A Vision-driven Model of Hippocampal Place Cells and Temporally Asymmetric LTP-induction for Action Learning

Angelo Arleo, Wulfram Gerstner
MANTRA, Centre for Neuro-Mimetic Systems, EPFL
Swiss Federal Institute of Technology, CH–1015, Lausanne, Switzerland
Email:{Angelo.Arleo, Wulfram.Gerstner}@epfl.ch

Abstract

We describe a hippocampal neural model in which spatio-temporal features of the environment are extracted by visually driven neurons. The neuronal firing activity implicitly measures properties like agent-landmark distance and egocentric orientation to visual cues. This leads to a neural representation where populations of place cells encode spatial locations within the environment. In addition, temporally asymmetric long-term potentiation of synapses between place cells is used to learn a vector field representation providing a navigational map. We present experimental results obtained by testing our model with the mobile Khepera robot.

1 Introduction

Navigation of perceptually complex animals relies on (more or less explicit) spatial representation. The *Hippocampus* is a much-studied example of a neurophysiological structure with such a spatial representation property. Based on experimental evidence for spatially-tuned neurons (*place cells*) in rat hippocampus [14], it has been suggested that this area of the brain is involved in mammalian navigation.

The first order correlate of place cell activity is the location of the rat within the environment. Each place cell fires maximally only when the rat is in a specific region, which defines the *place field* of the cell. Place fields are determined by combination of environmental cues, whose mutual relationships code for the current animal location [14]. Experiments on rats show that visual landmarks are mainly responsible for place identification [10]. However, place cells can maintain stable receptive fields even in absence of reliable orienting cues [13]. This suggests a more complex architecture where multimodal information is used for learning and maintaining hippocampal place fields. For instance, in the dark, proprioceptive information could partially replace external stimuli.

We present a hippocampal model that relies on this idea of a *sensor-fusion* process for driving place cell activity. Receptive fields are learned by extracting spatio-temporal properties of the environment from visual sensory inputs. Incoming visual stimuli are interpreted by means of neurons that only respond to combinations of specific visual patterns. Using such a filtering process, properties like agent-landmark distance and egocentric orientation to visual cues may be measured implicitly, without image processing. The activity of the neural filters propagates through the model yielding place cell activity. Unsupervised Hebbian learning is used to build the hippocampal neural structure incrementally. In order to interpret the ensemble place cell activity as spatial locations, we use *population vector coding*.

In addition to visual input we also consider proprioceptive information. An extra-hippocampal path integrator drives Gaussian-tuned neurons modeling the internal movement-related information. During the agent-environment interaction, synapses between visually driven place cells and path-integration neurons are established by means of Hebbian learning. This allows us to correlate internal and external stimuli to obtain a more stable neural representation.

In order to accomplish their functional role in navigation, hippocampal neural mod-

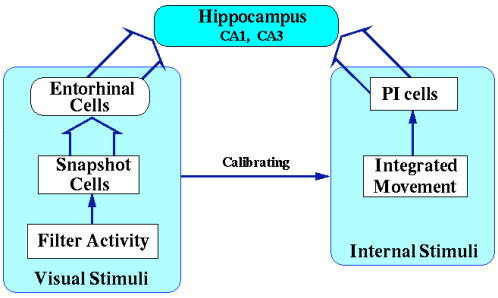

Figure 1: Architectural overview of our hippocampal neural model.

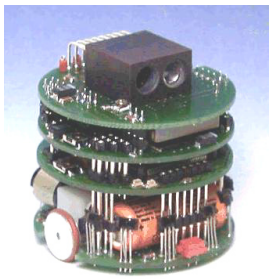

Figure 2: The mobile robot Khepera equipped with the linear-vision turret.

els must incorporate knowledge about relationships between the environment, its obstacles and specific target locations. Given the hippocampal representation, we derive a *navigational map* by applying the approach proposed by Blum, Abbott, and Gerstner [8, 1]. This model relies on two principal concepts: *(i) population vector coding* of spatial location by large numbers of hippocampal cells [16], *(ii)* use of *temporal asymmetry in long-term potentiation* to learn recurrent synapses between place cells [5].

