Neural Networks 72 (2015) 62–74

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

2015 Special Issue

Goal-oriented robot navigation learning using a multi-scale space representation

M. Llofriu [a,c,*], G. Tejera [c], M. Contreras [b], T. Pelc [b], J.M. Fellous [b], A. Weitzenfeld [a]

[a] University of South Florida, United States
[b] University of Arizona, United States
[c] Universidad de la Republica, Uruguay

ARTICLE INFO

Article history:
Available online 19 October 2015

Keywords:
Place cells
Hippocampus
Spatial cognition model
Multiscale spatial representation
Reinforcement learning

ABSTRACT

There has been extensive research in recent years on the multi-scale nature of hippocampal place cells and entorhinal grid cells encoding which led to many speculations on their role in spatial cognition. In this paper we focus on the multi-scale nature of place cells and how they contribute to faster learning during goal-oriented navigation when compared to a spatial cognition system composed of single scale place cells. The task consists of a circular arena with a fixed goal location, in which a robot is trained to find the shortest path to the goal after a number of learning trials. Synaptic connections are modified using a reinforcement learning paradigm adapted to the place cells multi-scale architecture. The model is evaluated in both simulation and physical robots. We find that larger scale and combined multi-scale representations favor goal-oriented navigation task learning.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

We present a model for spatial cognition based on the multi-scale organization of the dorsal–ventral axes of the hippocampus. The theoretical concept of a cognitive map in the brain was first proposed by Tolman (1948) as the essential module responsible for estimating the rat's position in the environment. Through an extensive review of the literature at the time, O'Keefe and Nadel proposed that the cognitive map laid within the brain's hippocampus (O'Keefe & Nadel, 1978). This suggestion was in part based on the discovery of O'Keefe and Dostrovsky (O'Keefe & Dostrovsky, 1971) place cells in the rat's hippocampus, termed as such due to the high correlation between their firing and the rat location in the environment. The region of the environment specific to each cell, is called their place field.

This property turns these cells into a rich source of information for navigational purposes, as they provide an encoding of the rat location. Even though no individual place cell provides accurate location information by itself, it has been shown that the location of the animal can be predicted with an error of 1 cm based solely on the activity levels of an ensemble of cells (Guger et al., 2011; Jensen & Lisman, 2000; Wilson & McNaughton, 1993; Zhang, Ginzburg, McNaughton, & Sejnowski, 1998).

The discovery of grid cells in the rat's Medial Entorhinal Cortex (MEC) was first published by Fyhn, Molden, Witter, Moser, and Moser (2004), it suggested that grid cell firing signaled the rat's changing position in the environment. Hafting, Fyhn, Molden, Moser, and Moser (2005) presented findings that grid node spacing increased in a modular fashion in MEC in overall correspondence with the gradual increase in place field size along the dorsoventral axis of the hippocampus (Brun et al., 2008; Jung, Wiener, & McNaughton, 1994; Maurer, Vanrhoads, Sutherland, Lipa, & McNaughton, 2005). For a review of the multi-scale representation of space, we refer the reader to Geva-Sagiv, Las, Yovel, and Ulanovsky (2015).

The MEC also contains head-direction cells whose activity is related to the current head orientation of the rat (Blumberg, 2015). Altogether, these discoveries suggest that spatial navigation may originate in a fundamental multi-scale representation of space along the dorso-ventral axis of the hippocampal formation, modulated by self-motion and external cues that includes, although not limited to, visual detection of distal and proximal landmarks. Recent work has shown a gradient of head direction cells tuning width along the dorso-ventral axis of the MEC (Giocomo et al., 2014) in the upper layers (inputs to the hippocampus), but not in the deep layers. While there is some theoretical evidence that grid

* Corresponding author at: University of South Florida, United States.
E-mail address: mllofriualon@mail.usf.edu (M. Llofriu).

http://dx.doi.org/10.1016/j.neunet.2015.09.006
0893-6080/© 2015 Elsevier Ltd. All rights reserved.

cells may help the accuracy of spatial navigation (Guanella & Verschure, 2007), there is little evidence that they do so experimentally (Hales et al., 2014).

There is extensive research on spatial cognition models inspired by place cells coding in the rat's hippocampus used to evaluate goal-oriented spatial navigation with simulation and with real robots (Arleo, Smeraldi, & Gerstner, 2004; Barrera & Weitzenfeld, 2008; Brown & Sharp, 1995; Burgess, Recce, & O'Keefe, 1994; Caluwaerts et al., 2012; Dollé, Sheynikhovich, Girard, Chavarriaga, & Guillot, 2010; Filliat & Meyer, 2002; Gaussier, Revel, Banquet, & Babeau, 2002; Guazzelli, Bota, Corbacho, & Arbib, 1998; Krichmar, Nitz, Gally, & Edelman, 2005; Milford & Wyeth, 2007, 2009; Pata, Escuredo, Lallée, & Verschure, 2014; Recce & Harris, 1996; Redish & Touretzky, 1997; Sukumar, Rengaswamy, & Chakravarthy, 2012; Tejera, Barrera, Fellous, Llofriu, & Weitzenfeld, 2013). However, few of them incorporate some aspects of multi-scale representation of space.

Chen, Jacobson, Erdem, Hasselmo, and Milford (2013) implement an array of support vector machines on video segments to recognize places, where different segment lengths represent different scales of representation. The introduction of larger scales of representation improves recall in their classification system. Pata et al. (2014) develops a model of the hippocampus to explore the functional differences of DG and CA3 while accounting for the differences in scale across the dorsoventral axis. These two have a strong focus on place and grid cell formation. Our work, in contrast, focuses in goal directed navigation, on how to use the information provided by these cell to reach a desired goal.

Erdem and Hasselmo (2014) adds multiple scales of representation to a previous model of spatial navigation based on forward lookup probes, which resemble short-wave ripple (SWR) activity in the hippocampus. The addition of larger scales of representation improves the effective distance of forward lookup probes, improving navigational performance. We believe our work complements this approach. Short-wave ripple activity have been suggested to guide navigation (Johnson & Redish, 2007; Pfeiffer & Foster, 2013), but it occurs during sleep or when the rat is still (Foster & Wilson, 2006). Thus, while Erdem and Hasselmo (2014) work focuses on high level planning during key decision points, our model focuses on the decision making that takes place while the rat is in motion. Our working hypothesis is that this decision making can also benefit from different scales of representation.

We based our study on the biological role of dorsal and ventral hippocampal place cells and contrast the roles of the small, medium and large place fields represented across the septo-temporal (dorsoventral) axis of the hippocampus. We develop a neural architecture of multi-scale hippocampal place cells to be evaluated during a goal-oriented robot navigational task. This task involves learning to locate a fixed goal in a circular arena, inspired by the Morris' water maze task, where instead of a submerged platform, the goal can only be recognized when the robot is very close to it (de Jong, Gereke, Martin, & Fellous, 2011). Distal cues are set on the perimeter of the arena to facilitate localization. The task involves multiple trials during which navigation to the goal is reinforced by applying a Q-Learning algorithm (Sutton & Barto, 1998), inspired by the neuro-modulatory effects of dopamine (Cox & Krichmar, 2009), and adapted to the multi-scale nature of place fields. The task is evaluated using computer simulations and physical robots. Section 2 describes the spatial cognition model, Section 3 presents the goal-oriented task and the experimental layout, Section 4 presents simulated and robot experimental results, Section 5 includes a discussion of the results and Section 6 provides conclusions and discussion of future work.

2. The spatial cognition model

The spatial cognition model is comprised of six main modules, described below and shown in Fig. 1.

It has been proposed that navigation involves the interaction of four components: place cells, head direction cells, local view and path integration (Redish & Touretzky, 1997; Touretzky, 2002). We consider our path integration and local view components as solved. Namely, place cell firing values are derived from sources of location information directly, rather than computing them from path integration and visual information, as will be explained in the Experiments section. Thus, we focus in this work on the place cell and head direction cells components and their contribution to learning using multiple scales. Our model uses this multi-scale representation as the information source for a reward driven learning system (Krichmar & Röhrbein, 2013).

2.1. Modules

Place Cell Module. This module calculates the firing of a population artificial place cells. They take the current position x of the robot as input and calculate the firing rate as Eq. (1).

$$f_i = exp\left(-\frac{(x-c_i)^T \Sigma_i^{-1}(x-c_i)}{2}\right). \quad (1)$$

Where f_i is the firing rate of cell i, c_i its preferred location and Σ_i its covariance matrix. Namely, each cell fires according to a $2D$ gaussian function with a center on each place cell preferred position, as modeled by O'Keefe and Burgess (1996).

The key of this work involves the use of different scales of place cells, which we map to choosing different Σ. The covariances matrix are always of the form $\sigma^2 I$, where σ^2 models de specificity and I is the identity matrix.

Head Direction Module. This model computes the firing of a population of artificial head direction cells. This module takes the current heading θ of the robot and compute the firing rate of each cells as Eq. (2).

$$f_i = exp\left(-\frac{(\theta-\theta_i)^2}{2\sigma^2}\right). \quad (2)$$

Where f_i is the firing rate of the ith head direction cell, σ^2 its variance and θ_i its preferred orientation. Thus, this cells are also computed as a gaussian function with the peak in the cell's preferred value.

Multi-Scale QL Module. This module performs Q-Learning on the information provided by the place cells and head direction cells. Place and orientation information is obtained by selecting all possible pairs from both sets and computing the resulting activity as the product of both the place cell and head direction cell. This combined source of information is passed onto the QL module, which will be explained in detail below.

