

Biologically Inspired Visual Landmark Processing for Simultaneous Localization and Mapping

D. P. Prasser, G. F. Wyeth, M. J. Milford

School of Information Technology and Electrical Engineering
University of Queensland
Brisbane, Australia
{prasserd, wyeth, milford}@itee.uq.edu.au

Abstract—This paper illustrates a method for finding useful visual landmarks for performing Simultaneous Localization and Mapping (SLAM). The method is based loosely on biological principles, using layers of filtering and pooling to create learned templates that correspond to different views of the environment. Rather than using a set of landmarks and reporting range and bearing to the landmark, this system maps views to poses. The challenge is to produce a system that produces the same view for small changes in robot pose, but provides different views for larger changes in pose. The method has been developed to interface with the RatSLAM system, a biologically inspired method of SLAM. The paper describes the method of learning and recalling visual landmarks in detail, and shows the performance of the visual system in real robot tests.

Keywords - SLAM; visual landmarks; mobile robot; visual cortex; robot vision

I. INTRODUCTION

RatSLAM is an approach to the problem of Simultaneous Localization and Mapping (SLAM) using techniques based on computational models of the rodent hippocampus. The RatSLAM system has previously been demonstrated performing SLAM using a vision system that detected artificial landmarks [1, 2]. This paper describes a new addition to the system that allows operation in an unmodified environment. This is achieved by replacing the artificial landmark system with a more general vision module. The replacement system is loosely based on the mammalian visual cortex and is designed as the input representation for the rest of the RatSLAM system.

The new vision system uses an appearance based method that learns to recognize particular places. The output of the robot's forward looking camera is associated with its believed position as it maps the environment. This avoids the difficulty of constructing a structural map of the environment and finding corresponding features between the structural map and the camera image.

Section II of the paper describes some approaches to the problem of localization and mapping using computer vision. This section also provides a brief outline of the RatSLAM system. Section III provides the implementation details of the new visual system and Section IV describes the experimental setup used for evaluating the vision

system. The performance of the new system is shown in Section V along with a discussion of the systems strengths and limitations. Finally the conclusions are detailed in Section VI.

II. BACKGROUND

A. Vision Based SLAM

There is a significant body of work on localization and SLAM systems that use vision. These works can be divided into two groups: those that map two or three dimensional information about the environment such as landmark location [3, 4]; and those that attempt to recover position directly from the appearance of an image [5-8]. The first group is inappropriate for the RatSLAM system as RatSLAM does not maintain a global Cartesian estimate of its pose which makes it extremely difficult to determine the global position of features in the environment. Additionally, it has been shown that RatSLAM can already function using appearance based methods [1, 2].

Appearance based SLAM can be thought of as a learning process. The first time the robot experiences a particular visual scene it records the appearance of the scene and its position estimate. Later when exposed to same visual scene it can use its recorded position estimate to re-localize. One of the complications of appearance based SLAM is the fact that learning entire images would be computationally difficult. Also unprocessed images are sensitive to small changes in camera position which are below the spatial resolution at which RatSLAM operates. To solve this problem the camera images must be represented in some way that is easier to learn. Representations that have been used for appearance based localization include Principal Component Analysis [5, 6] and histograms [7, 8]. Principal Components cannot be computed until after the environment has been learnt which makes it appear unsuitable for incremental SLAM. Histogram approaches have mostly been used with panoramic cameras and with the intention of developing highly topological orientation invariant representations of places. The concern is that histograms may provide too much invariance for the level of localization RatSLAM needs.

An alternative form of image representation is based on the complex cells of the visual cortex. Complex cells have

been previously demonstrated as a technique for performing place recognition [9]. Complex cells are generally considered to detect or respond to edges or bars at a particular orientation within a region of the retina, which is known as the cell's receptive field [10]. An example of the output of these cells is shown in Fig. 1. As complex cells are insensitive to changes in feature position within their receptive field, they are suitable for spatially sub-sampling an image and also for generalization. For example, if a complex cell has a receptive field of $6^\circ \times 6^\circ$ then panning or tilting the camera by 3° will not change the cell's activity significantly. This spatial invariance in the image domain translates to local invariance in the robot's position, allowing the robot to generalize in x, y, θ .

