2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

The Implementation of a Novel, Bio-Inspired, Robotic Security System

Robert Oates, Michael Milford, Gordon Wyeth, Graham Kendall, Jonathan M. Garibaldi

Abstract— **The implementation of a robotic security solution generally requires one algorithm to route the robot around the environment and another algorithm to perform anomaly detection. Solutions to the routing problem require the robot to have a good estimate of its own pose. We present a novel security system that uses metrics generated by the localisation algorithm to perform adaptive anomaly detection. The localisation algorithm is a vision-based SLAM solution called RatSLAM, based on mechanisms within the hippocampus. The anomaly detection algorithm is based on the mechanisms used by the immune system to identify threats to the body. The system is explored using data gathered within an unmodified office environment. It is shown that the algorithm successfully reacts to the presence of people and objects in areas where they are not usually present and is tolerised against the presence of people in environments that are usually dynamic.**

I. INTRODUCTION

Physical security systems are typically manned systems, consisting of guards patrolling and monitoring the environment. In some cases these systems are augmented with CCTV stations. In these situations a human operator is responsible for monitoring the environment from a base station receiving images from a static sensor network. This solution has two key advantages over the solely manual technique: by using a distributed network of sensors, a single operator can observe a large number of distant locations simultaneously; in the event of intruders entering the building, the operator can monitor the situation without coming to harm. However, as the number of camera feeds to be monitored increases, the probability of a human operator missing an event of interest also increases [18].

The use of mobile robotics within a physical security setting has the advantage of being able to explore the environment more thoroughly than a static sensor network. In addition, some sensor types, (such as chemical sensors) have very limited range, so when deployed statically they can only be effective at bottle necks. Such sensors can be used more effectively on a mobile robotic platform, as they can be moved around the environment. However, the issue of a human operator monitoring multiple feeds is still problematic and intuitively it is likely that monitoring a diverse range of sensors could make the monitoring problem worse. As such, automating the detection of events of interest would be advantageous for any system, as feeds could alert a human operator to situations that may require intervention.

From a robotics perspective the physical security problem has been typically sub-divided into two, distinct problem areas, one of routing and one of anomaly detection. [10] views the routing problem as an optimisation task, reducing the probability of missing an event of interest. [9] views the routing problem as one of reducing the predictability of the robots' movement to make avoiding them harder. Both such techniques rely heavily on accurate localisation and mapping. Visual anomaly detection is typically carried out using simple optic flow metrics and infra-red body-heat sensing [2] or colour-based histogram techniques [1]. Combining optic flow with infra-red sensors will only provide "person detection", useful for warehouse environments and at night, but ineffective in populated environments. Colour-based techniques have the advantage of reacting to both the presence of new objects and the absence of objects that should be there, but can be computationally expensive and require a large, localised database of sample images.

In this paper we present a technique for implementing a robotic security system that combines a neural-inspired algorithm for localisation and mapping and an immune-inspired algorithm for adaptive anomaly detection. By combining these two algorithms it is hoped that a scalable security solution can be constructed which takes advantage of localisation data to detect anomalous situations. The paper is laid out as follows: in Section II the two, bio-inspired algorithms used for this system are presented; in Section III the security system is outlined; in Section IV experimental results are presented and discussed; finally in Section V conclusions are drawn about this technique and future directions are outlined.

II. NEURAL-IMMUNE MODELS

A. The Dendritic Cell Algorithm

The dendritic cell algorithm (DCA) is a recent addition to the field of artificial immune systems. An explanation of the biology of dendritic cells is beyond the scope of this paper but the interested reader is referred to [3] and [5] for explanations of the derivation of the algorithm from the biology and to [8] for a full description of the underlying biological theory. The DCA is a population-based, signal-processing and decision-making algorithm that has been successfully applied within the field of computer security [4]. It has been shown that the algorithm contains an emergent filtering characteristic [16] that makes it suitable for decision making in uncertain environments. [15] demonstrated that the algorithm was light-weight enough to run in real-time on a mobile robot and could be introduced in to a robotic control system as a behaviour within a subsumption architecture.