Taxic Behavior Module. This behavior moves towards a visible goal. It works cooperatively with the QL learning module by assigning a fixed value to the action that will take the robot to the goal. In the framework proposed by Guazzelli et al. (1998), this module corresponds to the execution of the affordance of going to a visible goal.

Exploration Behavior. This module promotes exploration in early phases of an experiment. The exploration value is calculated as shown in Eq. (3), where episode is the episode number, maxReward is the maximum reward possible given to the robot, and β is a given parameter that models how fast the exploration

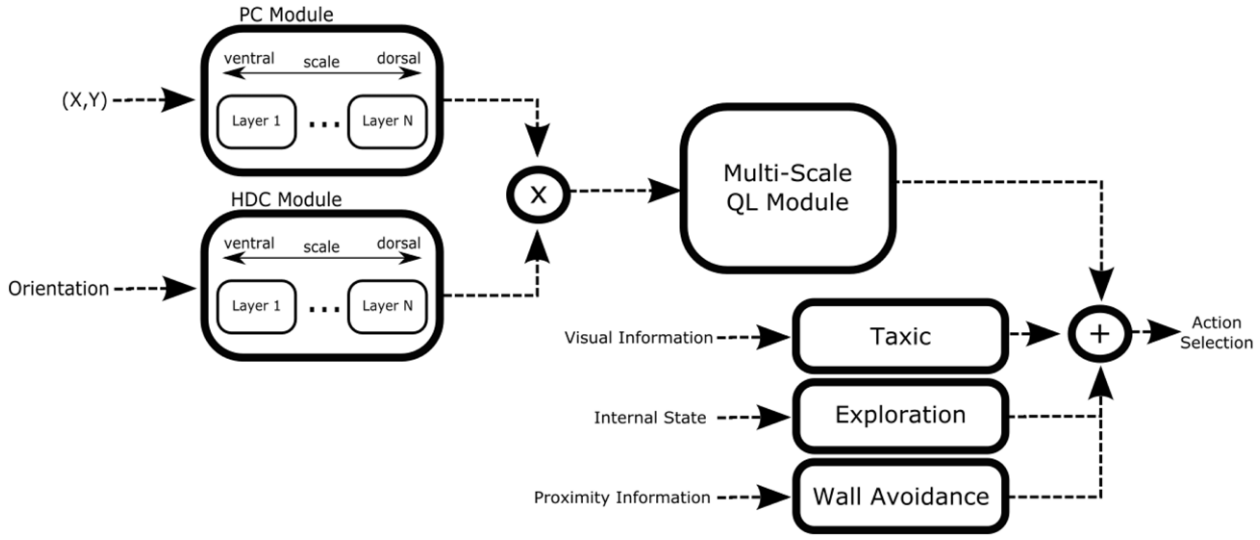


Fig. 1. The system modules and the flow of information. The dotted arrows represent flow of information. The Place Cell Module (PC) receives location information. The Head Direction Cell Module (HDC) receives orientation information. PC and HDC information are combined and sent to the Multi-Scale Q-Learning Module, which outputs action selection values. The Taxic Module receives visual information, the Exploration Module incorporates internal state information, the Wall Avoidance Module processes proximity information.

value decays. Higher values of β mean slower decay, and thus, more exploration.

$$expval = maxReward * 0.5 * exp(-episode/\beta). \quad (3)$$

This module works cooperatively too.

Wall Avoidance Behavior. This module also works cooperatively with the QL learning module. It prevents the robot from bumping into walls by assigning negative values to the actions that would lead to them.

Action Selection. In order to select an action a linear combination of each action value provided by the different modules (QL, Taxic, Exploration, Wall Avoidance) is performed. The action with the greater value is chosen as the next action to execute in a winner-take-all fashion.

2.2. Dorso-ventral multi-scale place cell driven learning

Place cells are a good source of information for location (Wilson & McNaughton, 1993). We adapted the classical RL algorithm in order to use a multi-scale space representation as we describe in more details next.

2.2.1. Place cells as the RL state

When a rat is in a specific location within a known environment, a set of place cells fire signaling that the animal is within their place fields. Since place fields overlap (O'Keefe, 1976), several cells might be firing at any given moment.

Thus, in the animal's brain, the location is encoded as the activity of an ensemble of cells. This contrasts with the intuitive representation of the state as vector holding position and orientation information. Thus, our RL algorithm input is comprised of a set of place cell activities that encode the current location (Arleo et al., 2004; Barrera, Tejera, Llofriu, & Weitzenfeld, 2015; Chavarriaga, Strösslin, Sheynikhovich, & Gerstner, 2005; Gaussier et al., 2002).

When dealing with a continuous state space, some RL solutions resort to discretization of the environment, due to its simplicity of implementation (Kober, Bagnell, & Peters, 2013). Fig. 2 illustrates the difference between an environment discretization and the use of place cell like states. Instead of discretizing the environment into a fixed grid, each place cell are laid out over the environment with overlaps with other place cell fields. The place cells ensemble activity will encode the location of the robot at any point in time. This activity can then be used as the current state in RL algorithms (Arleo et al., 2004).

More formally, we are considering the problem of dealing with continuous state RL, where the continuum corresponds to the position and orientation of the robot. Let the continuous state be $c = (c_1, \ldots, c_n) \in \mathbb{R}^n$. Note that variables c_i have a continuous domain. For example, in our case, c is composed of the position and orientation of a robot in a $2D$ plane, i.e. $c = (x, y, \theta)$.

Singh, Jaakkola, and Jordan (1995) describe work on RL over soft states and allows for the use of the place cell ensemble activity directly as the state. The main idea behind Singh et al. soft-clusters, or soft-states, approach is to consider a new discrete set of states $S = \{s_1 \ldots s_m\}$, where the states s_i are soft-clusters on the space of c, $\mathbb{R}^n$. Soft-clusters are defined with a conditional probability, where every possible value c will belong to a cluster s_j with certain probability $p(s_j|c)$. We call the probabilities $p(s_i|c)$ the *activation value* of state s_i to emphasize the place cell metaphor and denote it as $A(s_i, c)$. Fig. 2.b shows an environment where 5 soft state are laid out. Given the robot position x, two of them show an activity $A(s_i, x)$ greater than zero, while the other three are inactive.

Notice that when the robot is in a specific continuous state c, more than one cluster may be active simultaneously, the same way as more than one place cell might be firing simultaneously.

Summarizing the notation:

- variable c represents the state in it original continuous nature, e.g. the position of a mobile robot in our navigation problem,
- variable s_i represents a state of the multi-scale algorithm, it corresponds to the place cell that fires in a subregion of the environment or the soft states from the work by Singh et al. (1995)
- $A(s_i, c)$ is the activation of the soft-cluster s_i when the continuous state is c.

With the soft state framework in mind, Fig. 2.b can be reinterpreted as a set of soft states, s_j, laid out over the environment and their level of blue represents their activation, or $p(s_j|location)$.

Our clusters will always have a decreasing activation as c gets away from a preferred value, or center, of cluster s_i. That is, the activation will be decreasing with the distance of c to the cluster center. The activation will range from 0, when c is far away from

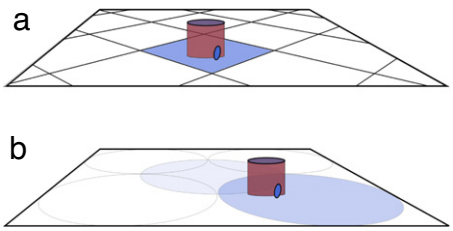


Fig. 2. Environment discretization (a) vs place cell states (b). The trapezoid represents the environment and the robot is shown in magenta. Each state activation is shown in blue, the darker the color, the more active the state is. In a usual discretization, only one state is fully active given the position of the robot (a), whereas activation is shared among many place cell states in the other case (b), two for this figure. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

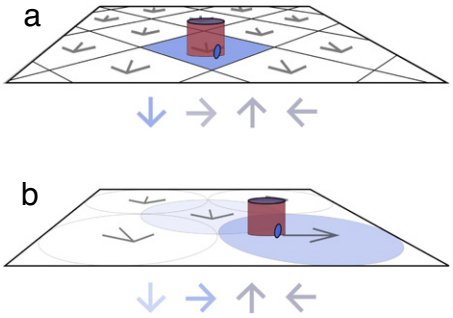


Fig. 3. The difference in the action selection process for an environment discretization (a) and soft states (b). Each state has a preferred action in both cases. In the discretization, the only active state's preferred action is picked. With soft states, each active state contributes with its preferred action in proportion to their activation value.

this preferred value; to 1, when c and s_i preferred value are the same. This correlates also with the way place cells fire.

Recapitulating, we have applied the concept of place cells as a way to represent the RL state, which is made possible by Singh et al. (1995) algorithm for QL using soft states. This allows us to perform QL over the activity of an ensemble of place cells or soft-states.

2.2.2. Reinforcement learning equations on place cells

The canonical Q-Learning algorithm maintains a table of the expected reward $Q(s, a)$ of performing an action a when being in a state s. In order to select an action when in state s, a greedy approach would pick the action to perform according to (4).

$$a = argmax_a Q(s, a). \tag{4}$$

In the soft state framework, the action selection becomes as described in Eq. (5). Note that canonical QL could be understood as having only one active soft state at a time, thus $A(s_i, x)$ would be 1 for only one s_i. Then the canonical QL action selection equation becomes a special case of Eq. (5). This resembles the use of place cells as radial basis functions proposed by Burgess et al. (1994).

$$a = argmax_a \sum_S Q(s_i, a) * A(s_i, c). \tag{5}$$

Fig. 3 contrast the action selection process when using a fixed discretization and when using soft states.