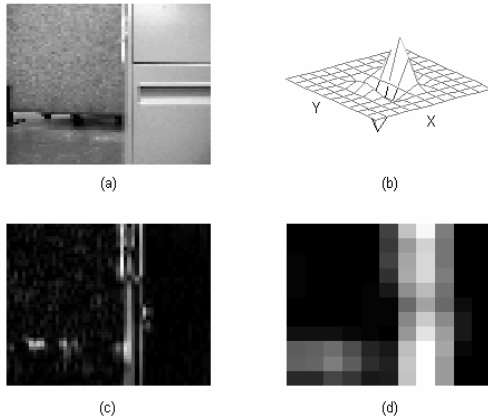


Figure 1. The complex cell model applied to a test image. (a) Original image. (b) Gabor filter tuned to detect changes along the X axis. (c) The absolute output of the Gabor filter. (d) Complex cell outputs. The implementation in this paper has another set of cells tuned to detect horizontal edges.

B. RatSLAM Architecture

Fig. 2 shows the basic RatSLAM model. The robot's pose is represented by the activity in a competitive attractor network called the *pose cells*. The pose cells are arranged as a three dimensional array with two dimensions corresponding to x and y position, and the third dimension to orientation. Wheel encoder information is used to perform path integration by injecting activity into the pose cells thereby shifting the current activity packets. Vision information is converted into a local view representation which if familiar, injects activity into the particular pose cells that are associated with that specific local view. This paper focuses on the vision pathway and to a small extent the link between the local view and the pose cells.

The local view module is a collection of neural units that represent what the robot perceives to the rest of the RatSLAM system. Significantly the local view contains no explicit spatial information such as distance and bearing to a landmark, instead RatSLAM learns to associate visual appearance with different poses. Our approach has two levels: a biologically motivated feature extraction of the image using the complex cells described earlier; followed by a primitive scene recognition stage. The output of the system is a group of neural units, each of which responds

to a different visual scene. RatSLAM associates local view activity with pose cell activity using Hebbian learning (Fig. 3).

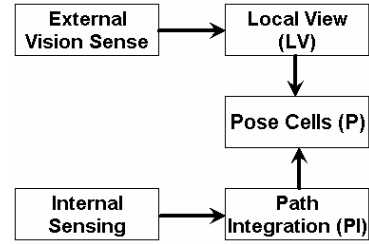


Figure 2. Pose is represented by activity in the pose cells. This pose is updated continually by path integration and local view activity input.

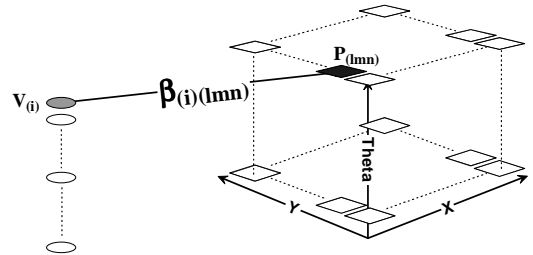


Figure 3. Illustration of the local view network and pose cell network. Activated units in the local view, V_i , become associated with activated units in the pose cells, P , through learnt weighted connections between the two networks.

III. IMPLEMENTATION

The units in the local view are controlled by a two stage computer vision system. The first stage uses a complex cell model to extract features from the image. These features are then used to represent the image. The second stage uses a sum of absolute differences metric to compare the output of the complex cells against previously learnt templates. Each template has a corresponding local view unit which is activated when the input matches the template.

