To appear in Proc. of European Conference on Mobile Robotics (ECMR), 2019, Prague, Czech Republic.

Towards combining a neocortex model with entorhinal grid cells for mobile robot localization

Stefan Schubert[1], Peer Neubert and Peter Protzel

***Abstract*— Motion and navigation are fundamental abilities of all terrestrial animals. It is essential for foraging, reproduction, and more generally for survival. There are a couple of strategies to conduct navigation from simpler visual homing in ants to more complex and cognitive demanding techniques in mammals. Many species of mammals use several specialized cell types in the hippocampus and the entorhinal cortex to represent space in the brain like head direction cells to encode their orientation and grid cells to keep track of their position.**

In our recent work, we presented MCN – an algorithm that is inspired by working principles of the human neocortex for the navigational subtask visual place recognition. MCN makes decisions based merely on camera data without odometry about whether or not a currently visited place has been seen in the past. In this work, we intend to answer the question if we can combine our neocortex-inspired model with entorhinal cortex cells for space representation to exploit additional metric data like odometry in our system. We believe that the combination of bio-inspired techniques could help someday to create a biologically plausible and more robust navigation system like in animals. In this paper, we give an introduction to our neocortex-inspired algorithm MCN and to two cell types of the entorhinal cortex, answer how these concepts can be combined to perform visual place recognition, and provide proof-of-concept experiments with a mobile robot to show the performance of the proposed system.

I. INTRODUCTION

Navigation is an essential skill for mobile robots to perform exploration and to maneuver in an environment. To address complex aspects of this task and to potentially understand how animals do this, bio-inspired techniques may be used for learning. The subtask visual place recognition tries to answer the question if a currently seen place has already been visited in the past within a changing environment. Visual place recognition is essential for relocalization after position loss or loop closure detection for mapping.

In our recent work [1], [2], we presented MCN (Mini-Column Network): MCN is an adaption of Hierarchical Temporal Memory (HTM), which is a biologically plausible model for aspects of the human *neocortex*, so that ideas from HTM can be used for place recognition. We have shown that MCN performs well on real world datasets in comparison to multiple state-of-the-art algorithms. Further, we ran MCN online on a real mobile robot to perform place recognition in two challenging indoor environments.

MCN currently uses only visual input that limits its potential capabilities. In this paper, we want to make one more step with MCN in the direction of mobile robot navigation. Metric information like odometry data is required to perform tasks like Simultaneous Localization and Mapping (SLAM) in order to build a metric map and to localize within this map. Grid cells represent a structure in the *medial entorhinal cortex* in the rodent brain that represents the position of an individual in space. Since MCN and grid cells are bio-inspired models, it seems reasonable to use both jointly; however, we make no claim to biological plausibility for this combination. In this work we want to answer the question: *How can MCN be combined with entorhinal grid cells to additionally exploit odometry for place recognition?*

To answer this question, we first give a short introduction to MCN in Sec. III and to cell types in the entorhinal cortex in Sec. IV. As the main contribution of this paper, we show in Sec. V how MCN can be combined with grid cells. Subsequently, we provide results in Sec. VI for proof-of-concept experiments with two datasets recorded with a mobile robot to show that (1) a combination of MCN with grid cells basically works and (2) the combination actually benefits from additional odometry data. Finally, we discuss potential future steps in Sec. VII to push our ideas further to a full navigation system.

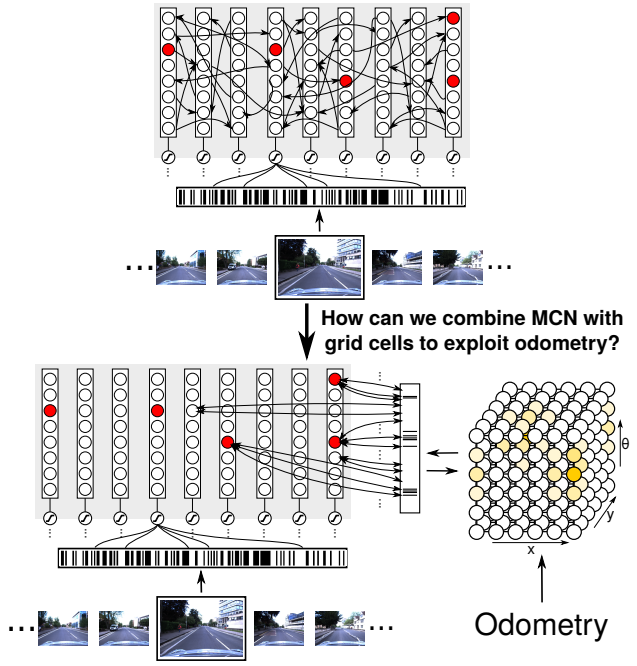


Fig. 1. How can MCN be combined with a grid cell network (GCN) to exploit odometry information? The top shows an MCN with predictive lateral connections (PLCs) whereas the bottom proposes an architecture with predictive sensory input connections (PSICs) to a grid cell network.

[1]All authors are with Faculty of Electrical Engineering and Automation Technology, Chemnitz University of Technology, Chemnitz, Germany stefan.schubert@etit.tu-chemnitz.de

978-1-7281-3605-9/19/$31.00 ©2019 IEEE

II. RELATED WORK

This paper extends a camera-based place recognition approach with odometry information. Visual place recognition is a well studied problem. Lowry et al. [3] provide a recent survey. Usually, basis for visual place recognition is the image similarity based on local landmarks [4], [5] or global descriptors [6], [7]. For place recognition in changing environments, descriptors from early convolutional layers (e.g. conv3) from off-the-shelve CNNs like AlexNet [8] showed impressive results [9]. More recently, CNN descriptors were also particularly designed and trained for place recognition, e.g. NetVLAD [10]. We will use this NetVLAD descriptor in our experiments.