Our approach is similar in spirit to earlier work by Burgess, Recce, and O'Keefe [4], Mallot et al. [12], Trullier and Meyer [15], Brown and Sharp [3].

2 The Hippocampal Model

Our model, Fig. 1, integrates external and internal stimuli to build and maintain stable place fields (modeling CA1 and CA3 hippocampal place cells). In the hardware implementation on a Khepera robot, the vision system is a 64-pixel linear camera (Fig. 2). Internal movement-related information is provided by dead-reckoning. A simulated compass system provides the robot with allocentric orientation information. Finally, infrared sensors endow the robot with obstacle detection capability.

2.1 Learning Place Fields

Relationships between visual landmarks are interpreted by mapping images into a *filter-activity space*. We define a set of classes of *Walsh*-like filters [2]. Each class corresponds to a visual pattern. The set of filters in that class corresponds to different frequencies for that pattern. In total we have 5 different classes each containing filters at 10 different frequencies.

Let F_k be one our *Walsh* filters, where k is the index of the filter, and let l_k be its length. The response a_k of filter k to an input $\mathbf{x} = (x_1, \ldots, x_{64})$ is given by convolution

$$a_k = \max_n \Big\{ \sum_{i=0}^{l_k-1} F_k(i)\, x_{n+i} \Big\} \qquad (1)$$

where $0 \le n \le 64 - l_k$. Since $|x_j| \le 1$, we find that $|a_k| \le l_k$.

Snapshot Cells. The idea is to represent each image by the cluster of filters having the highest correlation value, defined by Eq. 1. We call the set of active filters

$$sc = \{F_k \mid a_k \ge C_k\} \qquad (2)$$

a *snapshot cell*. Here, $C_k = 0.7 \cdot l_k$ is the threshold above which a filter is considered as active.

For each visited location, the robot takes four snapshots corresponding to the north, east, south, and west views. Thus, each location in the environment is characterized by four snapshot cells, forming a local view. During exploration, the robot maintains a database of snapshot cells, whose activity depends on the current view. The firing activity of a snapshot cell j is given by

$$r_j = \frac{\sum_{k_j} \mathcal{H}(a_{k_j} - C_k)}{N_j} \qquad (3)$$

where $\sum_{k_j}$ sums over all the N_j filters forming the cell j, and $\mathcal{H}$ is the Heaviside function. The normalization has been chosen so that $0 \le r_j \le 1$. For instance, if a snapshot cell consists of a cluster of ten filters and only five of them are maximally activated by the current view, the resulting firing level will be 50% of maximum cell activity.

Entorhinal Cells. The snapshot cell activity depends on the current view, but does not code for spatial locations. In order to obtain such a spatial discrimination property, we apply unsupervised learning to create a population of cells one synapse downstream of the snapshot layer (Fig. 1). We

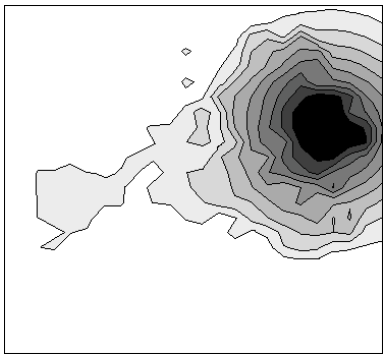

Figure 3: An example of place field in the layer CA1-CA3 of our model. The darker a region, the higher the cell firing rate when the robot is in that region of the environment.

call such units *entorhinal cells* because *(i)* neurons in the entorhinal cortex are the main cortical input to the rat hippocampus, and *(ii)* observed entorhinal cells show spatially correlated firing, but tend to have place fields less defined than those in the CA3 or CA1 hippocampal regions [4].