After every action a is taken from state s and arriving to state s', tabular QL algorithms update this table according to (6).

$$Q(s, a) = Q(s, a) + \alpha * \left(r + \gamma max_{a'} Q(s', a') - Q(s, a)\right) \tag{6}$$

where α and γ correspond to a learning parameters and r is the obtained reward signal.

In the soft state framework, all soft states s_i must be updated according to their activation value, see Eqs. (7) and (8). The maximal expected return from the successor state is replaced with a sum over all soft state's Q value for each possible action, where c' is the successor state in the continuous space. The value is normalized over the total activation to keep Δ (Eq. (8)) bounded.

$$Q(s_i, a) = A(s_i, c) * (Q(s_i, a) + \Delta) + (1 - A(s_i, c)) * Q(s_i, a) \tag{7}$$

$$\Delta = \alpha * \left(r + \gamma max_{a'} \sum_{s'} \frac{Q(s_i, a') * A(s_i, c')}{\sum_{s'} A(s_i, c')} - Q(s_i, a) \right). \tag{8}$$

This equation is also a special case for the canonical QL update equation. Since, for the canonical QL, $A(s_i, c)$ would be different from 0 for only one s_i, only the first term would apply to the active state and the second to all other states, leaving the value unmodified. The maximization over the successor state also degenerates to the canonical formula when only one state is active.

2.2.3. Multiple scales of place cells

Place fields from different parts of the Hippocampus have different sizes (Jung et al., 1994; Keinath et al., 2014; Long, Bunce, & Chrobak, 2015). Dorsal place fields are smaller with higher spatial specificity, providing a fine grained spatial discretization. In contrast, ventral place fields are larger and have consequently lower spatial specificity, providing a coarse grained source of spatial information.

Because of these cells fire simultaneously they provide a redundant multi-scaled encoding of the animal location.

Fig. 4 illustrates the multi-scale soft states representation and the action selection process.

The key of the multi-scale concept is to have soft-clusters with different degrees of selectivity, as there are different scales of place cells. Some soft states would be active in a confined range of the continuous state space while others would be active in larger regions of the continuous state space. For example, in a navigational task, some soft states would be active only within a radius of 0.1 m of the cluster center, whereas others would be active anywhere within 1 m of the cluster center.

At any given moment, many soft states could be active at the same time. Some of them would be more selective states, covering a small neighborhood near the current continuous state c, whereas other active states would be less selective. After an action is performed, the learning rule is applied to all soft states. Thus, the outcome of that action would be learned for states that are going to modify the behavior only locally in the future and for states that influence the behavior in larger region of the continuous state space. Thus, the agent will learn fine grained policies and coarse grained policies at the same time, combining them into a single policy when performing the action selection.

2.3. Model implementation details

The model was implemented using the Mobile Internet Robotics (MIRO) (Weitzenfeld, Gutierrez-Nolasco, & Venkatasubramanian, 2003) simulator and the Neural Simulation Language (NSL) (Weitzenfeld, Arbib, & Alexander, 2002). The same model implementation, with the exact same parameters, was used for both the continuous simulated environment and the real robot one. The only difference was whether the model moved a simulated robot or the real one after each decision.

The robot possible actions consisted on: go forward 0.05 m, rotate $\frac{\pi}{8}$ to the left and rotate $\frac{\pi}{8}$ to the right. Note that the actions are relative to the robot.

The state for this algorithm was comprised of location and orientation information. Orientation information was needed due

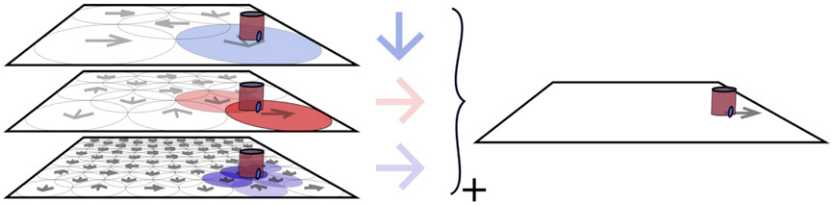


Fig. 4. The action selection process for the multi-scale case. The active place cells on every layer contribute a desired action weighted by their activation value. All values are added together and the action with the maximum action is performed.

Table 1
Model parameters and their default value.

Parameter	Default value
Learning rate α	0.1
Exploration decay rate β	1
Number of place cell layers	3
Place cell diameter	(0.1, 0.3, 0.6)
Place cells per layer	100
Number of head direction cell layers	4
Maximum head direction cell width	π
Minimum head direction cell width	$\frac{\pi}{16}$
Step size	0.05 m
Step angle	$\frac{\pi}{8}$

to the fact that rotations were coded in the robot's frame, so it needed to be aware of its orientation to take the right choice.

Location was encoded using gaussian place cells fields, as explained earlier. Three layers of 100 uniformly distributed place cells each were used. Place field diameter was varied from 0.10 to 0.6 m throughout these layers, corresponding to data reported on dorsal and medial hippocampus (Maurer et al., 2005). The activation function was nulled if it was lower than 0.2 for computational reasons.

Orientation was also encoded using gaussian head direction cells. Four layers of orientation functions were used and the selectivity varied from π to $\pi/16$. The number of functions per layer varied depending on the selectivity in this case.

Soft-states were computed by combining all possible location and orientation cells activity levels. The resulting activity was computed by multiplying the activity of the individual cells as shown in Eq. (9), where x represents a 2D location, θ is the robot orientation, x_s is s preferred location, θ_s is s preferred orientation, σ variables are the specificities of s for location and orientation and g represents a unnormalized gaussian function.

$$A(s, x, \theta) = g(x, x_s, \sigma_{s,x}).g(\theta, \theta_s, \sigma_{s,\theta}). \quad (9)$$

Table 1 summarizes the default parameter values for the model.

2.4. Biological correlation of the model

The size and complexity of the environment is used by the hippocampus to achieve multiscale navigation in bats and rats (Geva-Sagiv et al., 2015). The mechanisms of this multi-scaling lies in the different receptive fields size along the anatomical dorsoventral axis of the hippocampus and entorhinal cortex (Kjelstrup et al., 2008; Royer, Sirota, Patel, & Buzsáki, 2010). Anatomical evidence in the hippocampus suggests that dorsal and ventral levels project onto each other and exchange information internally (Strange, Witter, Lein, & Moser, 2014). These space representations are then projected to multiple structures including the ventral striatum and the ventral tegmental area, structures involved in reward and decision making. There is a well-known functional loop structure between the hippocampus and the ventral tegmental area (dopamine center), which could support the type of reinforcement learning used in our model (Lisman & Grace, 2005). Our algorithm takes advantage of the ability for large place fields to provide a global view of the environment. This scenario matches the common conception regarding the ventral striatal functionality of driving behavior on the basis of the motivational value of the environment (Lansink & Pennartz, 2015). It is also consistent with the fact that ventral hippocampus projects more strongly to ventral striatum (Arszovszki, Borhegyi, & Klausberger, 2014; Gasbarri, Packard, Campana, & Pacitti, 1994).

The reader may have noticed that head direction information is used at the same level as place cell information, contrary to the usual models of information flow, in which head direction information promotes grid cell formation, which later on drives place cells. Despite the fact that the prevailing view is that the place cells do not encode head direction, recent data from rats and bats suggest possible head-directional tuning of the hippocampal place cells. For example, single neurons in the hippocampus of bats showed sensitivity to both, the animal's spatial location and its head direction (Rubin, Yartsev, & Ulanovsky, 2014). In rats, directionality bias has been observed experimentally in a subset of cells (Navratilova, Hoang, Schwindel, Tatsuno, & McNaughton, 2012). Additionally, grid cells in the deep MEC layers were co-localized with head-direction cells and conjunctive grid $\times$ head-direction cells (Sargolini et al., 2006). Furthermore, retrograde optogenetic experiments found that the broad spectrum of entorhinal cell types, including grid cells, border cells, and head-direction cells, projects to the hippocampal place cell population (Zhang et al., 2013). Also, the firing of hippocampal place cells and thalamic head direction cells were strongly coupled when rats exposed to unstable external cues (Knierim, Kudrimoti, & McNaughton, 1995). Thus, the combination of place cell and head direction cells could be interpreted as adding directionality bias to all place cells. In this case, place cells that are combined with less specific head direction cells would be non directional place cells, where as those combined with more specific head direction cells would be directional place cells, as those found rats performing corridor tasks (Brunel & Trullier, 1998).

2.5. Flow of events

Fig. 5 shows the flow of events of a single cycle. A cycle starts with the robot reasoning about its next motion and ends after it has moved and learned about the outcome of that last move.

First, the robot location is acquired. In the case of simulation, this information is provided by the simulator itself. In the case of the physical robot experiments, this data is provided by an implemented Fast-SLAM (Montemerlo, Thrun, Koller, & Wegbreit, 2002, 2003) system.

Then, the location information, in the form of (x, y, θ), is used to compute the firings of place cells and head direction cells. Eqs. (1) and (2) are used in this step.

After that, the PC and HDC information is combined and fed to the Q-Learning action selection algorithm. This step uses Eq. (5) to compute a value for each possible action (move forward, rotate left, rotate right).