Based on such image processing front-ends, a variety of approaches exists to compare and match images. Beyond simple pairwise comparison and using statistics of feature appearances (e.g. FAB-MAP [11]), the benefit of exploiting sequence information is well accepted in the literature, e.g. [12], [13], [14], [15]. To improve the localization performance of a place recognition approach, additional odometry information can be used [3]. For example, CAT-SLAM [16] uses a particle filter with particle weighting based on local appearance and odometry information. SMART [17] extends the sequential image processing approach from SeqSLAM [12] with additional odometry input. There are various other approaches to visual localization [3]. In the following, we want to focus on brain-inspired approaches.

Madl et al. [18] provide a review of existing computationally implemented cognitive models of spatial representations for navigation. Oess et al. [19] discuss how different reference frames are used in brains and under what conditions they might be used for navigation. Möller et al. [20] presented a biologically inspired navigation algorithm based on visual homing of dessert ants. A similar approach has also been used for camera-based localization in 3d maps [21]. RatSLAM [22] is an approach to simultaneous localization and mapping that builds upon insights about cells in the rodent's brain that participate in navigation, in particular entorhinal grid cells. Recently, Deepmind [23] used grid cell representations to learn vector based navigation in a deep reinforcement learning agent. In this present work, we use a grid cell representation similar to RatSLAM to encode odometry information. In RatSLAM, grid cells are implemented in form of a three dimensional continuous attractor network (CAN) with wrap-around connections; one dimension for each degree of freedom of the robot. The activity in the CAN is moved based on proprioceptive clues of the robot (e.g. wheel encoders) and new energy is injected by connections from local view cells that encode the current visual input, as well as from previously created experiences. The dynamics of the CAN apply a temporal filter on the sensory data. Only in case of repeated consistent evidence for recognition of a previously seen place, this matching is also established in the CAN representation. RatSLAM's exploitation of sequence information allowed to demonstrate impressive navigation results [24]. More details about involved cells will be provided in Sec. IV.

This present work also builds upon our previous work [2] on using Hierarchical Temporal Memory (HTM) [25] for place recognition. HTM is a theory on working principles of the human neocortex. HTM builds upon the assumption of a single learning algorithm that is deployed all over the neocortex [26]. It is a continuously evolving theory with the goal to explain more and more aspects of the neocortex as well as extending the range of practical demonstrations and applications. Currently, these applications include anomaly detection, natural language processing, and object detection [27]. An implementation is available [28]. An important evolution is the extension with location information [27]. Recent work [29] combines HTM with a grid-cell encoding [30] for object recognition. The following Sec. III provides more details on how we used the HTM algorithm for visual place recognition in our previous work [2].

III. MCN IN A NUTSHELL: A NEOCORTEX INSPIRED CELL MODEL FOR PLACE RECOGNITION

In our recent work [2], we introduced the Mini-Column Network (MCN) that is a simpler, modified version of HTM. MCN is designed for visual place recognition, and we have shown that it performs well in simulation, on real world datasets, and online on a mobile robot. In the following explanation, we will refer for clarification to elements in Fig. 1, top, that shows a simple MCN. It has its basic architecture in common with HTM: First, it consists of cells (circles) which are organized in minicolumns (vertical rectangles with circles). Minicolumns are activated by a feed forward connection (lines from barcode to circle), the *spatial pooler*, which gets a binary encoded image as input (barcode). The image is encoded with a deep neural network like NetVLAD and binarized with *sparse locality sensitive binary hashing* (sLSBH) [2]. Further, cells are potentially interconnected with *lateral connections* (lines between circles) that predict the cells' activations for the subsequent timestep; the cells with their predictive connections are termed as *temporal memory*.

In the following, we give a very short overview about the cells' states and state transitions; a more comprehensive description can be found in our recent work [2]. Every cell in each minicolumn can have four different states: inactive, predicted, active, and winner (often in this order). An inactive cell becomes predicted if at least one of its predictive connections becomes active. A predicted cell becomes active if the corresponding minicolumn is activated by the spatial pooler; in this case all active cells become winner cells within this minicolumn. In conclusion, predicted cells in a minicolumn try to predict a potential activation of their minicolumn in the next timestep, so they try to predict a potential next input to the MCN; if the minicolumn does not become active, the corresponding cells simply become inactive. However, if an activated minicolumn has no predicted cells, the minicolumn is *bursted*: all cells become active and one winner cell is chosen.

Differences between HTM and MCN are mainly how the feed forward connections and lateral connections are trained: Basically, all connections in HTM have a *permanence* property; a permanence is a scalar value that defines whether a connection is used or not, depending on if the permanence is above a threshold or not. The permanences are learned in a hebbian way to increase or decrease their value in order to get potentially active or inactive. Connections with a random permanence are randomly created in advance to the input elements for feed forward connections, and in dependence of active predecessor cells for lateral connections. Full details on HTM can be found in [25]. In contrast, feed forward connections and lateral connections in MCN are trained as one-shot learning and do not have a permanence property: In case of feed forward connections when a new minicolumn is created, connections between the input elements and the minicolumn are established for a random subset of currently active elements (1s) in the input. Lateral connections are created from all winner cells of a previous timestep to all winner cells of a current timestep. The number of minicolumns in HTM is fixed, whereas in MCN the number of minicolumns is increased if not enough minicolumns were activated in a current timestep. Full details about learning and activation rules in MCN can be found in [2].

IV. Grid Cell Networks (GCN): An entorhinal cortex cell model to represent space

Besides MCN, the second essential part of the proposed system is a grid cell network that is inspired by entorhinal grid cells.

A. Biological background

In the area of research on biological navigation and mapping, rodents are one of the most popular and best understood subjects. Their hippocampus is one of the most thoroughly investigated mammal brain areas. Rodents do not build detailed geometric maps of their environment, instead they rely on learned connections between integrated self-motion cues and external perception. In the following, a small subset of cell types that are used to represent space in the brain are presented; a more comprehensive presentation of cell types is given by Grieves and Jeffery [31].