Robert Oates, Graham Kendall and Jonathan M. Garibaldi are from The School of Computer Science, The University of Nottingham, Jubilee Campus, Wollaton Road, Nottingham, NG8 1BB, UK. Michael Milford^{ab} and Gordon Wyeth^{a} are from The School of Information Technology and Electrical Engineering^{a} and the Queensland Brain Institute^{b}, at The University of Queensland, Brisbane, QLD 4072, Australia

978-1-4244-2789-5/09/$25.00 ©2009 IEEE

The inputs to the DCA are a series of heuristics, two or more of these are time varying signals and one is always a stream of an enumerated type. The enumerated type stream, termed "antigen" allows information about the problem environment to be conveyed to the algorithm. In the port-scan detection application the antigen was the process ids of the programs currently active on the system. For the robotics application an enumeration that represented the robot's pose was used. The signal streams are used to classify the antigen stream items as being either "normal" or "anomalous". Signal streams produce real-valued numbers between 0 and 100. The *PAMP* signal, (named after a biological term) can be seen as the probability that a known anomalous condition is occurring. In biology pathogen-associated molecular patterns (PAMPs) are specific patterns that a species evolves to associate with common pathogens and infections. In some applications this level of certainty is not possible and it is acceptable to implement the DCA with no *PAMP* signal [6]. The *Danger* signal increases as the probability of an unknown anomalous event increases. As this mechanism is attempting to react to an unknown intruder, it is considered to have a higher false-positive rate than that of a *PAMP* signal. The *Safe* signal is inhibitory and is used to suppress the algorithm's response in situations where it is known that the *PAMP* and *Danger* signals are prone to error. Figure 1 is a diagrammatic representation of the DCA.

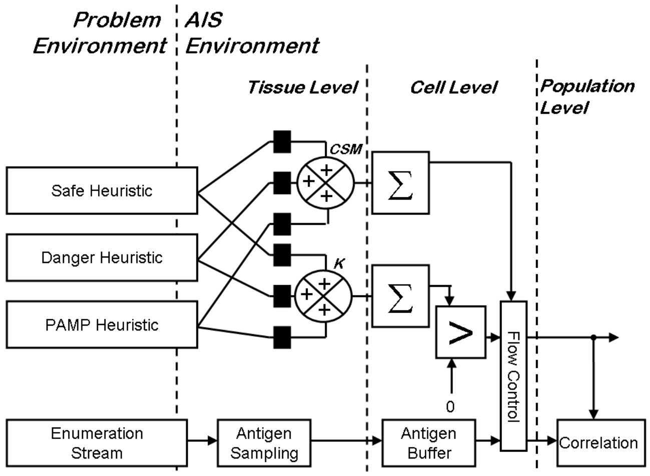

Fig. 1. A representation of the dendritic cell algorithm.

The real-valued input signals are first combined using weighted sums to produce intermediate signals which are in turn presented to the cell population. The weightings for these sums have been derived from biological dendritic cells [3]. The *CSM* signal is a intended to be a measure of how much useful information the current signal sample provides. By summing this over the life-time of a cell, it is possible to quantify how ready a cell is to make a decision. When the accumulation of *CSM* signal hits a cell-specific migration threshold the cell performs its classification. The classification is based on the cell's cumulated value of the *K* signal. Where this summation is greater than or equal to zero, a cell will classify all collected antigen as anomalous. Where this summation is less than zero, a cell will classify all collected antigen as normal. There are several techniques for combining the output classifications of the cells [6]. For this application we will only consider the average classification for a given antigen, referred to as the "mean context antigen value" or MCAV. The selection of an appropriate probability distribution to allocate the migration thresholds of the cell population has been shown to be of crucial importance to the performance of the algorithm [16],[5]. Typically trial and error is used on test data to ascertain the best choice.