In parallel, all other cooperative behaviors are executed. The relevant information is fed to them and they assign a value for each possible action.

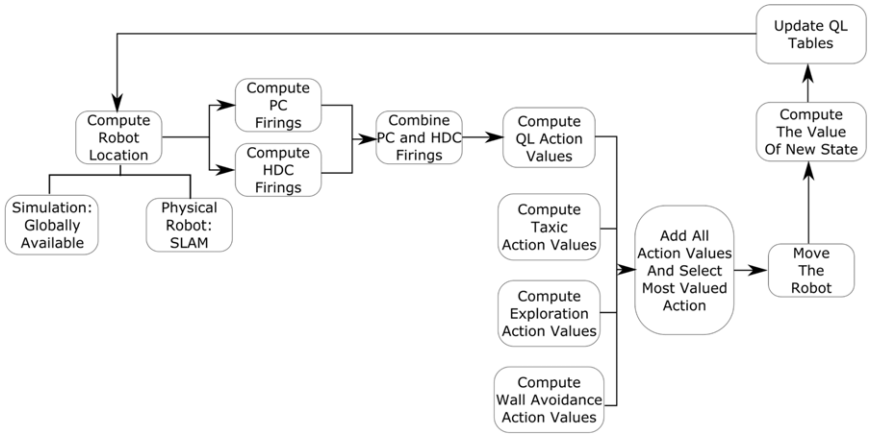


Fig. 5. Flow of events for the navigation model.

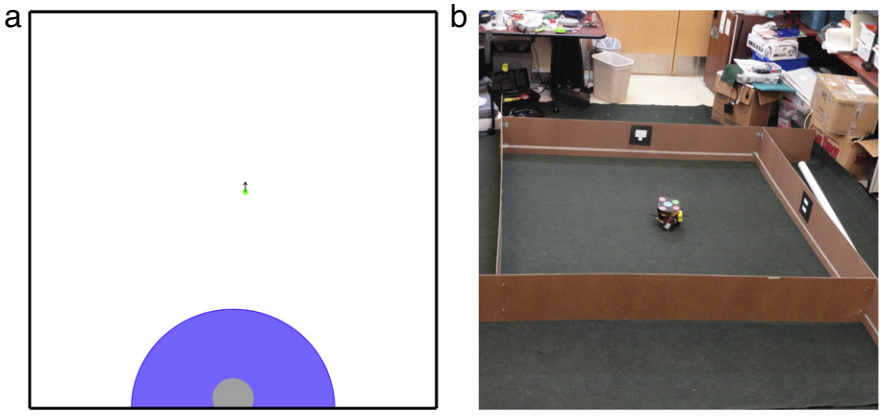


Fig. 6. Experiment 1 and 2 setup. Experiment 1 (a): Morris square dry maze. The gray circle at the bottom represents the goal, the blue semicircle the region where the goal is visible, within a 0.4 m radius, and the green dot and arrow the initial position and orientation. Experiment 2 environment (b): the robot is in the initial position and orientation with a patch on top for the SSL vision system to recognize. The black and white squares are the ARToolkit markers used by the SLAM system. The white line stripes at a fixed height used for wall detection is also shown. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Then, all action values are added and the most valued action is selected. Note that this deterministic maximization process does not mean that decisions are deterministic, since they include the values assigned by the exploration module, which are stochastic.

After selection the action, the robot is moved.

Finally, the arrived state is observed and QL tables are updated, according to Eq. (7). Notice that the computation of the arrived state involves observing the location, computing PC and HDC firings and combining the information again. This has been left out of the diagram for the sake of simplicity.

3. Experiments

3.1. The goal-oriented navigational task

We chose a dry version of the Morris (1981) water maze as our testbed for the navigation model. In this task, the robot navigates an 2 m $\times$ 2 m square environment to go from the initial position to a fixed location goal. Once the robot reaches that interest point, an episode is considered finished. Many episodes are needed for the robot to learn a suitable navigation policy that will take it to the goal faster.

The robot is able to make three types of movements: turn right, go forward or turn left. After a turn, a forward motion is followed. In the presence of obstacles, subsequent turns are made until a forward motion can be carried out.

For our experiments, the individual was always put in the middle of the field, facing in the opposite direction to the goal.

The goal position is visible to the animal within a 0.4 m radius. Fig. 6 illustrates the experimental setup.

3.2. The experiments

We first tested the algorithm on a continuous and stochastic simulated environment to evaluate the multi-scale algorithm's performance in comparison to a single layer model. Then, we applied the policies learned during the simulation to a physical robot navigation task, to validate the system performance under real environment conditions.

A more detailed description of each experiment is included next, while table Table 2 summarizes the main characteristics for all experiments. A description of the implemented robot programs for the experiments is included at the end of this section.

3.2.1. Experiment 1: simulated robot task

In this experiment we compared the performance of the proposed learning algorithm using three scales at once and using each of those scales of representation separately. The single scale systems were implemented using the soft state QL with only one scale for all place cells.

Table 3(a) summarizes the groups used in this experiments with their place field diameters.

Table 2
Experiment summary.

Experiment	Domain	Action outcome	Simulated/Physical	Location information
1	Continuous	Stochastic	Simulated	Global Vision
2	Continuous	Stochastic	Physical	Local Vision

Table 3
Experiment 1 and 2 groups.

(a) Experiment 1 group parameters	
Group	Place field diameter
Multi-scale	(0.1, 0.3, 0.6)
Small scale	0.1
Medium scale	0.3
Large scale	0.6
(b) Experiment 2 groups	
Group	Description
Learned	Policy learned through simulation and executed in the physical robot.
Naive	The physical robot with no policy learned.

Noise was added to the outcome of each movement performed by the robot. The added noise was sampled from a uniform distribution in the interval $[0, .2 * m]$ where m is the magnitude of the movement, i.e. meters traveled or radians turned. Namely, a 0%–20% noise was added to each movement. This was done in order to simulate physical robot conditions more accurately.

No noise was added to the position information provided to the robot.

Fig. 6 shows the experimental setup. The robot started always from the same position and the goal position was fixed. An experiment consisted on a number of episodes that ended when the robot was able to reach the goal. One hundred different individuals were simulated, with 25 episodes each.

3.2.2. Experiment 2: physical robot tests

This experiment consisted of performing the same goal reaching task using a physical robot.

Fig. 6 shows the testing environment. It consisted of a 2×2 m side square with small walls. In each side, an artificial marker was placed for the robot to use as landmark in a small Fast-SLAM system. In the physical robot experiment, position information was derived from local sensory information, as opposed to using global information like it was done in the simulation tests.

Each wall also included a white line stripe at a known height to allow the robot to derive the distance to them using monocular vision.

A differential robot powered by AX-12 motors was used. It used a BeagleBone Black single-board computer and a web camera as its only sensor. All sensorial information was processed on-board, including landmark and wall detection. Piloting algorithms and self-motion computations were run on-board as well. Finally, the implemented visual Fast-SLAM system was also run on-board. The robot program with the MSQL algorithm, however, was run off-board on a personal computer, connected to the robot using a bluetooth network (PAN).

A global camera recorded the robot position at each iteration using the Small Size League vision software (Zickler, Laue, Birbach, Wongphati, & Veloso, 2009). This information, however, was not made available to the robot program during the decision making process.

A policy was learned during 25 simulated iterations. Then, the policy was loaded into the robot and one episode was performed.

As a comparison, a robot without knowledge was put to perform the same task.

Table 3(b) shows the groups used in this experiment.

Each individual started at the center of the maze, as shown in Fig. 6. Before the beginning of the experiment, an initial routine of twelve 90 degree rotations was performed. This allowed the SLAM system to build a stable map in a similar coordinate system as the global camera, because the coordinate frame is not determined by the initial position of the robot, but by the map of the surviving particles in the Fast-SLAM algorithm. The episode ended when the robot was within a radius of 0.4 m of the goal.

The SLAM system position was fed to the place cell layer to compute the firing values at each iteration. Namely, the location information used to determine the firing values of the artificial place cells was derived from local sensory information only.

Six different individuals were used for each group.

4. Results

4.1. Experiment 1

Fig. 7 shows the average number of steps needed to reach the intended goal as a function of the episode number. The default model parameters were used for this experiment, namely $\beta = 1$ (exploration decay) and $\alpha = .9$ (learning rate). Standard deviation per repetition is also shown.

An ANOVA test was done for each group to compare the completion times for episodes 1 and 100. A statistical difference was found in all cases ($p < 0.05$).

In order to assess the impact of the exploration decay parameter on the robot performance we tested different configurations. Fig. 8 shows the time to reach the goal for all groups over 100 individuals, for a fixed α of 0.9 and different β values.

The groups Multi-Scale and Large Scale behaved similarly, so we include a Table 4 with results of an ANOVA test and Tukey HSD post-hoc for each β value. The tests were carried out using the completion times for the second half of the episode, i.e. the last 50 trials.

Adding to this, we include sample paths from different values of β, for different groups, at different episodes.

We include below a plot of the agent policy for some individuals of different groups. In order to plot the policy, a grid of sampling points was laid out over the environment. Then, the simulated robot was placed in every point and the orientation was varied. For each orientation, the value of the most valued action was taken. The orientation that gave the maximum expected value was plotted in each sampling point.