In 1948 Tolman [32] showed a kind of latent learning of spatial references in a cognitive map: rats store navigation information, even if they are not relevant for the current tasks. Subsequent researchers identified place cells in the rodent hippocampus, which respond to spatial locations of the rodent [33], and later *head direction cells* [34] which respond to the orientation of the rodent's head. The firing rate of a head direction cell peaks when the rodent's head is facing in a specific direction and decreases for all other directions. They are related to a global reference frame (they are allocentric), thus they are independent from the orientation relative to body but they are related to the orientation in the environment. Head direction cells are influenced by external spatial clues [35]. For example, the rotation of the dominant visual cue in the environment cause approximately the corresponding shift in the head direction cells [36]. However, rats are also able to determine their orientation to some degree in absence of visual cues [37].

For navigation, a rodent has to encode orientation and spatial position. For the latter, besides place cells, rodents use a second type of cells called *grid cells*. Grid cells have place cell like properties but with multiple firing fields arranged in a grid. Grid cells are active when the rodent is at a vertice of a tessellating hexagonal pattern across the environment [38]. This way, a finite number of cells can be used to represent pose in potentially infinitely large space. This encoding is similar to residue number systems in math, which use a finite set of numbers and modulo operation to encode the infinite set of all integers. Similar to head direction cells, the grid cell representation is influenced by external and proprioceptive clues [31]. The activation patterns of groups of grid cells are coordinated and remain coordinated in new environments. They are supposed to form an attractor network that is used for path integration [39].

B. Grid Cell Network (GCN)

Grid cells and head direction cells represent a specific property of space in the brain. Another concept are *conjunctive cells* that do not represent a specific property in space like orientation, place or location. Rather they respond to a combination like a specific spatial location with a particular head direction [40], or there may be grid cells that respond only in combination with a specific head direction [41]. In this work, we build upon the latter cell type aligned in a grid to represent positions together with specific orientations. Fig. 2 shows this (conjunctive) grid cell network (GCN) that encodes the space as (x, y, θ) for 2D position and orientation. Every grid cell has undirected connections or wrap-around connections to all adjacent grid cells so that they span a continuous attractor network (CAN); this recurrent dynamic neural network evolves to stable patterns over time in the absence of an external input [42]. In large environments, when self-motion integration causes the activity peak to move over wrap-around connections, each grid cell corresponds to multiple physical places in the environment. This behavior is similar to the one of grid cells in rodents medial entorhinal cortex. There is biological evidence for CAN-like neural structures for head direction cells and grid cells [39].

In our experiments, we use a grid cell network with $100 \times 100 \times 18$ cells and each cell spreads $0.5m \times 0.5m \times 20°$. The computations in the GCN are based on three functions:

move(x', y', θ') This function is used to accumulate incremental pose changes (like odometry). Input is the relative robot motion (x', y', θ') in the local robot reference frame. Motion can be implemented as activity *shift* in the GCN. Since GCN cells represent space in the *global* reference frame, the shift direction is different in each GCN layer (since each layer represents a different global orientation). To incorporate the input translation (x', y') the current activation of GCN cells within each layer is shifted according to the orientation θ of this GCN layer. In a second step, whole layers are shifted according to the input rotation θ' (they are

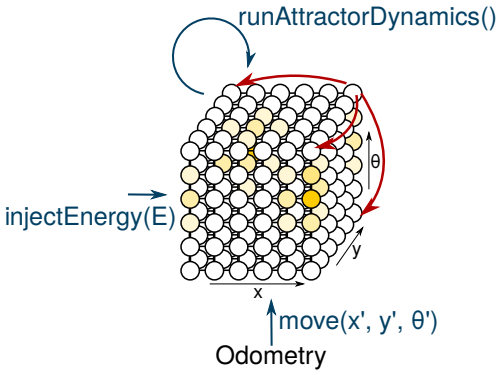


Fig. 2. Grid cell network (GCN) with (conjunctive) grid cells for three dimensions. Wrap-around connections for one grid cell are illustrated in red.

moved "vertically" in the GCN in Fig. 2). Both shifts use the wrap-around structure of the GCN.

injectEnergy(e, m) To promote the activation of a cell of the GCN from the outside, energy can be injected to the network at a particular cell. In our implementation, the injected energy e is added to the activation of the m-th cell.

runAttractorDynamics() The goal of the attractor dynamics is to filter current cell activation and injected energy over time and to form a stable single activation peak (or a small set of stable activation peaks). Similar to the pose cell network from [24], we apply a local excitatory kernel and a larger inhibitory Gaussian kernel. The excitatory kernel has standard deviations of 0.001 cell in each dimensions, inhibition has standard deviations $(7, 7, 2.5)$ cells in (x, y, θ) direction. Additionally, we normalize the total sum of cell activations in the GCN to 1 and set cells with activation smaller than 0.001 to activity zero (followed by an additional normalization to sum 1). Cutting off too small activity values is beneficial for computational efficiency, since we apply the Gaussian kernels only to the sparse set of cells with non-zero activity.

V. Combining MCN with GCN to Exploit Odometry for Place Recognition

How can we combine grid cells as a concept of the entorhinal cortex with MCN that uses neocortex-inspired working principles? The goal of a combination of MCN with grid cells is to involve metric information or odometry data into the MCN to make it potentially suited for more navigation tasks beyond place recognition like localization or mapping.

