In: Proc. 26th International Conference on Artificial Networks (ICANN 2017).

Robot Localization and Orientation Detection based on Place Cells and Head-direction Cells

Xiaomao Zhou[1,2], Cornelius Weber[2] and Stefan Wermter[2]

[1] Harbin Engineering University, College of Automation,
Nantong Street. 145, 150001, Harbin, China
[2] University of Hamburg, Department of Computer Science,
Vogt-Kölln-Strasse. 30, 22527 Hamburg, Germany
{zhou,weber,wermter}@informatik.uni-hamburg.de
http://www.informatik.uni-hamburg.de/WTM

Abstract. Place cells and head-direction cells play important roles in animal navigation and have distinguishable firing properties in biology. Recently, a slowness principle has been argued as the fundamental learning mechanism behind these firing activities. Based on this principle, we extend previous work, which produced only a continuum of place and head-direction cells and mixtures thereof, to achieve a clean separation of two different cell types from just one exploration. Due to the unsupervised learning strategy, these firing activities do not contain explicit information of position or orientation of an agent. In order to read out these intangible activities for real robots, we propose that place cell activities can be utilized to build a self-organizing topological map of the environment and thus for robot localization. At the same time, the robot's current orientation can be read out from the head-direction cell activities. The final experimental results demonstrate the feasibility and effectiveness of the proposed methods, which provide a basis for robot navigation.

Keywords: place cell, head-direction cell, slowness principle, unsupervised learning, robot localization

1 Introduction

Precise metric map building and pose detection are always convenient for robot navigation. During the last decades, various sensors [1] and approaches [2] have been proposed to build high-precision world representations and to detect the exact postures of a mobile agent. However, the necessity of using high-accuracy sensory information for navigation is still an open issue. Many animals present excellent navigational capabilities in various environments with limited sensing abilities compared to man-made sensors. For example, the human visual system cannot measure distance with an accuracy as a laser rangefinder does. Thus, the underlying mechanisms of visual processing should be a possible basis for successful navigations.

For animal navigation, findings of place cells [3], head-direction cells [4] and grid cells [5] in the hippocampus and entorhinal cortex can give some insights into how visual information is processed. In the hippocampus, certain neurons termed

place cells have been found to become active when an animal reaches a certain position in the environment. Their firing activities are largely independent of the direction in open areas while displaying a certain degree of orientation-dependence in specific environmental structures for example, in linear tracks [6]. In contrast, head-direction cells are related to the direction of the animal's head. When the animal turns its head to a specific direction, these cells will fire significantly and their activities show a clear invariance to the position [7].

To model these firing activities computationally, the slowness principle [8] has been shown to be a candidate mechanism for learning. With a hierarchical structure, place cells or head-direction cells in a virtual rat were reproduced by processing its visual inputs only [9]. However, the strategy of changing movement patterns to generate different cell types was not plausible considering that, to get various cell types, an animal will not explore the same environment many times with different moving strategies. Note that this simple feed-forward model is memoryless and does not use recurrent circuitry for path integration.

In this paper, we propose a biologically plausible approach to generate place and head-direction cells simultaneously, in which an agent can produce different cell types from just one exploration. Then, we present a method of interpreting the encoded information. Considering that place cell activities encode positional information but in an implicit fashion, we bypass this problem by using a self-organizing network [10] to learn the relational structure of these activities in an unsupervised manner. By doing this, a topological map of the explored space is built for localization, which can be used for future navigation without explicit positional information. Based on head-direction cells which only fire when it comes close to their preferred direction, we calculate the agent's real-time orientation and estimate how much directional information is encoded in these activities in order to assess whether they contain sufficient information for navigation.

2 Experimental Setup and Methodologies

2.1 Experimental Set-up

We use the RatLab simulator to generate our training data and to test our proposed approach [11]. RatLab provides an easy way to simulate a rat's random exploration in a variety of environments. It allows to change the environmental enclosures and textures, to place obstacles and to configure the moving strategies of the virtual rat. The rat's visual input during moving can be collected as training images to the Slow Feature Analysis (SFA) algorithm, in order to train place cells and head-direction cells. In this work, we use RatLab to replicate a real robot experiment.