B. RatSLAM

RatSLAM is an established biologically inspired robot SLAM and navigation system, based on the mechanisms of navigation in the rodent brain [12], [13], [14]. At its core is a continuous attractor network of *pose cells*, which represent the three dimensional pose state of the robot, shown in Figure 2. Each cell is connected to proximal cells by both excitatory and inhibitory connections, with global inhibition and normalization also applied. The network connectivity is such that the stable state is a single cluster of active cells, known as an activity packet or activity bump. The robot's internal estimate of its current pose is read out from the dominant activity packet in the network. Path integration is performed by using robot motion information to shift cell activity. RatSLAM also uses appearance based vision to learn distinct scenes in the environment, each of which is represented by a *local view* cell. As the robot moves around an environment, connections are learnt between active local view cells and active pose cells, binding the appearance of places with the robot's internal representation of that place in the world. When the robot returns to a familiar place, accumulated errors in odometry are corrected when local view cells representing that place activate, and in turn activate the pose cells associated with that place through the local view - pose cell connections.

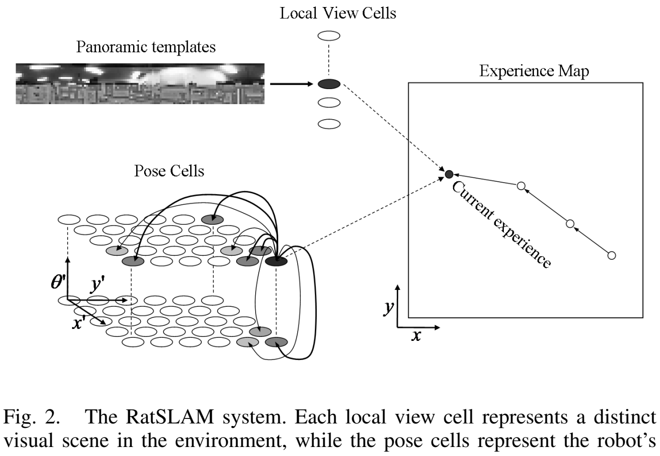

Fig. 2. The RatSLAM system. Each local view cell represents a distinct visual scene in the environment, while the pose cells represent the robot's internal estimate of its location within the environment. The experience map creates a semi-metric graph-like map from the output of the pose and local view cells.

The second major component of the RatSLAM system is an algorithm known as *experience mapping*. The experience mapping algorithm creates a semi-metric graph-like map of the environment containing node-like experiences, which represent distinct spatial and perceptual places in the environment, and transitions, which represent movement between these places. The experience map is used by the RatSLAM system when performing action planning [13], and is also the main input to the DCA described in this paper. Each experience in the experience map represents a distinct combination of local view cell and pose cell activation states, which are the robot's internal representation of the environments appearance and spatial layout. When either the pose or local view cell states are sufficiently different from any of the stored experiences, a new experience is created, with an associated transition from the previous experience. The equation that governs whether to match an existing experience or learn a new one is given by

$$S = 1 - \mu_p |P^i - P| + \mu_v |V^i - V| \tag{1}$$

where P and V are the current pose cell and local view cell activity states, respectively, and μ_p and μ_v weight the relatively contributions to the experience matching process. If the maximum matching score for all existing experiences drops below a threshold S_{min}, a new experience is created. In practice, dynamic parts of the environment result in denser experience representations than static parts of the environment. This is because the RatSLAM perceptual system does not perform any high level processing of scenes to remove dynamic objects such as people or bags. A pertinent example is one of the offices the robot visited, where only on some days a bike was present in the room. The experience mapping algorithm learnt enough experiences to sufficiently localize in both situations, without having any explicit knowledge of the bike as an object.

C. Combining Neural and Immune Models

There is biological justification for combining algorithms taken from immunology and neuro-science. [7] describes the excitatory effect of the stress hormone "DHEAS" on the immune response, whilst the inhibitory effect of the so-called "fight or flight" hormones provides the basis for the use of adrenaline for treating severe allergic reactions [17]. It is of note however that the direct link between the hippocampus and dendritic cells is a simplification.