Every time the robot is in a new location, all simultaneously active snapshot cells are connected to a newly created entorhinal cell. Each new synapse is given a random weight in $(0, 1)$. Let i and j be indexes for entorhinal cells and snapshot cells, respectively. If r_j is the firing activity of a snapshot cell j, then

$$w_{ij}^{new} = \mathcal{H}(r_j - \epsilon)\, \mathrm{rnd}_{0,1} \qquad (4)$$

where $\epsilon = 0.75$ is the activity threshold above which a neuron is considered maximally firing, and $\mathrm{rnd}_{0,1}$ is a random value in $(0, 1)$. The firing rate of an entorhinal cell i is given by the average activity of its presynaptic neurons

$$r_i = \frac{\sum_j w_{ij} r_j}{\sum_j w_{ij}} \qquad (5)$$

Once synapses are established, their efficacy is changed according to a Hebbian learning rule

$$\Delta w_{ij} = r_j\ (r_i - w_{ij}) \qquad (6)$$

For a given location, new connections from snapshot cells to entorhinal cells are not created if

$$\sum_i \mathcal{H}(r_i - \epsilon) > A \qquad (7)$$

that is, if the number of maximally firing entorhinal cells at that location exceeds a threshold $A = 10$ cells. This simple technique allows us to control redundancy in the resulting neural representation. We call such a learning scheme an *unsupervised growing network* (see, e.g., [6]).

Place Cells. Entorhinal cells project to the layer of hippocampal place cells (Fig. 1). As learning proceeds, new downstream synapses are incrementally created by means of the above *unsupervised growing network* scheme. Simultaneously active entorhinal cells are connected to create new place cells. If now i and j represent place cells and entorhinal cells, respectively, synapses are created according to Eq. 4 and changed on-line by Hebbian learning (Eq. 6). The firing rate of each place cell is a weighted average of its presynaptic cells (Eq. 5).

2.2 Path Integrator

Place cells of the rat hippocampus continue to show stable fields even in the dark [13]. In order to compensate for unreliable visual data using internally generated information, we define Gaussian-tuned neurons driven by the *path integrator* system. These hypothetical extra-hippocampal units are used to model movement-integrated information by means of neural activity. Every time the robot moves, the activity of these path-integration cells changes according to the current orientation and velocity of the robot. During the agent-environment interaction, *one-shot* Hebbian learning is used to learn synapses between visually driven place cells and path-integration neurons. This means that new connections are given weight equal to 1, and are not changed any further.

As a consequence, place cell activity depends on the activity of both entorhinal and path-integration cells. This combination of internal and external stimuli yields a more stable neural spatial representation. On the one hand, unreliable visual data are compensated by means of the path integrator neural activity. On the other hand, reliable visual information is used for re-calibrating the path integrator system (Section 2.4).

Fig. 3 shows a typical place field of a neuron in the layer CA1-CA3 of our architecture.

2.3 Population Vector Coding

The proposed hippocampal model results in a spatial representation consisting of a large number of neurons with overlapping place fields. Fig. 4 shows the experimental

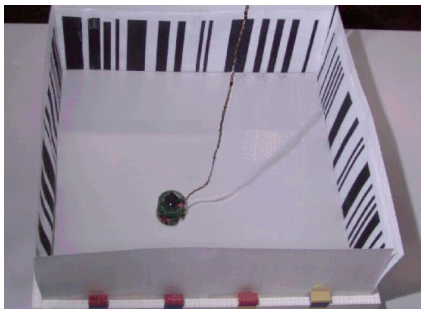

Figure 4: The 60×60 cm testing environment and the Khepera robot within it.

setup: a Khepera robot equipped with a linear vision system (Fig. 2) within a 60 $\times$ 60 cm square arena. Walls are covered by a fixed random sequence of black and white stripes of variable width. Combinations of these stripes form the visual input patterns for the system. Fig. 5 shows the population of CA1-CA3 place cells created by the robot after a learning session of about 2000 time steps. Each dot represents a place cell, and its position represents the center of the cell's place field. After learning, place cells cover the environment uniformly and densely. In Fig. 5 we show the ensemble network activity modeling the robot location of Fig. 4.

In order to interpret the information represented by the ensemble pattern of activity, we use a *population vector decoding* scheme [7]. This approach consists of averaging the activity of the neural population to yield the encoded spatial location. If r_i is the firing activity of a neuron i and $\mathbf{x}_i$ is the location associated with the neuron i during exploration (see below), the population vector $\mathbf{p}$ is the center of mass of the network activity:

$$\mathbf{p} = \frac{\sum_i \mathbf{x}_i \, r_i}{\sum_i r_i} \qquad (8)$$

Notice that the encoded spatial position is near, but not necessarily identical to, the true robot's location $\mathbf{x}$, that is $\mathbf{p} \approx \mathbf{x}$. In Fig. 5 the center of mass encoding the robot location is represented by the white cross.