2.2 Slow Feature Analysis

SFA is an unsupervised learning algorithm based on a slowness principle [8]. For raw sensory inputs, SFA intends to capture the slowly varying signals and

leave out quickly changing ones, such as trivial noise. In most cases, these slowly varying features encode the underlying causes of input changes, which contain the most descriptive statistical regularities.

Mathematically, the learning problem behind SFA can be described as follows: Given an I-dimensional input signal $x(t) = [x_1(t), x_2(t), \ldots, x_I(t)]$, find a set of J real-valued input-output functions $g(t) = [g_1(t), g_2(t), \ldots, g_J(t)]$ such that the output signal $y(t) = [y_1(t), y_2(t), \ldots, y_J(t)]^T$ with $y_j(t) = g_j(x(t))$ satisfies the criteria:

$$\Delta(y) = \langle \dot{y}_j^2 \rangle_t \quad \textit{is minimal} \tag{1}$$

under three constraints:

$$\langle y_j \rangle_t = 0 \quad \textit{zero mean}$$
$$\langle y_j^2 \rangle_t = 1 \quad \textit{unit covariance}$$
$$\forall j' < j : \langle y_{j'} y_j \rangle_t = 0 \quad \textit{decorrelation}$$

with $\langle \cdot \rangle$ and $\dot{y}$ indicating temporal averaging and the time derivative of y, respectively. Equation 1 expresses the primary objective of this optimization problem. The first two constraints guarantee the output signals with meaningful information, instead of a trivial constant value. Decorrelation avoids uninteresting solutions where different output signals encode the same information. In our work, we implement SFA with a hierarchical architecture proposed in [11] to learn place and head-direction cells from raw visual input.

2.3 Growing When Required Network

For the purpose of map building and localization, a self-organized Growing When Required (GWR) network [10] learns a spatial map by extracting the important topological relations of place cell activities during exploration. To map the input space, GWR exerts a dynamic growing criterion and grows whenever the current nodes cannot represent the input accurately. It can respond quickly to changes in the input distribution by dynamically creating or deleting nodes and edges during the learning process.

The GWR network starts with two random nodes n_1 and n_2 representing the input space. For each iteration, the two best matching nodes s and t, determined by the distance to the input, are first found and connected. Whenever s and t fail to represent the current input sufficiently well, a new node will be added halfway between them. The criterion of adding new nodes also relies on the winning node's firing counter. The training will drive the weights of the winner node and its neighbours to move towards the input and the rarely used nodes will be deleted by an aging mechanism. The algorithm will keep iterating until meeting a stop criterion, such as performance behavior and network size.

3 Approach, Experimental Results and Discussion

3.1 Training with One Exploration

The training process and corresponding model configurations are illustrated in Fig. 1. Each network includes a hierarchical architecture containing three SFA

layers and one ICA layer. Each SFA layer consists of a regular grid of SFA nodes which contain 30 or 40 output channels (cells) and act on a local receptive field, for example, the first layer has 63×9 SFA nodes working directly on the raw input images. On top of the SFA layers, one final ICA node performs a function of sparse coding on the raw SFA outputs to produce a more localized representation. We use 40 units in each SFA node for the head-direction cells outnumbering 30 units for the place cells to increase the precision of the orientation estimation (Sec. 3.3). For the training, during the *forward movement*, the robot changes its position continuously, while rotational information changes slowly. Obviously, the emerging slow features during forward movement compactly encode the robot's direction. Since SFA is sensitive to slowly varying signals, it will extract the directional information within this phase. Thus, the visual data from this moving period can be used for head-direction cell network training. Similarly, positional information will be a relatively slow signal during the robot's *turning movement* where directional information changes quickly. So we can use this phase to train the place cell network. This mechanism assumes that learning is modulated by behavior and, more specifically, that transitional and rotational motion can be differentiated to train different types of cells. This can in principle be supported by behavioral modulation of head-direction cells [12] and of place cells [13].