III. A BIO-INSPIRED PHYSICAL SECURITY SYSTEM

A. Combining the DCA with RatSLAM

A crucial stage in the implementation of the DCA is the selection of appropriate input heuristics. For this application normality can be viewed as the standard operation of the building. For a physical security application the most analogous source of a *PAMP* signal would be a specific recognition algorithm for a universally dangerous situation, such as a building fire or the absence of an item of specific interest. Here it is the interaction between the neural model and the immune model that is of interest so this signal will be left out. For the *Danger* signal it was decided to monitor RatSLAM's localisation score, S. In RATSlam a low score implies that the robot is lost and a high score indicates that the input data corresponds to the algorithm's current estimate of pose. From analysis of the algorithm's performance it is possible to discern that any value of S greater than 80% represents a good estimate of pose, while any value below that represents an increasingly poor estimate, with 0 representing a total localisation failure. Using this knowledge the *Danger* heuristic was generated using equation 2.

$$Danger(S) = \begin{cases} 100 - S, & \text{if } S \geq 20 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

The purpose of the *Safe* signal is to ensure that the immune response is tolerised against circumstances where a certain level of abnormality is acceptable. An example of this mechanism in action within biology is found in the stomach, which is filled with foreign proteins as part of its normal operation so an immune response would be unwanted. As discussed, the RatSLAM algorithm stores "experiences" to assist in the mapping of an environment. The number of experiences required to map dynamic environments is higher than the number of experiences required to map largely static or uniform environments. As a result it is possible to quantify the typical level of 'abnormality' for a given environment by calculating the density of the experience points around that position. Figure 3 demonstrates the variation between experience point density for four regions of space. In order to convert experience density for a given point in space into a usable *Safe* heuristic it is necessary to define two parameters: the radius r of the circle to be drawn around that point; and the upper density N to be used that will correlate to 100% signal output. These parameters will be explored as part of this investigation. It is of note that the RatSLAM algorithm regularly prunes experiences that have been unhelpful. In this way the algorithm should adapt to night-time conditions and re-tolerise against busy environments in the morning after a period of adjustment. The *Safe* signal is expressed in equation 3

$$Safe(x, y) = \frac{100 \times \sum_{i=0}^{E} f(x, y, x_E, y_E)}{N} \tag{3}$$

Where N is the number of points required to achieve the maximum *Safe* signal of 100, E is the total number of experience points created and $f(x, y, x_E, y_E)$ is given in equation 4.

$$f(x, y, x_E, y_E) = \begin{cases} 1, & \text{if } \sqrt{(x - x_E)^2 + (y - y_E)^2} < r \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

In previous robotic applications of the DCA a grid-based system was used to relate pose to an enumeration of state. This was ultimately flawed as using a enumerated type to describe physical position in space limited the area that

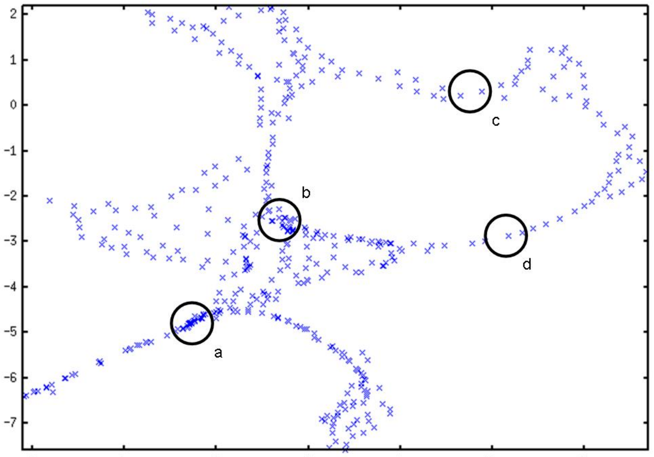

Fig. 3. Generating the *Safe* signal. The experience density for a given point in space is calculated by identifying the number of experiences (crosses) that are contained within a circle of radius r centered on that point. The parameter N scales the number of experiences to be between 0 and 100 where 100 correlates to N or more experiences. Using this metric, the points at the center of circles a and b would generate high levels of *Safe* signal and the points at the center of circles c and d would generate low levels of *Safe* signal.