2.4 Exploration

We use an exploration strategy which emulates the exploratory behavior of animals. The robot starts exploration from an initial location (e.g., the nest location) and, as exploration proceeds, it creates new place cells. Associated with each place cell i is a vector $\mathbf{x}_i$ which represents the location where the robot thinks it is (based on odometry) when it creates the cell i.

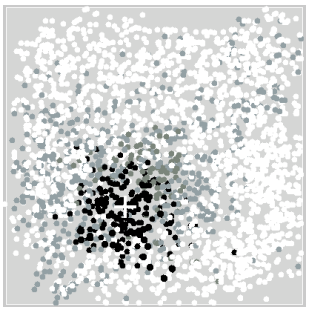

Figure 5: The learned population of place cells. The ensemble activity corresponds to the robot position shown in Fig. 4. The darker a cell, the higher its firing rate. The white cross represents the center of mass of the population activity.

After a while, the robot feels the need to calibrate its path integrator system. It stops creating place cells and starts following the homing vector to return to the starting point. As soon as it finds a previously visited location, it tries to use the learned spatial model to localize itself. We consider the entorhinal cell activity (i.e., visually driven neurons) to perform such a calibrating process. Let σ be the variance of the entorhinal activity around the center of mass $\mathbf{p}_{ec}$. If σ is smaller than a fixed threshold T, then the spatial location $\mathbf{p}_{ec}$ is used to re-calibrate the robot (Fig. 6). More precisely, we define a weight coefficient

$$\alpha = \begin{cases} 1 - \frac{\sigma}{T} & \sigma \leq T \\ 0 & \text{otherwise} \end{cases} \qquad (9)$$

and then we use it to compute the calibrated robot position $\mathbf{p}^*$

$$\mathbf{p}^* = \alpha \, \mathbf{p}_{ec} + (1 - \alpha) \, \mathbf{p}_{dr} \qquad (10)$$

where $\mathbf{p}_{dr}$ is the position estimated by the dead reckoning system.

Once the robot has calibrated itself, it stops following the homing vector and resumes exploring. This technique keeps the odometry error within a bounded range. Fig. 7 shows calibrated versus uncalibrated odometry errors during exploration.

3 Learning Navigational Maps

In this section we model how the long-term potentiation of recurrent synapses between place cells (in particular those in the hippocampal CA3 region) affects the ensemble activity and yields a shifted encoded location. If learning involves a specific target position, this experience-induced shift provides a *navigational map* leading the agent

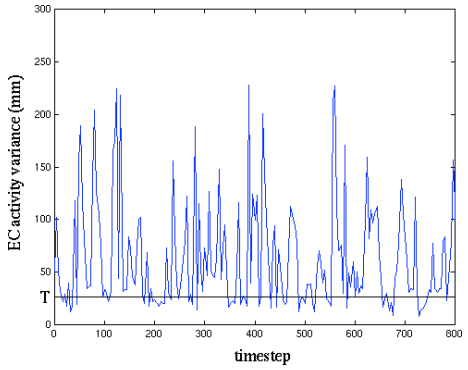

Figure 6: The variance of the entorhinal cell activity around the center of mass $\mathbf{p}_{ec}$. When the variance goes below the fixed threshold T the spatial location $\mathbf{p}_{ec}$ is used to calibrate the robot position.

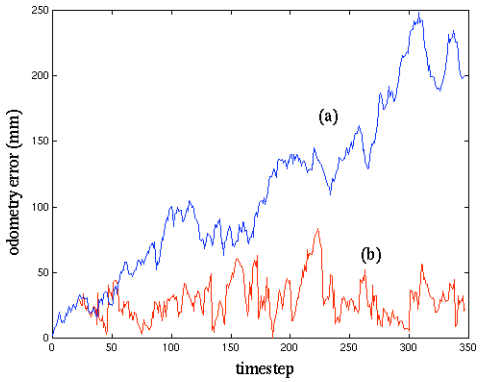

Figure 7: Uncalibrated dead reckoning error (a) versus calibrated robot positioning (b) using entorhinal cell activity.

toward the target from any position in the environment [1, 8].