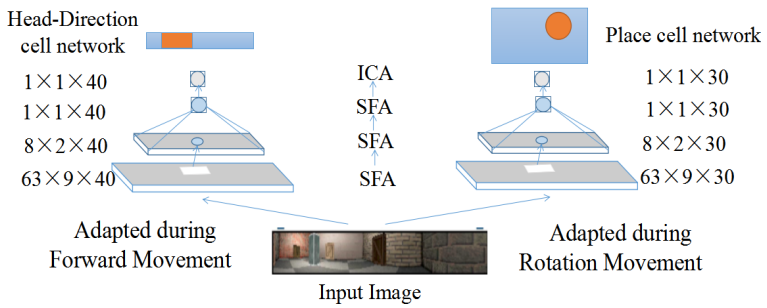

Fig. 1. Training place cell and head-direction cell networks in different phases of the same trajectory. Layers are trained sequentially from bottom to top.

Parts of the training results can be seen in Fig. 2. The learnt place cells only fire in a certain position in the environment (Fig. 2(a)) and they have little directional tuning, which means their activities are invariant to direction (Fig. 2(b)). Head-direction cells show little position preference (Fig. 2(c)), but they will be significantly active when it comes to their preferred direction (Fig. 2(d)).

In order to assess whether these two different cell types have obtained distinguishable firing properties, we adopt the concept of entropy. Assume a set of distributions using these activities as their probability values. For example, place cells have similar probabilities to be activated for different head directions. Their activities closely approximate a uniform random distribution and have a

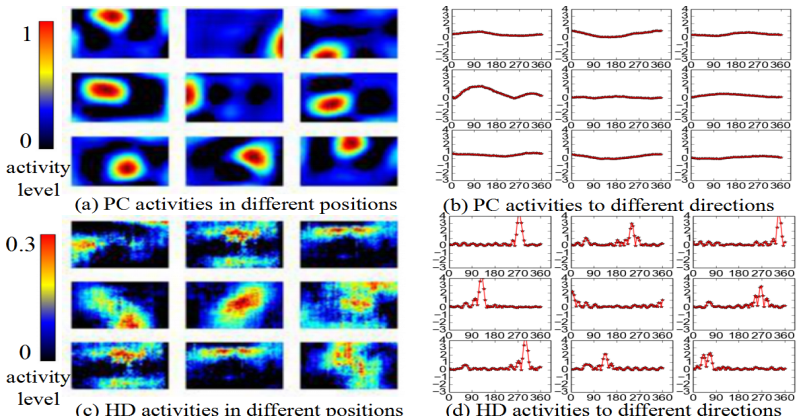

Fig. 2. Place cell activities and head-direction cell activities trained by the proposed approach (9 randomly chosen place cells are shown in (a) and (b), and 9 head-direction cells are shown in (c) and (d)).

large entropy of direction H_{dir}. In contrast, head-direction activities are more peaked since they have large probability values for a certain direction, thus with a smaller entropy H_{dir}. The entropies are calculated by:

$$H_{dir} = -\sum_{\theta}^{dir} a_\theta \ln(a_\theta); \quad H_{pos} = -\sum_{i}^{pos} a_i \ln(a_i) \tag{2}$$

where a_θ represents normalized cell activity at direction θ averaged over positions, while a_i represents normalized cell activity at position i averaged over directions. Fig. 3(b) shows that we obtain two cell types with clearly different properties using our proposed training. For comparison, Fig. 3(a) shows the training result of the existing model [9] which produces a continuum between place and head-direction cells, but not two distinct clusters.

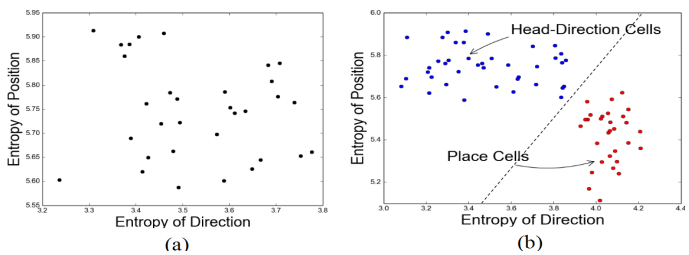

Fig. 3. Entropy analysis: (a) Training results from standard SFA, without knowledge to distinguish different cell types. (b) With our proposed training method, different cell types form two separate clusters.

