and visual search: a real time implementation and an automata-theoretic analysis. In *Proc. of the Int. Joint National Conf. of Aritificial Intelligence, (IJCAI'95)*, pp. 56–61, 1995.

[11] M.I. Jordan. Attractor dynamics and parallelism in a connectionist sequential machine. In *Proc. of Eighth Annual Conference of Cognitive Science Society*, pp. 531–546. Hillsdale, NJ: Erlbaum, 1986.

[12] M.I. Jordan and D.E. Rumelhart. Forward models: supervised learning with a distal teacher. *Cognitive Science*, Vol. 16, pp. 307–354, 1992.

[13] T. Kohonen. Self-organized formation of topographically correct feature maps. *Biological Cybernetics*, Vol. 43, pp. 59–69, 1982.

[14] B. Kuipers. A qualitative approach to robot exploration and map learning. In *AAAI Workshop Spatial Reasoning and Multi-Sensor Fusion (Chicago)*, pp. 774–779, 1987.

[15] M. Mataric. Integration of representation into goal-driven behavior-based robot. *IEEE Trans. Robotics and Automation*, Vol. 8, No. 3, pp. 304–312, 1992.

[16] K. Matsuno. *Physical Basis of Biology.* CRC Press, Boca Raton, FL., 1989.

[17] J.L. McClelland, B.L. McNaughton, and R. O'Reilly. Why there are complementary learning systems in the Hippocampus and Neocortex. Technical Report PDO.CNS.94.1, Carnegie Mellon University, 1994.

[18] T. Omori and A. Mochizuki. PATON: A Model of Context Dependent Memory Access with an Attention Mechanism. In *Proc. of Int. Conf. on Brain Ppocesses Theories and Models*, 1995.

[19] J.B. Pollack. The induction of dynamical recognizers. *Machine Learning*, Vol. 7, pp. 227–252, 1991.

[20] L.R. Squire, N.J. Cohen, and L. Nadel. The medial temporal region and memory consolidation: A new hypothesis. In H. Weingartner and E. Parker, editors, *Memory consolidation*, pp. 185–210. Erlbaum, Hillsdale, N.J., 1984.

[21] J. Tani. Essential Dynamical Structure in a Learnable Autonomous Robot. In *Proc. of the Third European Conf. of Artificial Life (ECAL'95)*, 1995.

[22] J. Tani. Model-Based Learning for Mobile Robot Navigation from the Dynamical Systems Perspective. *IEEE Trans. System, Man and Cybernetics Part B, Special issue on learning autonomous robots*, Vol. 26, No. 3, pp. 421–436, 1996.

[23] J. Tani and N. Fukumura. Self-organizing internal representation in learning of navigation: a physical experiment by the mobile robot YAMABICO. *Neural Networks*, Vol. 10, No. 1,, 1997.

[24] L.G. Ungerleider and M. Mishkin. Two cortical visual systems. In D.G. Ingle, M.A. Goodale, and R.J. Mansfield, editors, *Analysis of Visual Behavior.* Cambridge, MA: MIT Press, 1982.

[25] M. Wilson. Reactivation of hippocampal ensemble memories during sleep. *Science*, Vol. 265, pp. 676–679, 1994.

[26] B. M. Yamauchi and R. D. Beer. Spatial learning for navigation in dynamic environment. *IEEE Trans. Syst. Man Cybern.*, Vol. 26, No. 3,, 1996.