All place cells are coupled by lateral synapses (modeling collateral synapses between CA3 and CA1 neurons) that are adjusted during a training period. Initially, the strength W_{ij} of the connection from presynaptic neuron j to postsynaptic neuron i is zero. We model synaptic modifications according to experimental results on *temporally asymmetric long-term potentiation*: synapses are strengthened (LTP) if presynaptic activity occurs either before or simultaneously with postsynaptic activity, otherwise they are depressed (LTD) [11, 5]. We model this temporal aspect of synaptic plasticity by a time-window function $H(t)$ of the form

$$H(t) = \begin{cases} \tau^{-1} \exp(-t/\tau) & t \geq 0 \\ -\beta\, \tau^{-1} \exp(t/\tau) & t < 0 \end{cases} \qquad (11)$$

where τ is a time constant and the parameter β varies the relative importance of long-term depression: for $\beta = 0$ there is no LTD at all, for $\beta = 1$ contributions of LTP and LTD are of equal magnitude. As a consequence, if presynaptic neuron j fires at a time t' and postsynaptic neuron i fires at time t, the change in the synaptic efficacy ΔW_{ij} is proportional to a factor $H(t - t')$.

After a learning period of duration T, consisting of a set of exploratory paths leading to a target location, the resulting synaptic strength W_{ij} of the connection from neuron j to neuron i is given by [9, 1]

$$W_{ij} = \int_0^T \int_0^T r_i(t)\, H(t - t')\, r_j(t')\, dt\, dt' \qquad (12)$$

The t-integral sums over the whole training time and, for a given t, the t'-integral sums over all possible time differences between pre- and postsynaptic firing activities.

Because of the experience-based coupling (Eq. 12) established during learning, the activity r'_i of the place cell i is influenced by the activity of other neurons, and it is given by

$$r'_i = r_i + \sum_j W_{ij}\, r_j \qquad (13)$$

The second term is responsible for the experience-induced shift in the spatial information encoded by the neural population. As a consequence, the ensemble network activity no longer encodes the agent's actual position $\mathbf{x}$ as in Eq. 8, but a different location nearby. If $\Delta\mathbf{p}$ is the experience-induced shift, and $\{X\}$ represents the family of exploratory paths taken during the training process, then the population vector after learning is given by

$$\mathbf{p}(\mathbf{x}; \{X\}) = \mathbf{x} + \Delta\mathbf{p}(\mathbf{x}; \{X\}) \qquad (14)$$

It turns out [8] that the shift $\Delta\mathbf{p}$ provides the direction that the animal must follow to reach the target location.

Fig. 8 and Fig. 9 show some experimental results obtained by applying the above technique to derive a navigational map from the learned spatial model (Fig. 5). During training the robot moves along straight exploratory paths avoiding collisions with walls and obstacles. Every exploratory path starts at a random position and with a random orientation and ends when the target is reached. When the robot reaches the target, learning continues for about 100 time steps, while the robot is sitting at the target location. The dashed path shown in Fig. 8 is an example of a training path. Fig. 9 shows the navigational map obtained after 10 training paths. We used a time-window function as

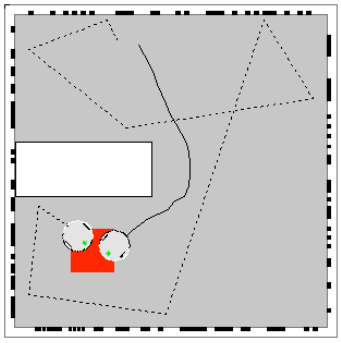

Figure 8: The dashed line is an example of training path leading the robot to the target (dark square). The solid line is an example of trajectory induced by the learned navigational map of Fig. 9. The white rectangle is an obstacle.