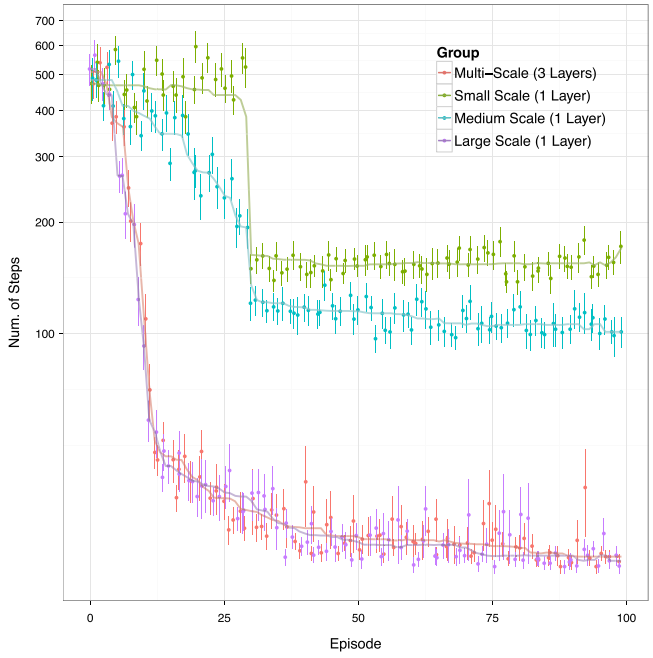


Fig. 7. Number of steps to reach the goal as a function of the episode number for all groups of Experiment 1.

Fig. 10 shows policies acquired after the 20th repetition for the Multi-Scale group for exploration values of 1 and 0.

Table 4
ANOVA test and Tukey HSD post-hoc results for the Multi-Scale and Large Scale groups, for all experiments varying β.

β	Mean difference (Large–Multi)	p value
1	−0.65	0.997
0.5	8.56	0.00057
0.25	2.34	0.608
0.125	5.75	0.017
0.0625	6.51	0.0033
0.03125	3.82	0.078
0	4.31	0.05

We also executed the experiment for different values of the learning rate parameter α. Fig. 11 includes the finishing times for each episode, averaged over 100 individuals, for all groups.

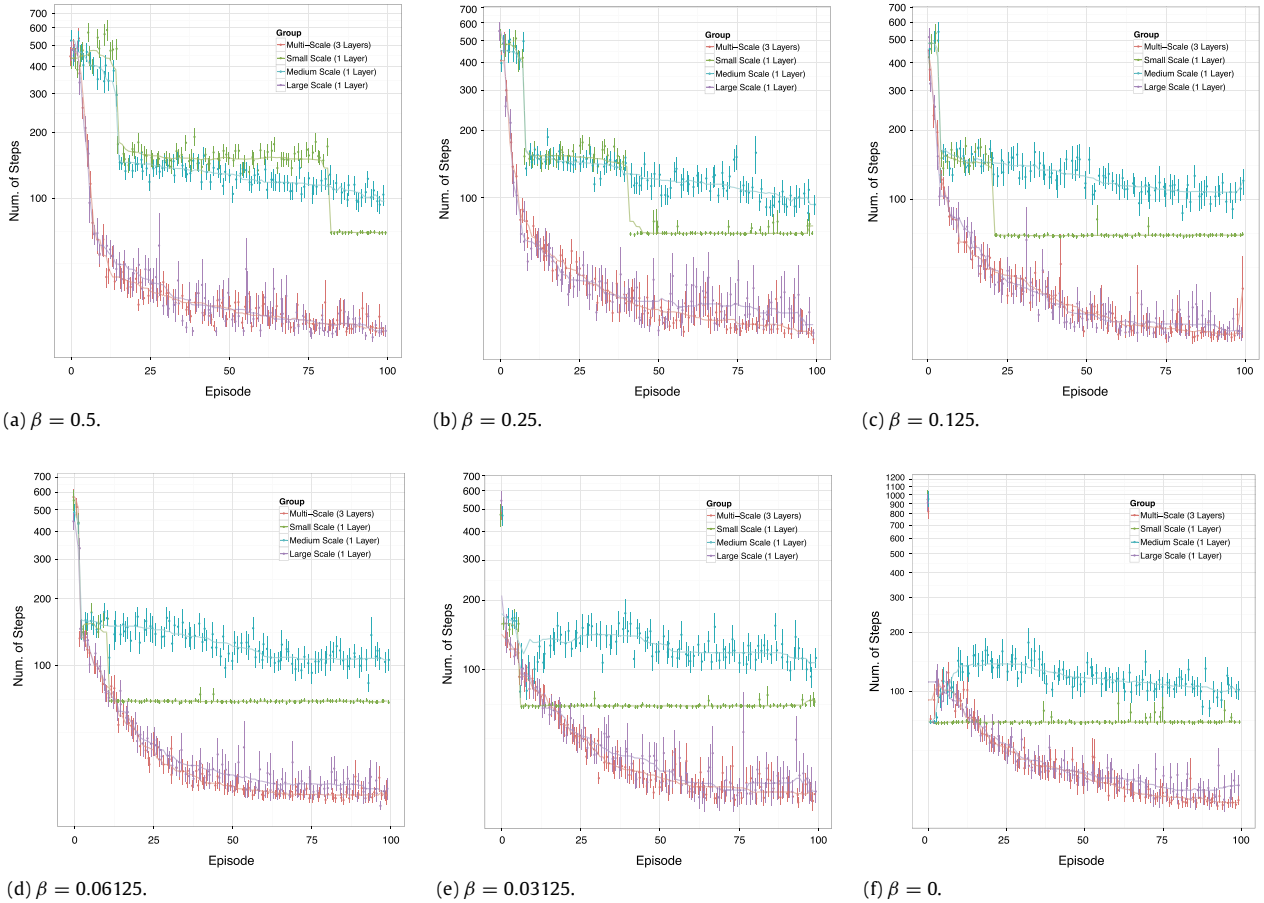
Fig. 12 shows two sample paths corresponding to episode 50 for one individual of the Multi-Scale group, for α values of 0.4 and 0.9.

4.2. Experiment 2

Fig. 13 shows the number of steps needed to reach the goal for the simulated learned policy tested on the robot. The Naive group had no learned policy, whereas the Learned group used the policy learned through 25 simulated repetitions.

A t-test was run on this data to check for significant difference of means. The groups means were found to be significant ($p < 0.05$).

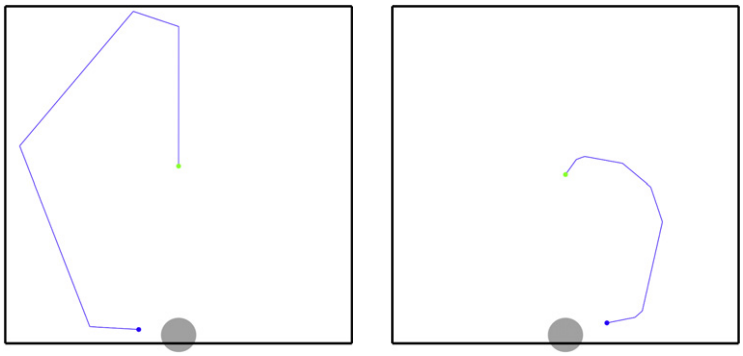
Fig. 14 shows two sample paths, one from each group.



(a) $\beta = 0.5$. (b) $\beta = 0.25$. (c) $\beta = 0.125$.

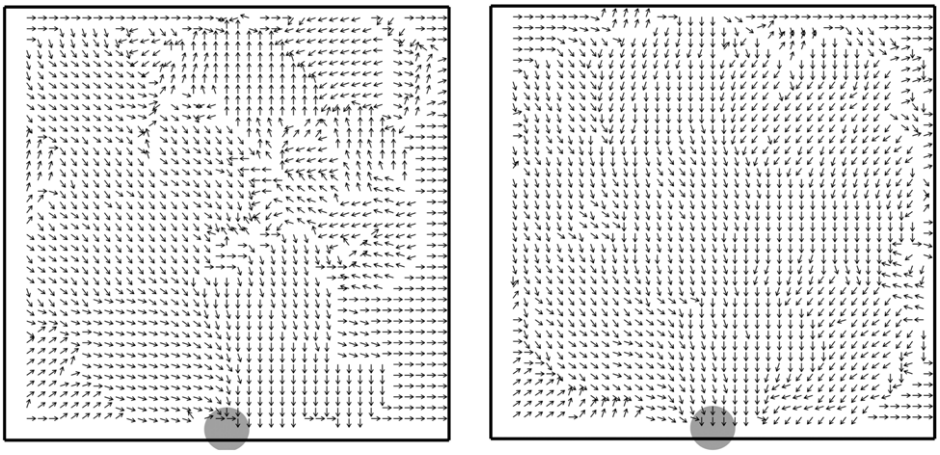
(d) $\beta = 0.06125$. (e) $\beta = 0.03125$. (f) $\beta = 0$.

Fig. 8. Task completion times for 100 individuals per group for different values of the exploration decay factor β. The learning rate α was fixed at 0.9.