Fig. 1 shows the architectures of MCN that uses solely images as input (top) and of the minicolumn network that uses images as well as odometry information encoded in the GCN (bottom). The architectural changes for a combination of MCN with GCN can be explained in four steps: (1) We remove the *predictive lateral connections* (PLCs) in MCN (the connections between cells or circles in Fig. 1, top) that maintain a context over a timeseries. (2) Beside a binary encoded image, we provide the binary encoded GCN activation pattern (vertical barcode in Fig. 1, bottom)

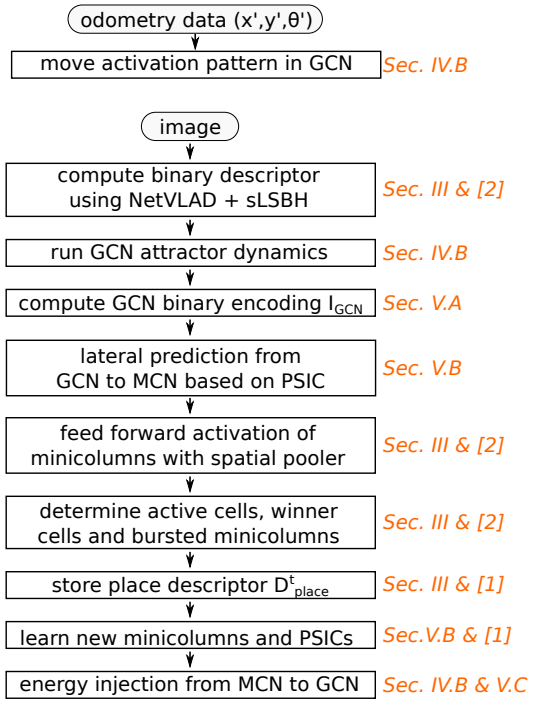


Fig. 3. Overview of the algorithmic steps with references to the details.

to the minicolumn network. The binary encoding of the GCN is explained below in Sec. V-A. (3) We introduce a new type of predictive connections: *predictive sensory input connections* (PSICs) that predict cells in minicolumns in dependence of the binary encoded GCN activation pattern. These connections are directed from the binary encoded GCN activation pattern (vertical barcode in Fig. 1, bottom) to cells in the minicolumn network (circles in Fig. 1, bottom). Their working principles for inference and learning are explained below in Sec. V-B. (4) In case of a loop closure, we want to establish activation peaks in the GCN at the same location as before when the place has been visited for the first time; i.e., we always encode the same pose in the GCN for the same place. Accordingly, our minicolumn network has to be enabled to inject energy back into the GCN. We achieve this with directed connections from cells in minicolumn network (circles in Fig. 1, bottom) to the GCN (circles in Fig. 1, bottom). Details about these connections are given below in Sec. V-C.

Fig. 3 gives a high-level algorithmic overview how the minicolumn network with PSICs interacts with the GCN starting from incoming odometry and image data.

A. Binary encoding of the GCN activation pattern

As predictive connections to cells in the minicolumn network require a binary input vector I_{GCN}, the continuous values of the grid cells in the GCN have to be binary encoded. As vector length $|I_{\text{GCN}}|$ we simply use the number of cells in the GCN, so that the m-th grid cell corresponds to the m-th vector element I_{GCN_m} in I_{GCN}. The $t_{\text{GCN}}\%$ (e.g., 10%) most active cells of all active grid cells in the GCN (activation > 0) are set to 1 whereas all remaining cells are

set to 0. This strategy returns a relatively sparse vector that only encodes the most active regions in the GCN.

B. Prediction through Predictive Sensory Input Connections

The predictive sensory input connections (PSICs) show the same behavior like predictive lateral connections (PLCs) in MCN except that the connections now have a vector element I_{GCN_m} in I_{GCN} as origin rather than another cell in the minicolumn network. So, if a cell $c_{i,j}$, which is the i-th cell in the $j-th$ minicolumn, has any connection to an active vector element I_{GCN_m} $(=1)$ in I_{GCN}, it gets a *predicted* cell $p_{i,j}$:

$$p^t_{i,j} = 1 \Leftrightarrow \exists m : I^t_{\text{GCN}_m} \wedge (I_{\text{GCN}_m}, c_{i,j}) \in \mathbb{P} \tag{1}$$

As described in Sec. III, if then the corresponding minicolumn is activated by the feed forward input (incoming image), the cell gets *active* and *winner*.

New PSICs are created to a winner cell $w_{i,j}$ from all active vector elements I_{GCN_m} $(=1)$ in I^t_{GCN} and added to the set of predictive connections $\mathbb{P}$ iff this winner cell $w_{i,j}$ is part of a bursted minicolumn b_j:

$$\mathbb{P} = \mathbb{P} \cup \{(I_{\text{GCN}_m}, c_{i,j}) : I^t_{\text{GCN}_m} \wedge w^t_{i,j} \wedge b^t_j\} \tag{2}$$

Again, a minicolumn is bursted if it has no predicted cells but is activated by the spatial pooler. Learning solely in case of bursting avoids that cells are connected too densely to the I_{GCN} vector.

C. Energy injection into the GCN

Connections from the cells of the minicolumn network to grid cells of the GCN are required in case of loop closures: A (revisited) place should always come along with the same activation pattern in a specific area of the GCN; this ensures that if the robot moves from a known place to a adjacent known place (and the GCN activation pattern is shifted accordingly), the right cells in the minicolumn network and the right place are predicted.

These connections for injection of an energy E_m to the m-th grid cell are basically identical to the existing PSICs in a minicolumn network except that they are directed from the minicolumn cells directly into the GCN. As the number of elements in I_{GCN} equals the number of grid cells in the GCN, a mapping from the I_{GCN}-elements to the GCN-cells is trivial. The energy E_m from the minicolumn network for the m-th grid cell in the GCN can be computed by:

$$E_m = s \cdot \sum_{i,j} a_{i,j} \wedge (I_{\text{GCN}_m}, c_{i,j}) \in \mathbb{P} \tag{3}$$

s is a scalar value > 0 that scales the energy from the active cells $a_{i,j}$. We use $s = 1e^{-4}$ in our experiments below. To avoid a self-maintenance of a current place, a newly learned connection from the minicolumn network to the GCN has to *cool down* for t_{cooldown} timesteps (e.g., $t_{\text{cooldown}} = 5$), so that it is first used after a couple of timesteps. This prevents the minicolumn network from learning from its own injected energy.