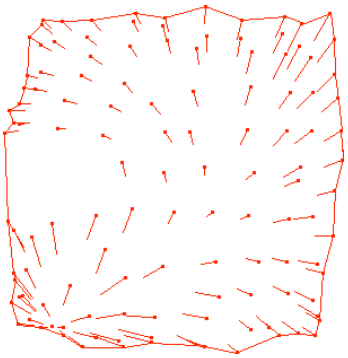

Figure 9: Vector field representing the navigational map learned after 10 training trials.

in Eq. 11 with time constant $\tau = 10$ and parameter $\beta = 0.7$. The vector field representation was obtained by rastering uniformly over the whole environment. Dots represent sampled positions and arrows indicate the directions suggested by the shifted population vector for each sampled position. Fig. 9 also gives an idea of how the robot shapes its world internally. Finally, the solid line in Fig. 8 is an example of a navigation path obtained using the navigational map of Fig. 9.

4 Discussion

We have presented a hippocampal neural model for learning spatial representations. The place cell driving system relies on extracting spatio-temporal properties of the environment from visual sensory inputs. Incoming stimuli are interpreted by means of Walsh-like filters that only respond to specific visual patterns. The activity of the filters contributes to place cell firing. At the very beginning, the robot starts with an empty model. Unsupervised Hebbian learning is used to build the hippocampal neural structure incrementally. New synapses and new place cells are created on-line to cover the environment uniformly and densely. In order to interpret the ensemble network activity, a population vector coding scheme is used.

Given the place fields, simple asymmetric learning rules can be used to guide the animal (or robot) to a target location [1, 8].

Acknowledgments

Supported by the Swiss National Science Foundation, No. 21-49174.96. The authors thank Dario Floreano for useful discussions.

References

[1] L.F. Abbott and K.I. Blum. *Cerebral Cortex*, 1996.

[2] H.C. Andrews. Academic Press, New York, 1970.

[3] M.A. Brown and P.E. Sharp. *Hippocampus*, 5:171–188, 1995.

[4] N. Burgess, M. Recce, and J. O'Keefe. In M.A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 468–472. Bradford Books, MIT Press, 1995.

[5] D. Debanne, B.H. Gahwiler, and S.M. Thompson. *Proc. Natl. Acad. Science USA*, 91:1148–1152, 1994.

[6] B. Fritzke. *Systems Biophysics, Inst. for Neural Comp., Ruhr-Universitt Bochum, http://pikas.inf.tu-dresden.de/ fritzke/JavaPaper/*, 1997.

[7] A.P. Georgopoulos, A. Schwartz, and R.E. Kettner. *Science*, 233:1416–1419, 1986.

[8] W. Gerstner and L.F. Abbott. *Journal of Computational Neuroscience*, 4:79–94, 1996.

[9] W. Gerstner, R. Ritz, and J.L. van Hemmen. *Biol. Cybern.*, 69:503–515, 1993.

[10] J.J. Knierim, H.S. Kudrimoti, and B.L. McNaughton. *Journal of Neuroscience*, 15:1648–1659, 1995.

[11] W.B. Levy and D. Steward. *Journal of Neuroscience*, 8:791–797, 1983.

[12] H.A. Mallot, M.O. Franz, B. Schölkopf, and H.H. Bülthoff. In W. Gerstner, A. Germond, M. Hasler, and J.D. Nicoud, editors, *Proc. of the Int. Conf. on Artificial Neural Networks (ICANN'97)*, pages 751–756, Lausanne, Switzerland, 1997. Springer Verlag.

[13] E.J. Markus, C.A. Barnes, B.L. McNaughton, V.L. Gladden, and W.E. Skaggs. *Hippocampus*, 4:410–421, 1994.

[14] J. O'Keefe and L. Nadel. Clarendon Press, Oxford, 1978.

[15] O. Trullier and J.A. Meyer. In W. Gerstner, A. Germond, M. Hasler, and J.D. Nicoud, editors, *Proc. of the Int. Conf. on Artificial Neural Networks (ICANN'97)*, pages 757–762, Lausanne, Switzerland, 1997. Springer Verlag.

[16] M.A. Wilson and B.L. McNaughton. *Science*, 261:1055–1058, 1993.