3.2 Map Building Based on Place Cells

In our approach, we obtain place cell activities based on an unsupervised learning algorithm, which means these activities do not have a predefined relation to the agent's real-world position. However considering place cells' firing preference to different positions, collectively, these activities encode a certain environmental position. For spatial learning, a GWR network is used to capture the relationships of these activities and to build a topological map. Since the inputs to the GWR network are high-dimensional place cell activities, we use multidimensional scaling (MDS) [14] to visualize the map in two dimensions. Fig. 4(a) shows that the trained GWR network represents the 2D topology of the agent's exploration area based on the place cells' activities. The robot's current position is represented by the winner neuron which has the weight vector closest to the input.

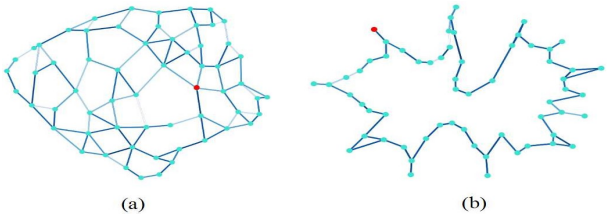

Fig. 4. (a) GWR result with input from a virtual rat's place cell activities in a square arena (rotation is caused by MDS). (b) GWR result based on the head-direction cell activities. The current state is represented by the winner node (red node).

3.3 Orientation Detection Based on Head-direction Cells

We applied the same analysis as in Sec. 3.2 to the head-direction cell activities. Fig. 4(b) shows that the trained GWR network captures the ring topology of a circular ground truth. However, due to the unsupervised learning, the head-direction cells' activation vector contains no predefined directional information.

In order to estimate the precision of the directional information lying in these activities, we use a simple vector sum process, based on knowledge of the real orientation. The principle behind the vector sum is that when the robot's current orientation is close to some head-direction cells' preferred direction, these cells will fire significantly and will dominate the sum result. However, in real scenarios, exploration may not fully cover all positions in the environment or perceive one position from different perspectives. We may have some head-direction cells with poor tuning, like Cell 2 in Fig. 5(a). So we assign to each cell k a reliability value R_k with a normalization as follows:

$$R_k = \Big| \sum_{\theta}^{dir} \boldsymbol{a}_{\theta}^{k} \Big| \Big/ \sum_{\theta}^{dir} |\boldsymbol{a}_{\theta}^{k}| \tag{3}$$

where $\boldsymbol{a}_\theta^k$ is the activity vector of head-direction cell k to the direction of θ.

By doing this, more reliable head-direction cells (with clear direction preference) will contribute more to the vector sum than unreliable ones. The orientation sum vector $\boldsymbol{v}$ is calculated as follows:

$$\boldsymbol{v} = \sum_{k}^{cells} (a^k \, R_k \sum_{\theta}^{dir} \boldsymbol{a}_\theta^k) \tag{4}$$

where a^k denotes the activity of cell K and R_k represents its reliability value. $\sum_\theta^{dir} \boldsymbol{a}_\theta^k$ approximates the activity vector of cell k to its preferred direction.

Fig. 5(c) shows that, while the robot rotates its head, the real-time orientation can be reconstructed by these cells activities with a certain accuracy. The average error over all directions is 14.73° in our case.

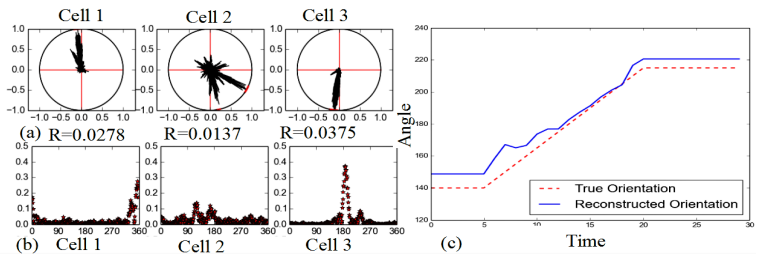

Fig. 5. (a) Head-direction cell activities to different directions. (b) Normalized head-direction cell activities to visualize their reliability R. (c) True and reconstructed robot orientation from 40 head-direction cell activities