the algorithm could be used in. RatSLAM uses a toroidally mapped, three-dimensional grid to represent pose. It has been shown that this wrapping of the representation of space is effective over significant distances [11]. As there is clearly a finite number of pose cells within the RatSLAM algorithm, it is easy to map the most active pose cell as the antigen being presented to the dendritic cell population.

B. Monitoring a Cluttered Office Environment

To explore the properties of the security system, it was presented with captured data from a cluttered, unmodified office environment. This data was gathered using a Pioneer 3DX robot using a parabolic mirror for localisation and a laser range finder for obstacle avoidance. Approximately 30 minutes of data was analysed. Logged data was used to aid repeatability for exploring the parameterisation of the system.

For the RatSLAM part of the system, the standard configuration was used. The input was from the lower half of a feed from a parabolic camera attached to the top of the robot. Laser and sonar sensors were used for obstacle avoidance.

It is standard practice to use a uniform distribution of migration thresholds centered around 15, 30, 60, 120 and 240 when identifying the appropriate range to use. In each case the width of the distribution is set to ±10% of the center. As the *Safe* signal heuristic is also parameterisable, a range of values of r will also be explored. The results for $r = 0.05, 0.1.0.2, 0.3$ and 0.5 will be presented here, (units in meters). N will be set to the maximum experience density found in the first few minutes of test data.

In each experiment, the performance of the algorithm will be compared to the results of a human operator identifying people walking around within the environment. Rather than simply identifying peaks where the algorithm successfully identifies a threat, the instantaneous difference between the human operator and the algorithm will be calculated, thus penalising the algorithm for a late identification or for prematurely returning to the "safe" state. For the purposes of analysis the output from the first 300 frames, (30 seconds) will be ignored. In this time the robot is considered to be lost, as it attempts to localize in the environment so the output from the algorithm is invalid.

TABLE I

THE VALUES USED FOR N, THE NUMBER OF EXPERIENCES WITHIN THE CIRCLE THAT CAUSED 100% OUTPUT FROM THE *Safe* SIGNAL.

r (m)	N
0.05	10
0.1	19
0.2	28
0.3	35
0.5	48

IV. RESULTS

Figure 4 displays the results from 25 experiments using the different values of r and the migration threshold distributions. In each case the width of the distribution was set to ±10% of the center point. Table I shows the number of experience points required for each radius size to generate 100% *Safe* signal.

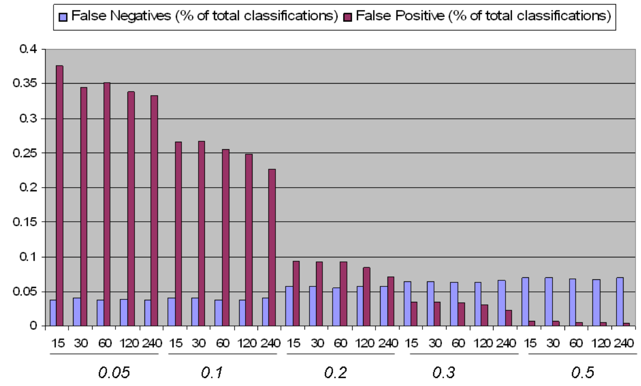

Fig. 4. The results from the 25 experiments. The small x-axis labels indicate the center point of the migration threshold and the larger x-axis labels indicate the groups of experiments with the same r value.