VI. PROOF-OF-CONCEPT EXPERIMENTS

In the following, two proof-of-concept experiments are used to demonstrate the feasibility of combining MCN with grid cells and to show that this system actually benefits from additional odometry data. We compare the place recognition performance of the two minicolumn networks with predictive lateral connections (PLCs) and predictive sensory input connections (PSICs), and show corresponding loop closure detections.

Experimental setup

A remote-controlled skid-steering mobile robot (Fig. 4, top) was used to collect two different datasets. The robot is equipped with a fisheye camera with $250°$ aperture angle to acquire panoramic images as well as with a 2D LiDAR for ground truth. ROS (Robot Operating System) was used for drivers, communication, and data recording. To create quite accurate ground truth data, we first mapped both environments with a LiDAR-based SLAM system. Subsequently, both maps were used during data collection to perform LiDAR-based localization for ground-truth. Note that LiDAR data and maps are solely used for ground truth and visualizations. To ensure a realistic temporal and spatial distance between consecutive images, images were collected while the full data processing pipeline for loop closure detection with MCN was running; odometry data was recorded with 10Hz and images with approx. 0.5Hz. We used NetVLAD [10] and sLSBH [2] to encode and binarize images for the minicolumns' feed-forward input. A description of the minicolumn network's parameters is given in [2]; they were set to: $k_{max} = 100$, $k_{min} = 50$, #connections per minicolum $= 800$, #cells per minicolumn $= 32$, $\theta = 0.55$.

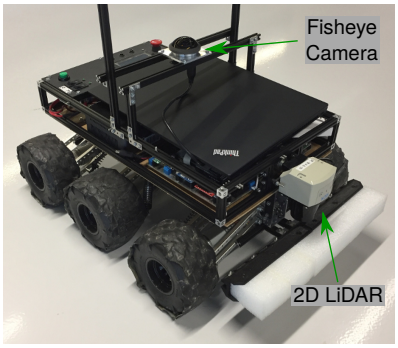
We recorded data in two environments: (1) The first environment is the foyer of a lecture hall building with passing students as depicted in Fig. 4 (middle). A distance of around 730m was covered while 561 images were collected. (2) The second environment is an approx. 160m long corridor with a very challenging repetitive structure, two almost identical elevator areas, and a few passing people (Fig. 4, bottom). The robot recorded 878 images along an approx. 860m trajectory by driving from one end to the other multiple times. Maps and scales are shown for the lecture hall building and the corridor in Fig. 6 and Fig. 7, respectively.

Place Recognition Performance

Both minicolumn networks with PLCs and PSICs return a binary vector of active winner cells for each timestep which serves as *place descriptor* D_{place} for the current place. After collecting all place descriptors, the descriptors D^i_{place} are compared pairwise with an overlap score

$$\text{overlap score}(D^{(1)}_{\text{place}}, D^{(2)}_{\text{place}}) = 2 \cdot \frac{< D^{(1)}_{\text{place}}, D^{(2)}_{\text{place}} >}{\sum D^{(1)}_{\text{place}} + \sum D^{(2)}_{\text{place}}}$$

to receive a similarity matrix S. S is finally used to compute the precision recall curve. Places with a ground truth distance ≤ 5m are considered as the same place in all experiments.



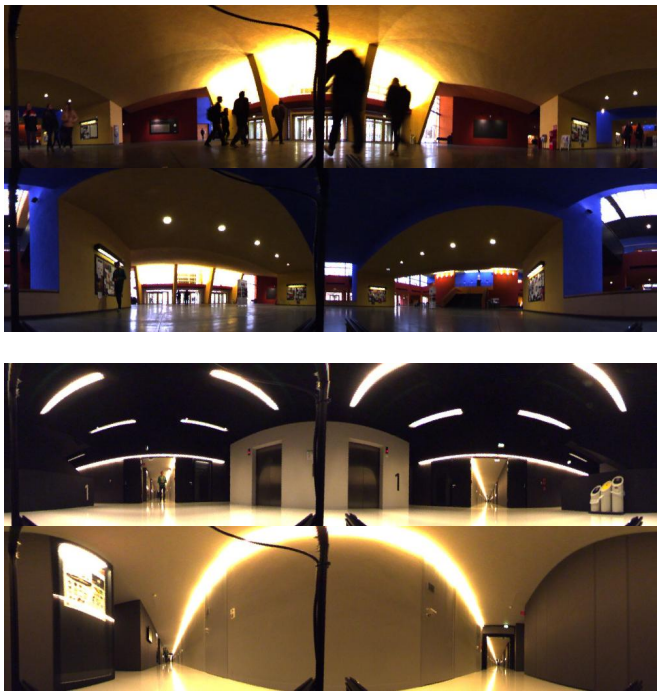


Fig. 4. Top: The robot used to collect data. Middle: Example images from the first test environment, a foyer of a lecture hall building with passing students. Bottom: Example images from the second test environment, a corridor with repetitive structure. The upper image shows the area in front of elevators which is partly confused in the experiments by the algorithms.

Fig. 5 shows the resulting place recognition curves for both predictive connection types in the two environments. The curves indicate that the described replacement of predictive lateral connections (PLCs) in MCN with predictive sensor input connections (PSICs) based on odometry can work in principle: the system is able to provide reasonable visual place recognition results and can benefit from the additional odometry information.