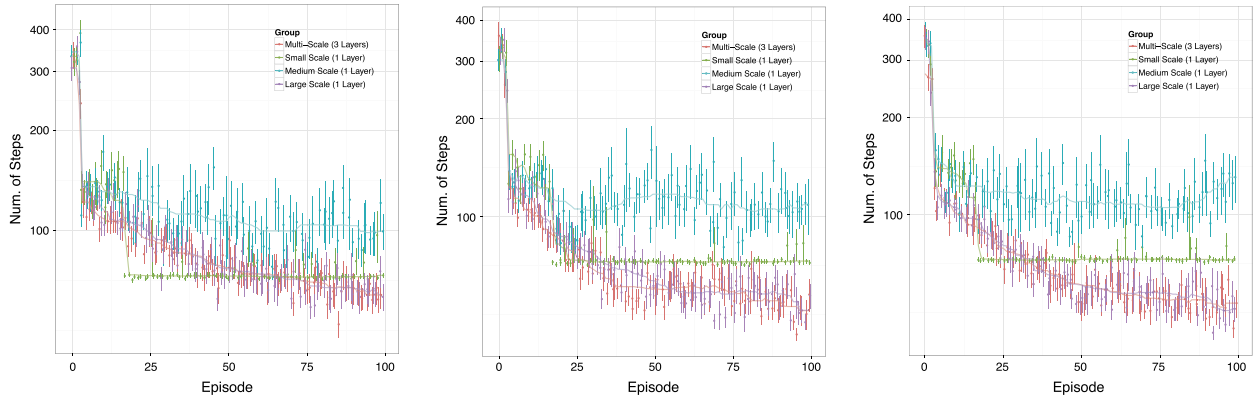
(a) $\beta = 0$, Small scale, episode 100. (b) $\beta = 1$, Multi-scale, episode 96.

Fig. 9. Sample paths from individuals from different groups, for different values of the exploration decay factor β.



(a) $\gamma = 0$. (b) $\gamma = 1$.

Fig. 10. Policy of a sample individual of the Multi-Scale group (b) after the 20th repetition.



(a) $\alpha = 0.4$. (b) 8. (c) $\alpha = 0.9$.

Fig. 11. Task completion times for 100 individuals per group for different values of the learning rate parameter α. The exploration decay parameter β was fixed at .125.

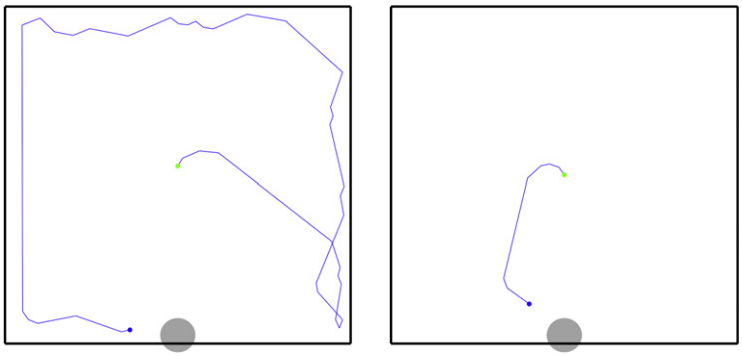
5. Discussion

5.1. Experiment 1

An initial experiment using the model default parameters showed faster learning and a better asymptotic solution for the Large Scale and Multi-Scale groups than for other groups.

Tests were carried out to assess the impact of the exploration parameter β on the completions time and several interesting phenomena were observed. A sudden drop in completion times can be observed in almost all cases. This drop is more noticeable for the Small and Medium Scale groups. This drop seems to occur earlier as the exploration decay velocity is increased. We attribute this to the fact that exploration is done by assigning a decaying value to a random action. Then, when action learned values meet this decaying value, the learned policy takes control and no more time is wasted in exploration.

In the case of the Small Scale and Medium Scale groups, there was a second drop in completion times late in the experiment. Our model assigns an infinitesimal value to forward motions to favor



(a) $\alpha = 0.4$. (b) $\alpha = 0.9$.

Fig. 12. Sample paths from one Multi-Scale individual at episode 50 for different values of the learning rate α.

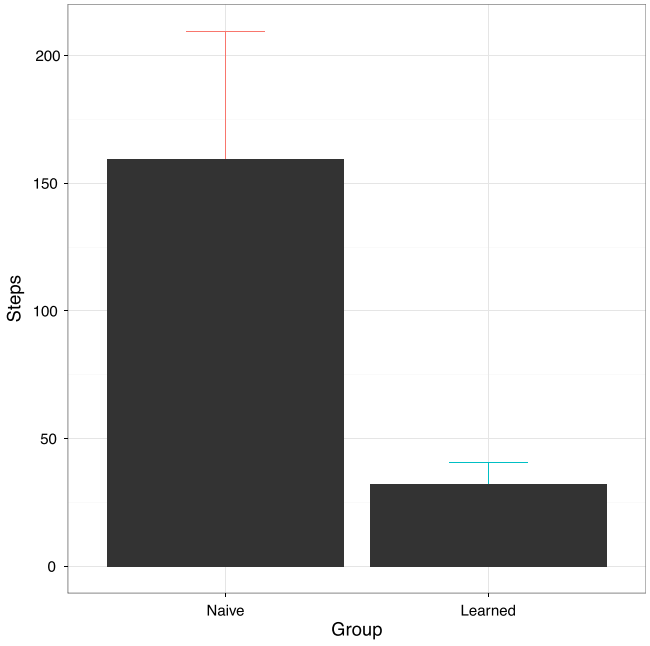


Fig. 13. Experiment completion time for the real robot trials for the Naive and Learned groups.

them over rotations in case of a complete tie. When the exploration value decays below this value, forward motions are favored over rotations. Then, the strategy turns into a route navigation (Redish, 1999) one, in which the simulated robot went forward until it reached a wall, turned and repeated the action, as seen in Fig. 9a. For a given number of place fields, the Small and Middle layers are not able to cover as much of the environment as the other two groups. The Small and Medium groups might therefore not be able to learn across all the whole field and perform actions that would take them out of this suboptimal route navigation strategy. This is supported by the fact that the Medium Scale group shows an increasing tendency to default to this behavior, showing a second decay in completion times, as exploration decays faster.

The Large Scale and Multi-Scale groups, on the other hand, showed more optimized learned paths, such as the one shown in Fig. 9b.

Differences in the learned policies for different values of exploration can be appreciated. Slower exploration decay allowed the simulated robot to explore more of the environment, learning the proper action to perform at every possible point. This resulted in more coherent policies.

Additionally, we observed that the Large Scale and Multi-Scale groups behaved similarly for all exploration parameters. They showed a faster initial learning (first 10 episodes) than the other two groups for all exploration parameter values. However, this relation was inverted once the other groups fell into praxic strategies. Statistical analysis showed some significant, but slight, differences in the average completion time– for some exploration parameter values, favoring the Multi-Scale group.

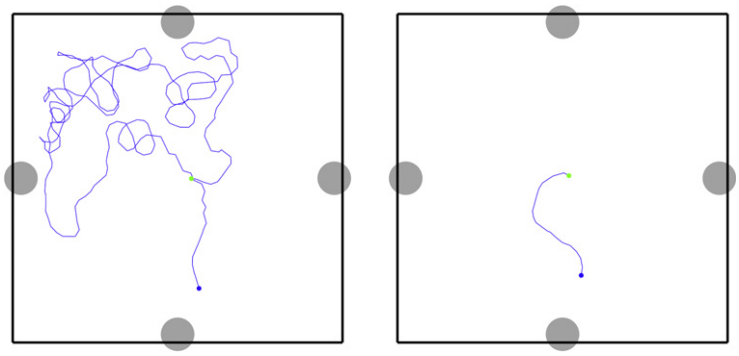
One interesting phenomena with regards to exploration could be observed. It would seem that by decreasing exploration, learning speed is increased. Specially in the case of the small scale system, which reaches a stable suboptimal solution sooner, as the exploration energy is reduced. However, we observe that the navigation problem has a suboptimal but simple praxic solution, as the one shown in Fig. 9. Namely, it can be solved by always going forward and turning at the walls always in the same direction. This solution could be learned fast by the model, as it only needed to learn the advantages of going forward and acquire a turning bias. Then, when exploration energy was low and no better solution had been learned, the individuals tended to use this solution, which despite being suboptimal, it allows the animat to reach the platform quite fast.

Variations in the learning rate parameter showed the greater learning potential of the Large and Multiple Scale groups. At higher values of α, these groups were able to continue learning until they improved completion times with respect to those achieved by the Small and Medium Scale groups.

Summarizing, this experiment has shown that large and multi-scale representations serve as a better source of information for this navigation task. We attribute this to the fact that they provide a better coverage per place field and that they allow for faster generalization of the learned value of a certain region. This goes in line with the fact that the ventral portion of the hippocampus has more projection to value estimation regions, such as the ventral tegmental area (Arszovszki et al., 2014; Gasbarri et al., 1994). Namely, the presence of large place fields allows for a faster propagation of the reward values to the rest of the environment. The fact that the less accurate cells are a better source of information for navigational decision making could seem counter intuitive. However, the fact that many fields overlap in a given place and that the value of an action is computed as a linear combination seems to compensate for the lack of precision of each individual field. The finding that the Multi-Scale group is significantly better for most exploration parameter values indicates that the presence of small cells is also beneficial.

5.2. Experiment 2

Experiment 2 was challenging to the algorithm for various reasons. This experiment involved the use of local sensory



(a). (b).

Fig. 14. Sample paths for the robot showing the position reported by the global camera system. The additional gray circles (north, east and west) indicate the position of the landmarks used by the SLAM system.

information to derive the robot location. Thus, the algorithm had to cope with noise in the reported position when making action selection decisions. What is more, there could be noise in the learning process when pose correction events occurred in the underlying SLAM system. After performing a single motion, the robot could find itself suddenly far away from its original position, due to relocalization in the SLAM system. Then, the algorithm would erroneously update the value of the performed action in the previous state.