A. Feature Extraction

The complex cell model used in this work is based on the first layer of [11]. In this model the input image is first normalized and then convolved with a number of odd Gabor filters to produce edge detected images q^l , with l denoting which filter produced the image. Each of these is then passed through a winner-takes-most mechanism across the orientation dimension:

$$r^l(x, y) = \begin{cases} 0, & \frac{|q^l(x, y)|}{M} < \gamma \\ \frac{|q^l(x, y)| - M\gamma}{1 - \gamma}, & \frac{|q^l(x, y)|}{M} \geq \gamma \end{cases} \quad (1)$$

Where $M = \max_k |q^k(x, y)|$ is the maximum absolute response at a particular pixel location for any of the edge orientations and is γ a competition parameter. These outputs, r^l , are similar to the 'simple cells' in the visual

cortex. The filter outputs are pooled with a local summation to produce the complex cell outputs:

$$c^l(i, j) = \tanh \sum_{x, y} p(x, y) H(r^l(x, y) - \theta) \quad (2)$$

Where H is a unit step function and $p(x, y)$ is a two dimensional Gaussian distribution which defines the spatial extent of the complex cell's receptive field. The centre of the distribution x_i', y_j' , specifies the centre of the receptive field while the σ parameter of the Gaussian controls the receptive field's size. The sensitivity of the complex cell is controlled by the θ parameter. The hyperbolic tangent is used to limit the result. In this work σ is constant and x_i', y_j' vary to create a square grid of evenly sized and spaced complex cells on the image surface.

In the current implementation only two orientations are used for the complex cell filters – horizontal and vertical. This reduction of features is motivated by the nature of the test environment. As the test environment is an indoor office space there is an abundance of both vertical and horizontal lines formed by doors and windows. These features are also less likely to be confused by dynamic objects such as people. The horizontal Gabor filter is defined as follows:

$$G(x, y) = \sin(\omega x) \exp\left(\frac{\omega^2}{2} \left(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2}\right)\right) \quad (3)$$

while the vertical filter has its sinusoidal component along the y axis.

B. Template Matching

The output of the complex cell filters is input for a sum of absolute differences (SAD) module. The SAD module operates as a learning system that compares the input against a previously learned set of templates. When the minimum distance between any of the learnt set and the input is larger than a threshold d_{max} the input is added to the set as a new template.

The output of the SAD module is a linear vector of cells each of which corresponds to one particular template. The activation, a_i , of each cell is inversely proportional to the distance, d_i , between the cell's template and the input, provided that the distance is not greater than d_{max} . A non-linearity, ϵ , is used to prevent problems when the distance is zero.

$$a_i = \begin{cases} 0, & d_i > d_{max} \\ 1/(d_i + \epsilon), & d_i \leq d_{max} \end{cases} \quad (4)$$

Finally the cell activation is normalized to unity over all of the LV cells. By using this representation the SAD module can respond to uncertain matches by weakly activating two or more cells rather than just signaling the most correct template. The performance of the SAD

module is chiefly controlled by the d_{max} parameter. Large values for this parameter will result in a small number of more ambiguous templates being found, whereas small values will produce many highly specific templates.

At several points in the environment the complex cell outputs can be all zero or very near zero. This causes a template to form which represents seeing no features. The frequency with which this template is found means that this template contributes no useful information to the RatSLAM system and does not help in localization. To prevent this, there is a special template that corresponds to the case of no visual input. This template has no corresponding LV unit and is never linked to any pose cells.

The vision system implements a crude form of expectation within the template matching process. In an ideal situation the order in which templates are added to the set of learned templates will correspond roughly to the relative physical position of the templates. The system uses the best template match from the previous frame, template b , as the input to an expectation system which suppresses templates that are not nearby in template index. This means the pose cell system now needs two frames before it can start to recover from global kidnapping.

$$v_i = \begin{cases} a_i, & |i - b| < \delta_i \\ 0, & |i - b| \geq \delta_i \end{cases} \quad (5)$$

IV. EXPERIMENTAL SETUP

The RatSLAM system was tested on a Pioneer 2DXE mobile robot. The complex cell processing was performed on the robot's onboard 400 MHz computer and the results were wirelessly transmitted to a 1.1 GHz laptop where the template matcher and the rest of the RatSLAM system operate. The local view units were updated at 1 Hz. The robot is equipped with a forward looking camera that has a 50° by 40° field of view.