Fig. 6 and Fig. 7 show corresponding loop closure detections when we apply a threshold t to the similarity matrices S of both environments and connection types. Matchings are only allowed for images with a distance of at least 20 timesteps to avoid matchings between consecutive images. We used $t = 0.42$ which was determined from the maximum F1 score of an earlier recording of the corridor. For the foyer of the lecture hall building the minicolumn network with PLCs achieved 892 true and 37 false matchings whereas the

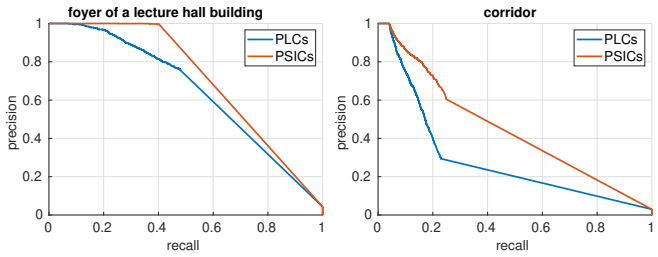


Fig. 5. Precision recall curves for the place recognition performance of the minicolumn network with PLCs and PSICs for both environments.

minicolumn network with odometry information and PSICs achieved 642 true and 0 false matchings. There are no false matchings with PSICs but the number of true matchings was reduced, too. To achieve a higher number of true matchings, we can use a less restrictive threshold, e.g. with $t = 1e^{-10}$ the MCN with PSICs achieves 1743 true matchings and still only 8 false matchings. For the challenging corridor dataset the minicolumn network with PLCs achieved 640 true and 277 false matchings and the minicolumn network with PSICs achieved 642 true and 188 false matchings. Again, the number of false matchings was reduced by using PSICs.

VII. DISCUSSION

In the previous section, we were able to show that a minicolumn network actually benefits from replacing predictive lateral connections (PLCs) with predictive sensory input connections (PSICs). PSICs predict next places in dependence of the motion of the robot encoded in the GCN whereas the PLCs forecast places through their temporal order. Regarding their working principles, presumably, it is beneficial to combine both prediction types to exploit spatial and temporal information jointly.

Hawkins et al. [27], [29] and Lewis et al. [30] presented theoretical ideas about a combination of their HTM-theory with locality information or grid cells. With our experimental results on real-world datasets, we could give potential evidence that their theoretical ideas can be transfered to practical applications.

The presented proof-of-concept experiments were intended to show the feasibility of a combination of a neocortex-inspired model with entorhinal grid cells. The parameters of the minicolumn network were not tuned for both the indoor environments and the new prediction connection types, and simply copied from our experiments in [2] with MCN with PLCs and real-world datasets. A next step is to investigate the influence of the extensions to the parameters and to perform a subsequent comparison to related algorithms. Since RatSLAM [43] is an algorithm that also uses a grid cell network, a comparison to the minicolumn network with the GCN is particularly interesting.

For further extensions of our minicolumn network, we want to consider if we can adopt concepts from HTM like permanences and hebbian learning, segments, or a fixed number of minicolumns.

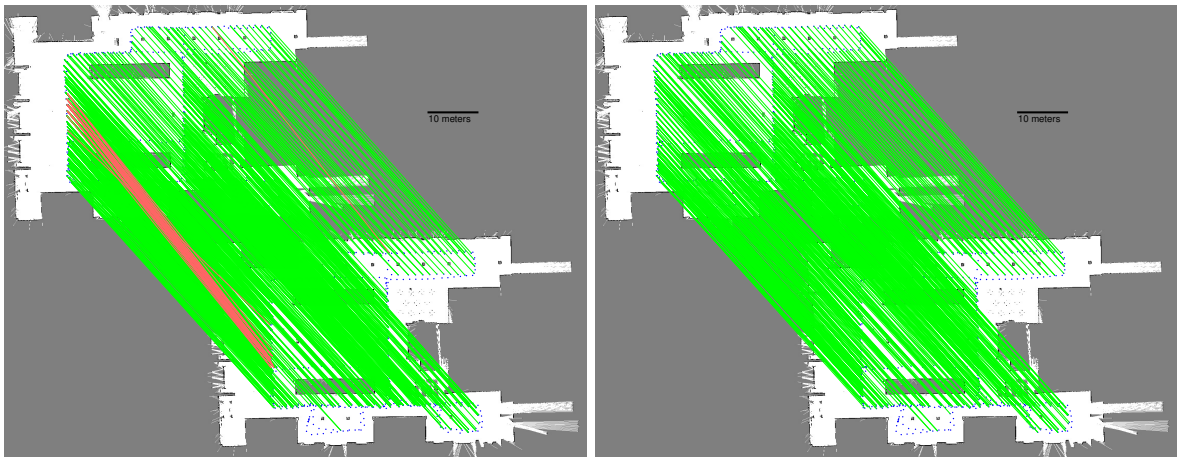


Fig. 6. Results of the loop closure detection in the lecture hall building with PLCs (left) and PSICs (right). Green lines show true positive matchings, red lines show false positive matchings. A matching is true if both positions have distance $\leq$ 5m.