Increasing the radius of effect for the *Safe* signal reduced the false positive rate significantly. However, there was a sudden fall from 0.1 to 0.2 which was associated with the largest rise in the false negative rate. The choice of migration threshold did not appear to have a significant effect on the false negative rate, but there was a noticeable variance in the false positive rate.

A. *Discussion of Results*

The input parameters had a significant effect on the performance of the algorithm which was to be expected for a system which relies so heavily on expert knowledge for tuning. It is arguable that as a security system the performance of the algorithm with a radius of 0.3 and a migration threshold distribution centered on 240 had the best performance. This combination of inputs gave a false positive rate of 0.02 and a false negative rate of 0.07. As an augmentation of a manned security system these are promising results as the low false positive rate is likely to prevent alerts from being ignored by the operator. The danger of such a system is that the operator could grow complacent, so the slightly higher false negative rate could be a problem. In a genuine security situation it is likely that the majority of what a security guard observes is "normal" behaviour and requires no intervention, a key advantage of this system is that it performs consistently regardless of how long it is in service. Figure 5 shows the output of the best performing system over time.

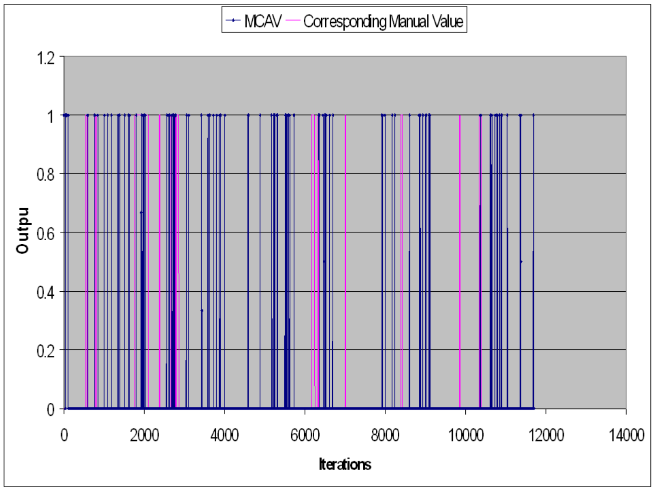

Fig. 5. The output from the best performing system.

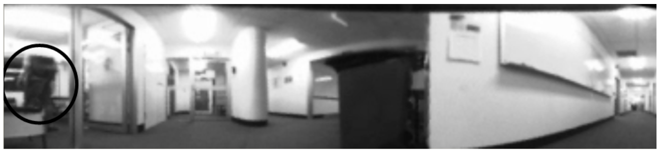

Fig. 6. Problems with occlusion. The figure (circled) has the bottom half of their body occluded by an object. As only the bottom half of the image is processed by the algorithm, this generates a false negative.

Secondly the human operator was only looking for individuals walking within the environment. The algorithm would ignore those pulses found in areas that were usually busy. The missed pulse at 2960 corresponds to an office environment, where the objects, such as chairs etc are regularly moved. In this case the *Safe* signal rises consistently to approximately 20 which inhibits the response to someone walking in front of the camera. For asecurity system, ignoring areas which are typically cluttered and crowded is an advantage.

V. CONCLUSIONS AND FUTURE WORK

A novel physical security system has been outlined based on the anomaly detection properties of the immune system and the localisation and mapping properties of the hippocampus. By merging two successful algorithms from two different fields it has been possible to explore parameterisation and architectures that produce adequate real-world results in a cluttered office environment. No explicit training or filtering was required for this system and the synergy of using metrics already relevant to localisation and mapping for anomaly detection has obvious computational savings.

In the future this system would benefit from the addition of a *PAMP* signal to recognise specific situations. This could be achieved using a standard classifier trained on data from artifical re-creations of those circumstances, or in the case of fires, a heat or smoke sensor could be phsically connected to the robot. Additional sources of *Danger* and *Safe* to encapsulate information such as time of day could potentially vastly increase the usefulness of this system. For example it would be possible to automatically increase the *Danger* signal at night to increase the reactivity of the system or suppress the system when the robot's pose estimate moved into a foyer or other such area.