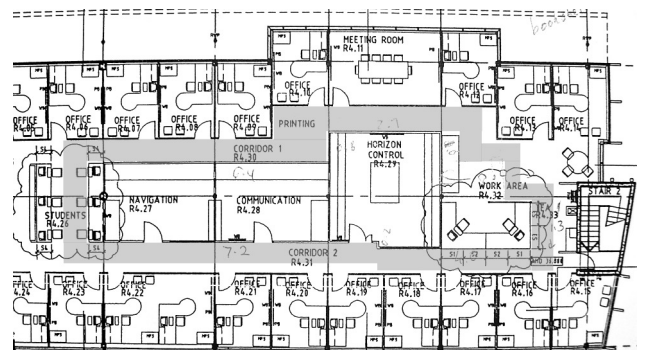


Figure 4. Floor plan of test environment. The shaded area indicates the approximate path of the robot.

The test environment was a loop of corridor in a building at our workplace (see Fig. 4). The robot used a sonar wall following behavior to navigate around the corridor. For these experiments the environment was

unmodified. The length of the loop was calculated to be approximately 70 meters long.

There were a number of parameters in the new local view system that required tuning to make RatSLAM function either successfully or at all. The values used for these parameters are shown in Table 1. Using these parameters the output of the complex cell feature extraction stage is 108 units divided into two layers, each 6 units high by 9 units wide. The network was configured to look for coarse, large scale features in the environment; consequently the fine detail in the input image can be ignored by deliberately using a low image resolution.

TABLE I. VISION SYSTEM PARAMETERS

Parameter	Value
Input image resolution	64×48 pixels
Gabor filter size	11×11 pixels
Gabor filter frequency	2 rad/pixel
Gabor filter σ	2
Complex cell θ	0.3
Complex cell γ	0.7
Complex cell σ	3
Complex cell spacing	6 pixels
Template matcher d_{max}	20
Template matcher temporal suppression δ_t	4

V. RESULTS

A. Performance of Template Matching

The appearance based reasoning used by the system is incapable of explicitly extrapolating its environment to new robot locations. Instead, this generalization must be implicitly performed by the Complex Cell layer and the template matching process. However the cost of increasing generalization is increased ambiguity when localizing as the template will now respond to more varied visual inputs. The balance found in these experiments was to keep the template matcher fairly selective with its threshold, d_{max} , set to 20. The result of this is that almost every second input frame is learnt as a new template during the first lap.

The effectiveness of the entire LV system can loosely be described by a graph such as Fig. 5. This plot shows which templates were detected and consequently which LV units were active at any given time. For the experiment which generated Fig. 5 the robot completed three traversals of the environment. In the first lap the robot is mostly learning its environment so the activated template is usually the most recently learnt template. On the second and third laps many of the templates seen on the first lap are recognized again, in the same order. It is these repeated templates that RatSLAM uses to localize. It can also be seen that the total number of learnt landmarks is increasing during the second

and third laps. This shows that the robot did not experience all possible variations of visual input during the first traversal. Increasing the d_{max} parameter in the template matching module would reduce the number of new templates per lap, but at the cost of increasing ambiguity.

There appears to be an amount of random noise in the template matcher's output, closer examination reveals that some of this 'noise' is in fact repeated between laps. This is merely the result of visual ambiguity in the environment – some places look like other places to the template matcher. Some templates are found with great frequency in the environment, for example template number 9 in Fig. 5. This template, shown in Fig. 6, occurs so frequently as to be almost useless for localization. By searching through a library of images of the test environment an input image that matches this template can be found, shown in Fig. 7. It is apparent that template 9 describes a blank wall with a distinct horizontal line between the wall and the floor. Since the majority of the walls in the environment are blank it is not surprising to find that one template will match so many inputs.