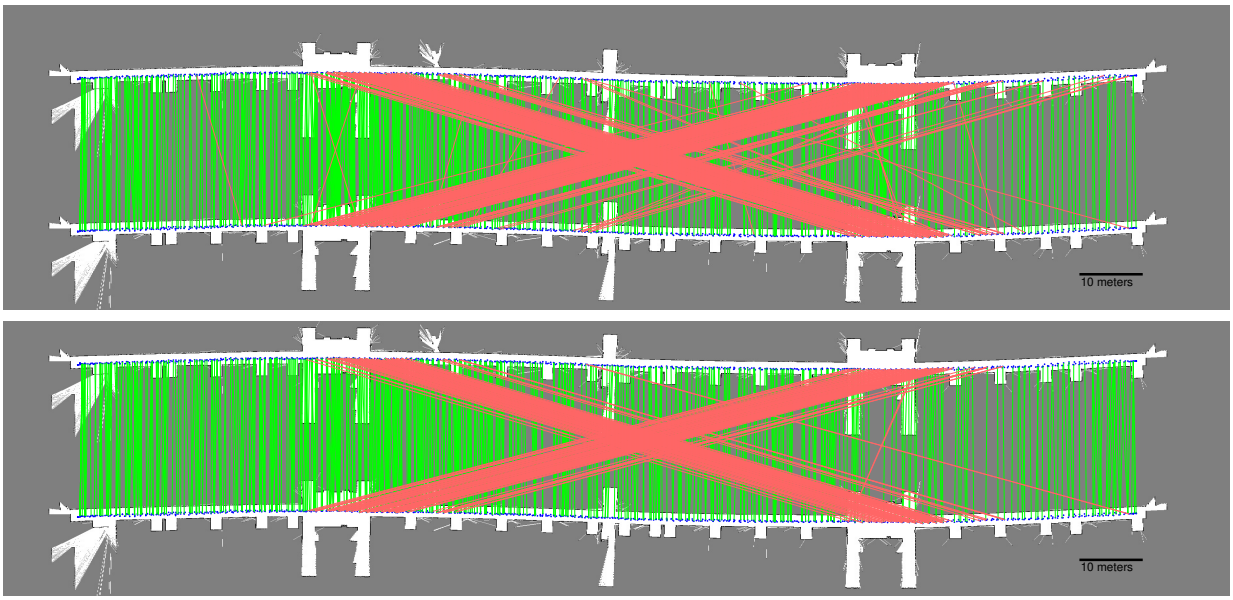


Fig. 7. Results of the loop closure detection on the corridor with PLCs (top) and PSICs (bottom). Green lines show true positive matchings, red lines show false positive matchings. A matching is true if both positions have distance $\leq$ 5m.

In this work, our system uses images from one camera as well as odometry data, however, as Hawkins et al. proposed in their work [27] additional sensors (or even sensor modalities) could be used by an introduction of *long-range lateral connections* and *feedback connections*. We are going to consider this idea in order to potentially use additional sensor data like more cameras or different modalities like 2D LiDAR data.

Our overall objective in the paper and beyond is to push MCN as a neocortex-inspired model into the direction of navigation. Starting from MCN which performs visual place recognition we now can exploit odometry data as metric information. A next step would be to build a SLAM system that builds and optimizes a pose graph to integrate an explicit metric representation of the world into our system, potentially building upon ideas from experience mapping in RatSLAM [43], or ideas from Pose SLAM [44].

VIII. CONCLUSION

In this work, we intended to answer the question *if it is possible to combine MCN that is inspired by the human neocortex with entorhinal grid cells* to exploit odometry data for visual place recognition. We gave a short overview of MCN in Sec. III and introduced the grid cell network (GCN) in Sec. IV that accumulates odometry data and manages location hypotheses. Subsequently, we proposed in Sec. V how MCN can be combined with the GCN by replacing the predictive lateral connections of the original MCN with predictive sensory input connections between the minicolumn network and the GCN. In two proof-of-concept experiments in Sec. VI, we were able to show that the minicolumn network with predictive connections from the grid cell network works in principle and, as expected, outperforms the minicolumn network with lateral connections since it can

additionally exploit odometry data. In conclusion, we could show that *it is possible to combine this neocortex-inspired model with entorhinal grid cells*, and that the system actually exploits the additional odometry data. Sec. VII discussed potential next steps like architectural extensions and further experiments.

REFERENCES

[1] P. Neubert, S. Ahmad, and P. Protzel, "A sequence-based neuronal model for mobile robot localization," in *Proc. of KI: Advances in Artificial Intelligence*, 2018.

[2] P. Neubert, S. Schubert, and P. Protzel, "A neurologically inspired sequence processing model for mobile robot place recognition," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, 2019.

[3] S. Lowry, N. Sunderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *Trans. Rob.*, vol. 32, no. 1, 2016.

[4] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. of Computer Vision*, vol. 60, 2004.

[5] P. Neubert and P. Protzel, "Beyond holistic descriptors, keypoints, and fixed patches: Multiscale superpixel grids for place recognition in changing environments," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, 2016.

[6] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. of Computer Vision*, vol. 42, no. 3, 2001.

[7] N. Sünderhauf and P. Protzel, "BRIEF-Gist - closing the loop by simple means," in *Proc. of Int. Conf. on Intelligent Robots and Systems (IROS)*, 2011.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.

[9] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of ConvNet features for place recognition," *Proc. of Int. Conf. on Intelligent Robots and Systems (IROS)*, 2015.

[10] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," *Trans. on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, 2018.

[11] M. Cummins and P. Newman, "FAB-MAP: Probabilistic localization and mapping in the space of appearance," *The Int. J. of Robotics Research*, vol. 27, no. 6, 2008.

[12] M. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights." in *Proc. of Int. Conf. on Robotics and Automation (ICRA)*, 2012.

[13] E. Johns and G. Yang, "Dynamic scene models for incremental, long-term, appearance-based localisation," in *Proc. of Int. Conf. on Robotics and Automation (ICRA)*, 2013.

[14] S. Lynen, M. Bosse, P. Furgale, and R. Siegwart, "Placeless place-recognition," in *Proc. of Int. Conf. on 3D Vision*, 2014.

[15] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Towards life-long visual localization using an efficient matching of binary sequences from images," in *Proc. of Int. Conf. on Robotics and Automation*, 2015.

[16] W. Maddern, M. Milford, and G. Wyeth, "CAT-SLAM: Probabilistic localisation and mapping using a continuous appearance-based trajectory," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 429–451, 2012.