VI. ACKNOWLEDGMENTS

The authors would like to thank MobileRobots Inc for their support of this research project. M.M. and G.W. would like to thank the Australian Research Council and National Health and Medical Research Council for support of this project through the Thinking Systems initiative.

REFERENCES

[1] Castelnovi M, Miozzo M, Scalzo A, Piaggo M, Sgorbissa A and Zaccaria R (2003). "Surveillance Robotics: Analysing Scenes by Colours Analysis and Clustering". Computational Intelligence in Robotics and Automation (1):229-234

[2] Everett H, Gilbreath G, Heath-Pastore T and Laird T (1994). "Controlling Multiple Security Robots in a Warehouse Environment". AIAA/NASA Conference on Intelligent Robots

[3] Greensmith J, Aickelin U and Twycross J (2006). "Articulation and Clarification of the Dendritic Cell Algorithm". The 5th International Conference on Artificial Immune Systems 404-417

[4] Greensmith J, Twycross J and Aickelin U (2006). "Dendritic Cells for Anomaly Detection. IEEE Congress on Evolutionary Computation

[5] Greensmith J (2007). "The Dendritic Cell Algorithm". PhD Thesis. University of Nottingham

[6] Greensmith J, Aickelin U (2008). "The Deterministic Dendritic Cell Algorithm". The 7th International Conference on Artificial Immune Systems 291-302

[7] Loria RM (2002). "Immune up-regulation and tumor apoptosis by androstene steroids". Steroids 67 (12): 95366. PMID 12398992.

[8] Lutz M and Schuler G (2002). "Immature, Semi-Mature and Fully Mature Dendritic Cells: Which Signals Induce Tolerence or Immunity?". Trends in Immunology 23 (9)

[9] Martins-Filho LS and Macau EEN (2007). "Trajectory Planning for Surveillance Missions of Mobile Robots". Autonomous Robots and Agents 76 109-117

[10] Massios N and Voorbraak F (1999). "Hierarchical Decision-Theoretic Robotic Surveillance". International Joint Conference on Artificial Intelligence, Workshop on Reasoning with Uncertainty in Robot Navigation

[11] Milford MJ and Wyeth G (2008). ”Single Camera Vision-Only SLAM on a Suburban Road Network”, proceedings of the International Conference on Robotics and Automation, Pasadena, United States.

[12] Milford MJ, Wyeth G, Prasser, D (2004). ”RatSLAM: A Hippocampal Model for Simultaneous Localization and Mapping”, proceedings of the International Conference on Robotics and Automation, New Orleans, United States.

[13] Milford MJ, Wyeth G, Prasser, D (2006). ”RatSLAM on the Edge: Revealing a Coherent Representation from an Overloaded Rat Brain”, proceedings of the International Conference on Intelligent Robots and Systems, Beijing, China.

[14] Milford MJ (2008). ”Robot Navigation from Nature: Simultaneous Localisation, Mapping, and Path Planning Based on Hippocampal Models”, Springer Tracts in Advanced Robotics.

[15] Oates R, Greensmith J, Aickelin U, Kendall G and Garibaldi JM (2007). “The Application of the Dendritic Cell Algorithm to a Robotic Classifier”. The 6th International Conference on Artificial Immune Systems 204-215

[16] Oates R, Kendall G and Garibaldi JM (2008). “Frequency Analysis For Dendritic Cell Population Tuning”. Evolutionary Intelligence 1 (1):145-157

[17] Serafeim A and Gordon J (2001). “The Immune System Gets Nervous”. Current Opinions in Pharmacology (1):398-403. PMID 11710739

[18] Tickner AH and Poulton E (1973). “Monitoring up to 16 Synthetic Television Pictures Showing a Great Deal of Movement”. Ergonomics 14 (4)