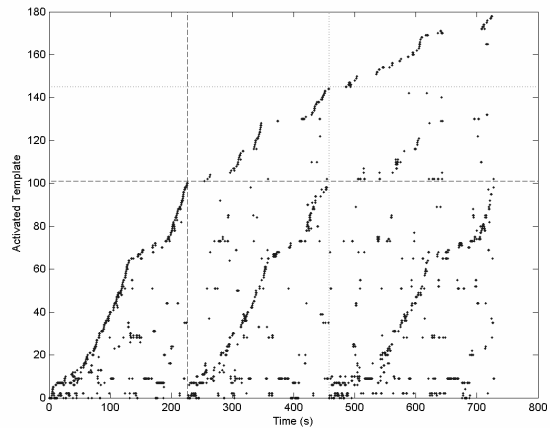


Figure 5. Activated LV units versus time as the robot traverses the environment. For any given input zero or more templates may be found. In this example the robot completed three laps of the environment, the start of the second and third laps are shown by the vertical lines. The number of learnt templates at the end of the first and second laps are shown by the horizontal dashed lines.



Figure 6. The most frequently found template from the experiment shown in Fig. 5. The horizontal complex cell activity levels are shown on the left and the vertical complex cells are shown on the right. In both cases black corresponds to no activity while white indicates the maximum possible level of activation

A better behaved template such as the one shown in Fig. 8 may only match to a few locations in the environment. A close, but imperfect, match to the template in Fig. 8 is shown in Fig. 9.

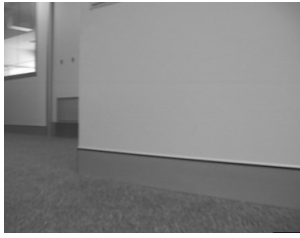


Figure 7. An image which was found to match closely with the template shown in Fig. 6 in an offline experiment. The dark near horizontal line in the foreground results in large amount of activity in the horizontal complex cells associated with the lower portion of the image (Fig. 6, left). The amount of ambiguity present in pictures that contain a large amount of blank wall explains the frequency with which this template is found.

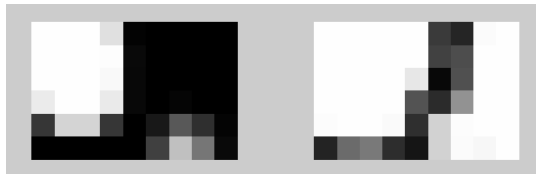


Figure 8. A template which is less spatially ambiguous.

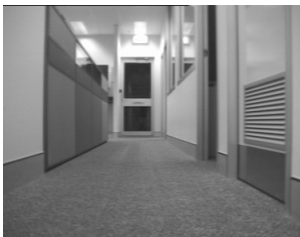


Figure 9. An image which produces complex cell outputs similar to the template in Fig. 8.

B. Relationship between Templates and Robot Pose

The spatial position to which each template corresponds can be recovered by examining the pose cells that are connected to the template. In the RatSLAM system pose cells are arranged as a three dimensional grid with the vertical axis representing orientation. Fig. 10 shows the pose cells associated with a reasonably discriminatory template. The orientation axis of the pose cell's wraps around at 360° so the two blobs on the left of Fig. 10 are actually part of the one position cloud. What this figure shows is that this LV unit is activated when the robot is in either of two places in the environment. One of these places is tightly defined, while the other is a longer region. It would appear that this area was learnt while traveling down a corridor with relatively unchanging visual input. It becomes clear then that it may not always be possible to completely localize from one frame of visual input, because one LV unit may signal more than one location. RatSLAM deals with this ambiguity by reinforcing position hypotheses over successive frames.

Not all templates are as spatially discriminating however. The template in Fig. 6 is linked to many places in the environment, shown in Fig 11. Recognizing this template does not help much when re-estimating position as the template is linked to almost half the pose cells the

robot has ever visited. To account for this RatSLAM weights the influence of each LV cell by the inverse of the number of pose cells it is associated with.