The levels of motion noise of the real robot were also greater than those used in the simulator.

Despite all these unreliabilities, the physical robot was able to execute a policy learned in simulation and significantly reduce completion times.

The path reported by the global camera system shows how the learned robot was able to reach the target faster (Fig. 14).

6. Conclusions

We presented in this paper a multi-scale hippocampal place cell space representation model used in goal-directed robot navigation. The multi-scale model was contrasted against single scale place cell models during robot navigation in a circular arena containing a fixed goal. Reinforcement learning was used to train the robot to find the goal after several episodes.

Larger and combined scales of representation proved to be the best for learning the task, as they allowed a faster initial learning and the potential to continue the learning process to get better solutions than the route navigation strategies learned using smaller scales. Although this might seem counter intuitive, due to the lower specificity of larger scale place cells, these cells allow for a better coverage of the environment with the same amount of cells. The combined multi-scale representation was slightly better than the large representation alone for some combination of parameters.

Additionally, the system was suitable for controlling a robot with noisy location information provided by a SLAM system.

Future work includes testing the fully integrated multi-scale place cell model with the Barrera–Weitzenfeld spatial cognition model that integrates idiothetic with allothetic information when evaluating the goal-oriented task under the circular arena. The current model does not perform the computation of place cells firing from the combination of idiothetic and allothetic information, but artificially computes place cell firing rates from the position given by a global positioning system, in simulation, or in a SLAM system, in the physical robot. Integration with the work by Tejera et al. (2013) that correlates place cells field firing to multi-scale grid cells based on a linear oscillatory interference model is also in our plans.

Performing a parameter space search is also due. This would allow us to assess the impact on performance of varying each of the involved parameters, such as the number of place and head direction cell layers, minimum and maximum widths and the learning rate.

In terms of goal-oriented navigational task, we plan to test under arenas that include fixed and dynamic obstacles. This work is currently being contrasted to results obtained from similar rat experiments (Fellous Laboratory). The fact that place cell ensemble activity drives the actions of the robot, poses an interesting challenge for obstacle avoidance and path relearning. It has been shown that when obstacles are introduced in a region where a place cell was firing, the place field vanishes altogether (Muller & Kubie, 1987). This could mean that the policies are automatically updated in the presence of new obstacles, avoiding to try to take paths that traverse them. Our model predicts that inhibition of the ventral portion of the hippocampus bilaterally during the learning or relearning phases of a navigation experiment could decrease learning performance. This prediction would have to be made using a task that is known to functionally involve the ventral hippocampus, perhaps with obstacles and more challenging task requirements. Further experimental and theoretical work is needed to assess the extent to which multi-scale encoding is required for spatial navigation with obstacles.

Finally, future work includes increasing the level of detail of the navigational model. Our current model works by the interaction of the hippocampus and striatum for locale navigation (Redish, 1999; Touretzky, 2002) and a source of reinforcement information like the VTA for learning. However, we currently model this with high level of abstraction, using an actor-critic module. We plan to increase the level of detail of our hippocampus model, to include grid cell to place cell interaction, complex hippocampal dynamics, explicit visual cue information integration and barrier information integration (Touretzky & Muller, 2006). We also plan to substitute the actor-critic module with an actual neural implementation of the striatum–VTA interaction.

Acknowledgments

This work is funded by NSF IIS Robust Intelligence research collaboration grant #1117303 at USF and U. Arizona entitled "Investigations of the Role of Dorsal versus Ventral Place and Grid Cells during Multi-Scale Spatial Navigation in Rats and Robots," and supported in part by the "Agencia Nacional de Investigacion e Innovación (ANII) " grant POS_EXT_2011_1_3998.

References

Arleo, A., Smeraldi, F., & Gerstner, W. (2004). Cognitive navigation based on nonuniform Gabor space sampling, unsupervised growing networks, and reinforcement learning. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, *15*(3), 639–652.

Arszovszki, A., Borhegyi, Z., & Klausberger, T. (2014). Three axonal projection routes of individual pyramidal cells in the ventral CA1 hippocampus. *Frontiers in Neuroanatomy*, *8*, 53.

Barrera, A., Tejera, G., Llofriu, M., & Weitzenfeld, A. (2015). Learning spatial localization: from rat studies to computational models of the hippocampus. *Spatial Cognition & Computation*, *15*(1), 27–59.

Barrera, A., & Weitzenfeld, A. (2008). Biologically-inspired robot spatial cognition based on rat neurophysiological studies. *Autonomous Robots*, *25*(1–2), 147–169.

Blumberg, M. S. (2015). The developmental origins of spatial navigation: Are we headed in the right direction? *Trends in Neurosciences*, *38*(2), 67–68.

Brown, M. A., & Sharp, P. E. (1995). Simulation of spatial learning in the Morris water maze by a neural network model of the hippocampal formation and nucleus accumbens. *Hippocampus*, *5*(3), 171–188.

Brunel, N., & Trullier, O. (1998). Plasticity of directional place fields in a model of rodent CA3. *Hippocampus*, *8*(6), 651–665.

Brun, V. H., Solstad, T., Kjelstrup, K. B., Fyhn, M., Witter, M. P., Moser, E. I., & Moser, M.-B. (2008). Progressive increase in grid scale from dorsal to ventral medial entorhinal cortex. *Hippocampus*, *18*(12), 1200–1212.

Burgess, N., Recce, M., & O'Keefe, J. (1994). A model of hippocampal function. *Neural Networks*, *7*(6–7), 1065–1081.

Caluwaerts, K., Staffa, M., N'Guyen, S., Grand, C., Dollé, L., Favre-Félix, A., Girard, B., & Khamassi, M. (2012). A biologically inspired meta-control navigation system for the Psikharpax rat robot. *Bioinspiration & Biomimetics*, *7*(2), 025009.

Giocomo, L. M., Stensola, T., Bonnevie, T., Van Cauter, T., Moser, M.-B., & Moser, E. I. (2014). Topography of head direction cells in medial entorhinal cortex. *Current Biology: CB*, *24*(3), 252–262.

Guger, C., Gener, T., Pennartz, C. M. A., Brotons-Mas, J. R., Edlinger, G., Bermúdez I Badia, S., Verschure, P., Schaffelhofer, S., & Sanchez-Vives, M. V. (2011). Real-time position reconstruction with hippocampal place cells. *Frontiers in Neuroscience*, *5*, 85.

Chavarriaga, R., Strösslin, T., Sheynikhovich, D., & Gerstner, W. (2005). A computational model of parallel navigation systems in rodents. *Neuroinformatics*, *3*(3), 223–241.

Chen, Z., Jacobson, A., Erdem, U., Hasselmo, M. E., & Milford, M. (2013). Towards bio-inspired place recognition over multiple spatial scales. In *Australasian conference on robotics and automation*.

Cox, B., & Krichmar, J. (2009). Neuromodulation as a robot controller. *IEEE Robotics Automation Magazine*, *16*(3), 72–80.

de Jong, L. W., Gereke, B., Martin, G. M., & Fellous, J.-M. (2011). The traveling salesrat: insights into the dynamics of efficient spatial navigation in the rodent. *Journal of Neural Engineering*, *8*(6), 065010.

Dollé, L., Sheynikhovich, D., Girard, B., Chavarriaga, R., & Guillot, A. (2010). Path planning versus cue responding: a bio-inspired model of switching between navigation strategies. *Biological Cybernetics*, *103*(4), 299–317.

Erdem, U. M., & Hasselmo, M. E. (2014). A biologically inspired hierarchical goal directed navigation model. *Journal of Physiology-Paris*, *108*(1), 28–37.

Filliat, D., & Meyer, J.-a. (2002). *Global localization and topological map-learning for robot navigation*.

Foster, D. J., & Wilson, M. A. (2006). Reverse replay of behavioural sequences in hippocampal place cells during the awake state. *Nature*, *440*(7084), 680–683.

Fyhn, M., Molden, S., Witter, M. P., Moser, E. I., & Moser, M.-B. (2004). Spatial representation in the entorhinal cortex. *Science*, *305*(5688), 1258–1264.

Gasbarri, A., Packard, M. G., Campana, E., & Pacitti, C. (1994). Anterograde and retrograde tracing of projections from the ventral tegmental area to the hippocampal formation in the rat. *Brain Research Bulletin*, *33*(4), 445–452.

Gaussier, P., Revel, A., Banquet, J. P., & Babeau, V. (2002). From view cells and place cells to cognitive map learning: processing stages of the hippocampal system. *Biological Cybernetics*, *86*(1), 15–28.

Geva-Sagiv, M., Las, L., Yovel, Y., & Ulanovsky, N. (2015). Spatial cognition in bats and rats: from sensory acquisition to multiscale maps and navigation. *Nature Reviews Neuroscience*, *16*(2), 94–108.

Guanella, A., & Verschure, P. F. M. J. (2007). Prediction of the position of an animal based on populations of grid and place cells: a comparative simulation study. *Journal of Integrative Neuroscience*, *6*(3), 433–446.

Guazzelli, A., Bota, M., Corbacho, F. J., & Arbib, M. A. (1998). Affordances. motivations, and the world graph theory. *Adaptive Behavior*, *6*(3–4), 435–471.