[17] E. Pepperell, P. I. Corke, and M. J. Milford, "All-environment visual place recognition with SMART," in *Proc. of Int. Conf. on Robotics and Automation (ICRA)*, May 2014, pp. 1612–1618.

[18] T. Madl, K. Chen, D. Montaldi, and R. Trappl, "Computational cognitive models of spatial memory in navigation space: A review," *Neural Networks*, vol. 65, pp. 18 – 43, 2015.

[19] T. Oess, J. L. Krichmar, and F. Rhrbein, "A computational model for spatial navigation based on reference frames in the hippocampus, retrosplenial cortex, and posterior parietal cortex," *Frontiers in Neurorobotics*, vol. 11, p. 4, 2017.

[20] R. Möller, "Local visual homing by warping of two-dimensional images," *Robotics and Autonomous Systems*, vol. 57, no. 1, pp. 87 – 101, 2009.

[21] P. Neubert, S. Schubert, and P. Protzel, "Sampling-based methods for visual navigation in 3d maps by synthesizing depth images," in *Proc. of Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[22] M. Milford, G. Wyeth, and D. Prasser, "RatSLAM: a hippocampal model for simultaneous localization and mapping," in *Proc. of Int. Conf. on Robotics and Automation (ICRA)*, 2004.

[23] A. Banino, C. Barry, B. Uria, C. Blundell, T. Lillicrap, P. Mirowski, A. Pritzel, M. J. Chadwick, T. Degris, J. Modayil, G. Wayne, H. Soyer, F. Viola, B. Zhang, R. Goroshin, N. Rabinowitz, R. Pascanu, C. Beattie, S. Petersen, A. Sadik, S. Gaffney, H. King, K. Kavukcuoglu, D. Hassabis, R. Hadsell, and D. Kumaran, "Vector-based navigation using grid-like representations in artificial agents," *Nature*, vol. 557, no. 7705, pp. 429–433, 2018.

[24] M. Milford and G. Wyeth, "Mapping a suburb with a single camera using a biologically inspired SLAM system," *IEEE Transactions on Robotics*, 2008.

[25] J. Hawkins, S. Ahmad, S. Purdy, and A. Lavin, "Biological and machine intelligence (BAMI)," 2016, initial online release 0.4. [Online]. Available: https://numenta.com/resources/biological-and-machine-intelligence/

[26] J. Hawkins, *On Intelligence (with Sandra Blakeslee)*. Times Books, 2004.

[27] J. Hawkins, S. Ahmad, and Y. Cui, "A theory of how columns in the neocortex enable learning the structure of the world," *Frontiers in Neural Circuits*, vol. 11, p. 81, 2017.

[28] "Nupic," https://github.com/numenta/nupic, accessed: 2018-05-09.

[29] J. Hawkins, M. Lewis, S. Purdy, M. Klukas, and S. Ahmad, "A framework for intelligence and cortical function based on grid cells in the neocortex," *Frontiers in Neural Circuits*, vol. 12, 2019.

[30] M. Lewis, S. Purdy, S. Ahmad, and J. Hawkins, "Locations in the neocortex: A theory of sensorimotor object recognition using cortical grid cells," *Frontiers in Neural Circuits*, vol. 13, p. 22, 2019.

[31] R. M. Grieves and K. J. Jeffery, "The representation of space in the brain," *Behavioural Processes*, vol. 135, pp. 113 – 131, 2017.

[32] E. C. Tolman, "Cognitive maps in rats and men." *The Psychological Review*, vol. 55, no. 4, p. 189, 1948.

[33] J. OKeefe and J. Dostrovsky, "The hippocampus as a spatial map: preliminary evidence from unit activity in the freely moving rat," *Brain Research 34*, pp. 171-175, 1971.

[34] J. Ranck, "Head direction cells in the deep cell layer of dorsal presubiculum in freely moving rats," *Society Neuroscience Abstracts 10*, p. 599, 1984.

[35] J. S. Taube *et al.*, "Head direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis," *Journal of Neuroscience 10 (2)*, pp. 420-435, 1990.

[36] J. S. Taube, "Head direction cells recorded in the anterior thalamic nuclei of freely moving rats," *Journal of Neuroscience 15 (1)*, pp. 70-86, 1995.

[37] S. J. Mizumori and J. D. Williams, "Directionally selective mnemonic properties of neurons in the lateral dorsal nucleus of the thalamus of rats," *Journal of Neuroscience 13*, pp. 4015-4028, 1993.

[38] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, and E. I. Moser, "Microstructure of a spatial map in the entorhinal cortex," *Nature*, vol. 436, pp. 801–806, Aug. 2005.

[39] B. L. McNaughton *et al.*, "Path integration and the neural basis of the cognitive map," *Nature Reviews Neuroscience 7*, pp. 663-678., 2006.

[40] F. Cacucci, C. Lever, T. J. Wills, N. Burgess, and J. O'Keefe, "Theta-modulated place-by-direction cells in the hippocampal formation in the rat," *Journal of Neuroscience*, vol. 24, no. 38, pp. 8265–8277, 2004.

[41] F. Sargolini, M. Fyhn, T. Hafting, B. L. McNaughton, M. P. Witter, M.-B. Moser, and E. I. Moser, "Conjunctive representation of position, direction, and velocity in entorhinal cortex," *Science*, vol. 312, no. 5774, pp. 758–762, 2006.

[42] D. J. Amit, *Modelling Brain Function: The World of Attractor Neural Networks*, 1st ed. Cambridge University Press, 1992.

[43] M. Milford, D. P. Prasser, and G. Wyeth, "Experience Mapping: Producing spatially continuous environment representations using RatSLAM," *Proc. of Australasian Conf. on Robotics and Automation (ACRA)*, 2005.

[44] R. Valencia and J. Andrade Cetto, *Mapping, Planning and Exploration with Pose SLAM*, 01 2018, vol. 119.