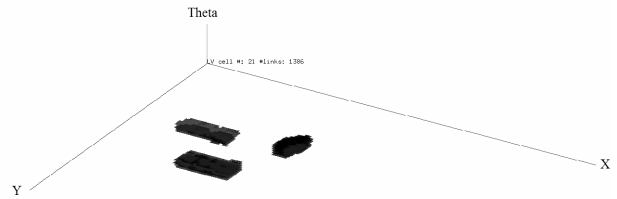


Figure 10. Pose cells associated with one particular template. Since the orientation (θ) dimension wraps around at 360° the activated pose cells on the left are all connected together.

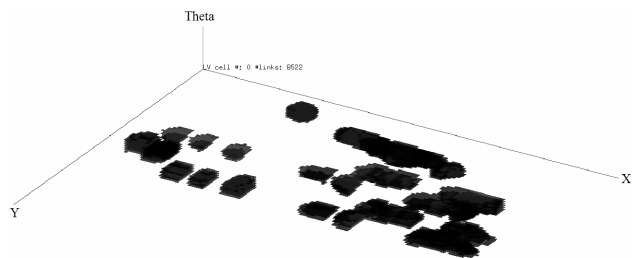


Figure 11. Locations that matched with the most frequently found template shown in Fig. 6.

C. SLAM Performance

The viability of the new visual system can be assessed by examining the performance of RatSLAM using the new system. The representations RatSLAM creates of its environment are not Cartesian although local areas may have strong Cartesian properties. During mapping the system builds a consistent topological representation of the environment. Consequently conventional performance metrics analyzing the closeness of a geometric map with the physical reality are not suitable, it is necessary instead to measure the consistency of the system. The main test indicator is consistency in measured trajectory over time. Over time uncorrected robot odometry drifts in an unbounded manner; the RatSLAM system produces trajectories that are bounded and consistent over time.

The trajectory plots in Fig. 12 and 13 show that using the new visual system RatSLAM is still able to construct a consistent representation of its environment even with inaccurate path integration, Fig. 13. The trajectory plots are smoother and better bounded than those created using the artificial landmark system [1]. Some of this improvement is attributable to changes made to other portions of the system. However in the new system the robot is constantly receiving visual input which allows it to recorrect the trajectory more frequently. The artificial landmark system used only a few landmarks in the environment limiting the opportunities for correcting odometry.

VI. CONCLUSION

The paper has shown that processing visual input in a similar manner to the visual cortex produces useful information for self-localization in unmodified office environments. The use of directionally tuned filters provides feature extraction that is relatively stable across lighting conditions, while the pooling of filter outputs provides robustness to small changes in pose. The sum of absolute differences module provides a sparse vector that is suitable for association with the RatSLAM representation of pose.

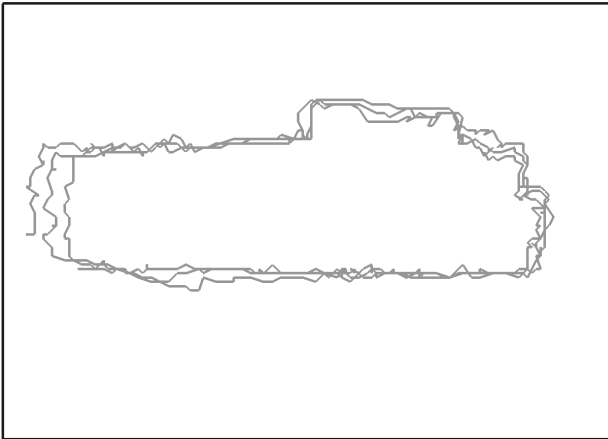


Figure 12. Trajectory of the most strongly over activated pose cell after mapping the test environment. This trajectory indicates that RatSLAM is able to use the new visual system to construct a consistent representation of the environment.