Hafting, T., Fyhn, M., Molden, S., Moser, M.-B., & Moser, E. I. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature*, *436*(7052), 801–806.

Hales, J. B., Schlesiger, M. I., Leutgeb, J. K., Squire, L. R., Leutgeb, S., & Clark, R. E. (2014). Medial entorhinal cortex lesions only partially disrupt hippocampal place cells and hippocampus-dependent place memory. *Cell Reports*, *9*(3), 893–901.

Jensen, O., & Lisman, J. E. (2000). Position reconstruction from an ensemble of hippocampal place cells: contribution of theta phase coding. *Journal of Neurophysiology*, *83*(5), 2602–2609.

Johnson, A., & Redish, A. D. (2007). Neural ensembles in CA3 transiently encode paths forward of the animal at a decision point. *The Journal of Neuroscience*, *27*(45), 12176–12189.

Jung, M. W., Wiener, S. I., & McNaughton, B. L. (1994). Comparison of spatial firing characteristics of units in dorsal and ventral hippocampus of the rat. *The Journal of Neuroscience*, *14*(12), 7347–7356.

Keinath, A. T., Wang, M. E., Wann, E. G., Yuan, R. K., Dudman, J. T., & Muzzio, I. A. (2014). Precise spatial coding is preserved along the longitudinal hippocampal axis. *Hippocampus*, *24*(12), 1533–1548.

Kjelstrup, K. B., Solstad, T., Brun, V. H., Hafting, T., Leutgeb, S., Witter, M. P., Moser, E. I., & Moser, M.-B. (2008). Finite scale of spatial representation in the hippocampus. *Science*, *321*(5885), 140–143.

Knierim, J. J., Kudrimoti, H. S., & McNaughton, B. L. (1995). Place cells, head direction cells, and the learning of landmark stability. *The Journal of Neuroscience*, *15*(3), 1648–1659.

Kober, J., Bagnell, J. A., & Peters, J. (2013). Reinforcement learning in robotics: A survey. *International Journal of Robotics Research*, page 0278364913495721.

Krichmar, J. L., Nitz, D. A., Gally, J. A., & Edelman, G. M. (2005). Characterizing functional hippocampal pathways in a brain-based device as it solves a spatial memory task. *Proceedings of the National Academy of Sciences of the United States of America*, *102*(6), 2111–2116.

Krichmar, J. L., & Röhrbein, F. (2013). Value and reward based learning in neurorobots. *Frontiers in Neurorobotics*, *7*.

Lansink, C. S., & Pennartz, C. M. A. (2015). Associative reactivation of place–reward information in the hippocampal–ventral striatal circuitry. In M. Tatsuno (Ed.), *Springer series in computational neuroscience: vol. 12. Analysis and modeling of coordinated multi-neuronal activity* (pp. 81–104). New York: Springer.

Lisman, J. E., & Grace, A. A. (2005). The hippocampal-VTA loop: controlling the entry of information into long-term memory. *Neuron*, *46*(5), 703–713.

Long, L. L., Bunce, J. G., & Chrobak, J. J. (2015). Theta variation and spatiotemporal scaling along the septotemporal axis of the hippocampus. *Frontiers in Systems Neuroscience*, *9*, 37.

Maurer, A. P., Vanrhoads, S. R., Sutherland, G. R., Lipa, P., & McNaughton, B. L. (2005). Self-motion and the origin of differential spatial scaling along the septo-temporal axis of the hippocampus. *Hippocampus*, *15*(7), 841–852.

Milford, M., & Wyeth, G. (2007). Spatial mapping and map exploitation: A bio-inspired engineering perspective. In S. Winter, M. Duckham, L. Kulik, & B. Kuipers (Eds.), *Lecture notes in computer science: vol. 4736. Spatial information theory* (pp. 203–221). Berlin Heidelberg: Springer.

Milford, M., & Wyeth, G. (2009). Persistent navigation and mapping using a biologically inspired SLAM system. *International Journal of Robotics Research*,.

Montemerlo, M., Thrun, S., Koller, D., & Wegbreit, B. (2002). FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the aaai national conference on artificial intelligence*, pp. 593–598. AAAI.

Montemerlo, M., Thrun, S., Koller, D., & Wegbreit, B. (2003). FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In Proc. of the Int. Conf. on Artificial Intelligence (IJCAI), pp. 1151–1156.

Morris, R. G. (1981). Spatial localization does not require the presence of local cues. *Learning and Motivation*, *12*(2), 239–260.

Muller, R. U., & Kubie, J. L. (1987). The effects of changes in the environment on the spatial firing of hippocampal complex-spike cells. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, *7*(7), 1951–1968.

Navratilova, Z., Hoang, L. T., Schwindel, C. D., Tatsuno, M., & McNaughton, B. L. (2012). Experience-dependent firing rate remapping generates directional selectivity in hippocampal place cells. *Frontiers in Neural Circuits*, *6*, 6.

O'Keefe, J., & Burgess, N. (1996). Geometric determinants of the place fields of hippocampal neurons. *Nature*, *381*(6581), 425–428.

O'Keefe, J., & Dostrovsky, J. (1971). The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, *34*(1).

O'Keefe, J., & Nadel, L. (1978). *The hippocampus as a cognitive map*. Oxford : New York: Oxford University Press.

Pata, D. S., Escuredo, A., Lallée, S., & Verschure, P. F. M. J. (2014). Hippocampal based model reveals the distinct roles of dentate gyrus and CA3 during robotic spatial navigation. In A. Duff, N. F. Lepora, A. Mura, T. J. Prescott, & P. F. M. J. Verschure (Eds.), *Lecture notes in computer science: vol. 8608. Biomimetic and biohybrid systems* (pp. 273–283). Springer International Publishing.

Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79.

Recce, M., & Harris, K. D. (1996). Memory for places: a navigational model in support of Marr's theory of hippocampal function. *Hippocampus*, *6*(6), 735–748.

Redish, A. D. (1999). *Beyond the cognitive map: From place cells to episodic memory*. MIT Press.

Redish, A. D., & Touretzky, D. S. (1997). Cognitive maps beyond the hippocampus. *Hippocampus*, 15–35.

Royer, S., Sirota, A., Patel, J., & Buzsáki, G. (2010). Distinct representations and theta dynamics in dorsal and ventral hippocampus. *The Journal of Neuroscience*, *30*(5), 1777–1787.

Rubin, A., Yartsev, M. M., & Ulanovsky, N. (2014). Encoding of head direction by hippocampal place cells in bats. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, *34*(3), 1067–1080.

Sargolini, F., Fyhn, M., Hafting, T., McNaughton, B. L., Witter, M. P., Moser, M.-B., & Moser, E. I. (2006). Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, *312*(5774), 758–762.

Singh, S. P., Jaakkola, T., & Jordan, M. I. (1995). Reinforcement learning with soft state aggregation. In *Advances in neural information processing systems 7* (pp. 361–368). MIT Press.

Strange, B. A., Witter, M. P., Lein, E. S., & Moser, E. I. (2014). Functional organization of the hippocampal longitudinal axis. *Nature Reviews. Neuroscience*, *15*(10), 655–669.

Sukumar, D., Rengaswamy, M., & Chakravarthy, V. S. (2012). Modeling the contributions of Basal ganglia and Hippocampus to spatial navigation using reinforcement learning. *PloS One*, *7*(10), e47467.

Sutton, R. S., & Barto, A. G. (1998). Reinforcement learning: an introduction. In *Adaptive computation and machine learning*. Cambridge, Mass: MIT Press.

Tejera, G., Barrera, A., Fellous, J.-M., Llofriu, M., & Weitzenfeld, A. (2013). Spatial cognition: robot target localization in open arenas based on rat studies. In *SPIE defense, security, and sensing*. International Society for Optics and Photonics, pp. 875600–875600.

Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review*, *55*(4), 189–208.

Touretzky, D. S. (2002). The rodent navigation circuit. In P. E. Sharp (Ed.), *The neural basis of navigation* (pp. 217–233). US: Springer.

Touretzky, D. S., & Muller, R. U. (2006). Place field dissociation and multiple maps in hippocampus. *Neurocomputing*, *69*(10–12), 1260–1263.

Weitzenfeld, A., Arbib, M. A., & Alexander, A. (2002). *The neural simulation language: A system for brain modeling*. Cambridge, MA, USA: MIT Press.

Weitzenfeld, A., Gutierrez-Nolasco, S., & Venkatasubramanian, N. (2003). MIRO: An embedded distributed architecture for biologically inspired mobile robots. In *ICAR-03*, Coimbra, Portugal.

Wilson, M. A., & McNaughton, B. L. (1993). Dynamics of the hippocampal ensemble code for space. *Science*, *261*(5124), 1055–1058.

Zhang, K., Ginzburg, I., McNaughton, B. L., & Sejnowski, T. J. (1998). Interpreting neuronal population activity by reconstruction: unified framework with application to hippocampal place cells. *Journal of Neurophysiology*, *79*(2), 1017–1044.

Zhang, S.-J., Ye, J., Miao, C., Tsao, A., Cerniauskas, I., Ledergerber, D., Moser, M.-B., & Moser, E. I. (2013). Optogenetic dissection of entorhinal-hippocampal functional connectivity. *Science*, *340*(6128), 1232627.

Zickler, S., Laue, T., Birbach, O., Wongphati, M., & Veloso, M. (2009). SSL-Vision: The shared vision system for the robocup small size league. In *RoboCup 2009: Robot Soccer World Cup XIII* (pp. 425–436).