4 Conclusion and Future Work

In this paper, we propose a method to train two distinct clusters of place cells and head-direction cells with SFA during just one exploration. The results showed that different cell types can emerge simultaneously with our proposed approach, using visual input and modulation from motion-related information. Using self-organised mapping and multidimensional scaling, we reconstructed a 2D area topology from the place cell activities and a 1D ring topology from the head-direction cell activities. With the head-direction cells, we obtained the agent's orientation by considering their firing preference to direction. Our experimental results demonstrate the potential of the proposed approach to provide fundamental information for robot navigation.

Obtaining of directional information allows future work of building a cognitive map, which has previously been done by using spatial information obtained from a ceiling camera [15]. The directional information of a link between neighboring GWR nodes, in our work, can be assigned based on the head-direction

cell activities, without knowledge of the exact direction, and can be reused during path planning. Thus, a natural next step is to validate this work on a real robot platform, which may lead to a robust nature-inspired navigation system.

Acknowledgments. We thank Xiaolin Hu for feedback and acknowledge support from the German Research Foundation DFG, project CML (TRR 169).

References

1. Lobo, J., Marques, L., Dias, J., Nunes, U., Almeida, A.T.: Sensors for mobile robot navigation. In: Almeida, T.A., Khatib, O. (eds.): Autonomous Robotic Systems 1998, vol. 236, pp. 50–81. Springer London (1998)
2. Borenstein, J., Everett, R.H., Feng, L., Wehe, D.: Mobile Robot Positioning - Sensors and Techniques. Journal of Robotic Systems 4, 231-249 (1997)
3. Keefe, J.O., Burgess, N.: Geometric determinants of the place fields of hippocampal neurons. Nature 381, 425–428 (1996)
4. Taube, J.S., Muller, R.U., Ranck, J.B.: Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis. Journal of Neuroscience 10, 420–435 (1990)
5. Moser, E.I., Kropff, E., Moser, M.B.: Place cells, grid cells, and the brain's spatial representation system. Annu. Rev. Neurosci 31, 69–89 (2008)
6. Dombeck, D.A., Harvey, C.D., Tian, L., Looger, L.L., Tank, D.W.: Functional imaging of hippocampal place cells at cellular resolution during virtual navigation. Nature neuroscience 13, 1433–1440 (2010)
7. Sharp, P.E., Hugh, T.B., Jeiwon C.: The anatomical and computational basis of the rat head-direction cell signal. Trends in neurosciences 24, 289–294 (2001)
8. Wiskott, L., Sejnowski, J., T.: Slow Feature Analysis: Unsupervised Learning of Invariances. Neural Computation 14(4), 715–770 (2002)
9. Franzius, M., Sprekeler, H., Wiskott, L.: Slowness and Sparseness Lead to Place, Head-Direction, and Spatial-View Cells. PLoS Comput Biol 3, 1605–1622 (2007)
10. Marsland, S., Shapiro, J., Nehmzow, U.: A self-organising network that grows when required. Neural Networks, pp. 1041-1058 (2002)
11. Schönfeld, F., Wiskott, L.: RatLab: an easy to use tool for place code simulations. Frontiers in Computational Neuroscience 7, 104–111 (2013)
12. Chen, L.L., Lin, L.H., Green, E.J., McNaughton, B.L.: Head-direction cells in the rat posterior cortex. Experimental Brain Research 101, 8–23 (1994)
13. Lu, X., Bilkey, D.K.: The velocity-related firing property of hippocampal place cells is dependent on self-movement. Hippocampus 20, 573–583 (2010)
14. Borg, I., Groenen, J.F.: Modern multidimensional scaling: Theory and applications. Springer Science and Business Media (2005)
15. Yan, W., Weber, C., Wermter, S.: Learning indoor robot navigation using visual and sensorimotor map information. Frontiers in Neurorobotics 7(15), 10.3389/fnbot.2013.00015 (2013)