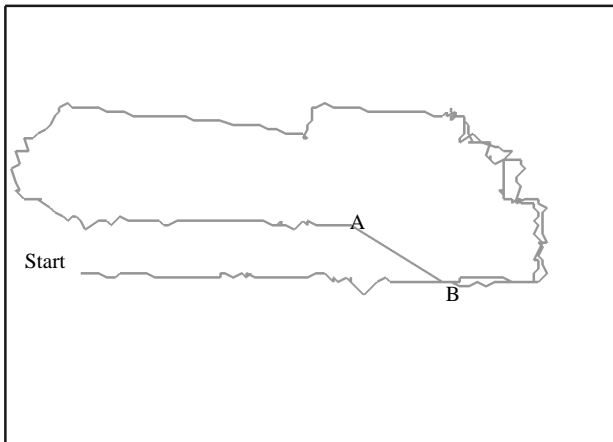


Figure 13. A second trajectory plot shows the RatSLAM system recovering from a major path integration error. The robot adjusts its believed position from point A to point B when sufficient visual information demonstrates that it is incorrectly localized. This result indicates that the visual input is able to relocalize the robot.

The vision system can sometimes produce outputs that are highly ambiguous and unhelpful for re-localization. While using more image features would reduce the number of these ambiguous outputs there will always be some ambiguity if the image learning system is allowed to generalize: natural scenes tend to be ambiguous. The visual

landmark processing system only provides part of the information required to remain localized, and RatSLAM must make use of the current pose estimate and path integration information to localize successfully.

There are a number of areas for further investigation with this system. There has been no significant investigation of the suitability and sensitivity of the various parameters in the system; although the work presented here required little tuning. It would be expected that the performance of the system could be improved further using more features such as complex cells at other orientations and scales. The addition of extra features would also improve the ability of the system to operate in other environments. There are plans to test the system on a visually guided tractor over the next few months.

ACKNOWLEDGMENT

Sincere thanks go the staff of the CSIRO Mining Automation group for allowing us to map their office space in our experiments.

REFERENCES

- [1] M. Milford, G. Wyeth, and D. Prasser, "RatSLAM: A Hippocampal Model for Simultaneous Localization and Mapping," in *Proceedings of the International Conference on Robotics and Automation*, 2004.
- [2] M. Milford and G. Wyeth, "Hippocampal Models for Simultaneous Localization and Mapping on an Autonomous Robot," in *Proceedings of the Australasian Conference on Robotics and Automation (ACRA)*, 2003.
- [3] S. Se, D. Lowe, and J. J. Little, "Vision-based mobile robot localization and mapping using scale-invariant features," in *Proceedings of the IEEE Conference on Robotics and Automation*, vol. 2, 2001, pp. 2051-2058.
- [4] S. Livatino and C. B. Madsen, "Acquisition and Recognition of Visual Landmarks for Autonomous Robot Navigation," in *SIRS'00 - 8th International Symposium on Intelligent Robotic System*, Reading, UK, 2000, pp. 269-280.
- [5] B. J. A. Kröse and R. Bunschoten, "Probabilistic Localization by Appearance Models and Active Vision," in *IEEE International Conference on Robotics and Automation*, vol. 3, Detroit, 1999, pp. 2255-2260.
- [6] F. Pourraz and J. L. Crowley, "Use of eigenspace techniques for position estimation," in *Proceedings of the 5th International Workshop on Advanced Motion Control, AMC '98-Coimbra*, 1998.
- [7] I. Ulrich and I. Nourbakhsh, "Appearance-based place recognition for topological localization," in *Proceedings IEEE Conference on Robotics and Automation*, 2000.
- [8] J.-J. Gonzalez-Barbosa and S. Lacroix, "Rover localization in natural environments by indexing panoramic images," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2002.
- [9] A. Arleo, F. Smeraldi, S. Hug, and W. Gerstner, "Place Cells and Spatial Navigation based on Vision, Path Integration, and Reinforcement Learning," *Advances in Neural Information Processing Systems*, 2001.
- [10] D. H. Hubel, *Eye, Brain, and Vision*. New York, 1988.
- [11] H. Wersing and E. Körner, "Learning Optimized Features for Hierarchical Models of Invariant Object Recognition," *Neural Computation*, vol. 15, pp. 1559-1588, 2003.