PURPOSEFUL PERCEPTION BY ATTENTION-STEARED ROBOTS

Bayu Slamet, Arnoud Visser

Informatics Institute, University of Amsterdam,
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands,
{baslamet, arnoud}@science.uva.nl,
http://www.science.uva.nl/research/ias/multi_agents/

Abstract

Designers of autonomous systems, embodied in an uncertain environment, have the tendency to build up a world model from everything that can be perceived. In contrast to this view, psychological researchers find for humans a selective interpretation of a scene, with phenomena like inattentional blindness. Objects remain unseen if they are not central to the current behavior, even while they are clearly within view. Previous research in the Dutch Aibo Team has proven that also for robots behavior-specific image processing can be very beneficial. In this article we design an experiment where we can not only indicate the appropriate moments to limit the perception to the objects relevant to the task, but also indicate the appropriate moments to release those limitations and to increase the overall situation awareness.

1 Introduction

In symbolic AI, knowledge representation is seen as a prerequisite to informed action. To enable generic problem solving, a representation should be maintained that contains complete descriptions of the environment. In robotics, the Sense-Reason-Act loop is popular, where a world model is maintained by continuously processing the sensor readings. Based on this world model, the reasoning engine takes asynchronous decisions, which are executed on a lower layer, in the form of behaviors. In a software architecture, the processing of the sensor readings can be decomposed in several steps, together forming a pipeline. Traditionally, the pipeline is generic, independent of the behavior performed at that moment.

Brooks has already indicated the benefits of a behavior based approach [1], with parallel processing paths to attain and maintain certain goals. Yet, the processing paths of Brooks' behaviors shared many modules. This is most profound for modules near the actual sensors, because the signals from the actual sensors often need the same pre-processing. In a pure parallel approach, each behavior has a separate pipeline, with dedicated modules. Recently, Mantz [2] has shown for the RoboCup 4-legged League that not sharing the modules in the first stage of the sensing could be very beneficial.

The RoboCup 4-legged League is part of the RoboCup initiative, a platform to promote AI and robotics research. In the 4-legged League a small team [9] of standardized robot dogs (Sony Aibos) play a game of soccer. Typical roles inside the team are goalie, defender and attacker. Each role is associated with a number of basic behaviors. The appropriate behavior is selected on the basis of the percepted model of the world. The robot dogs are equipped with a variety of sensors, but none of the other sensors can compete with the camera on detail and distance. Note that the camera is not mounted on a stable platform, but in the nose of the robot's head, which shakes heavenly when the robot moves at full speed, and is often used to handle the ball. Yet, a reliable world model is a prerequisite before any intelligent behavior can be achieved.

2 Perception Model

A major challenge in this League is the creation of a reliable world model based on the images of the camera mounted in the head of the robot dog, while the robot is moving fiercely in a crowded environment. This process has to be performed in real-time on board of the robots. The creation and maintenance of the world model can be divided in two stages:

1. Object recognition
2. Object modeling

The first stage is purely Bayesian, where the current image is compared with the previous image, while the objects in the last stage are lasting. The object recognition process is an example of a transitory representation, with only memory of the previous images to detect changes. In the object modeling process estimates of the location of all objects relevant for the application (robot soccer) are maintained, even when they are not visible. Naturally, the confidence in these estimates decreases as objects are not in view for a longer period of time. These two stages are worked out in more detail in the following sections.

2.1 Object recognition for soccer playing robots

The main perception channel for soccer playing Aibos is vision. Although the robots are equipped with other sensors, no other sensor gives the range and detail that vision can provide. The operating system of the Aibos, Aperios, is a simple real-time system, that allows to synchronize programs with sensors. This makes it possible to define a program that iteratively is called every time a new image frame is grabbed. It also forces the programmers to guarantee that the program is so fast that the previous frame is processed before the next frame arrives.

To support efficient image processing, the Aibo has a hardware supported color-segmentation routine on board. This color-segmentation routine maps the 254*254*254 YUV color information onto 10 possible colors: white, black, yellow, blue, sky-blue, red, orange, green, grey and pink. These are all possible colors of relevant objects on a soccer playing field.

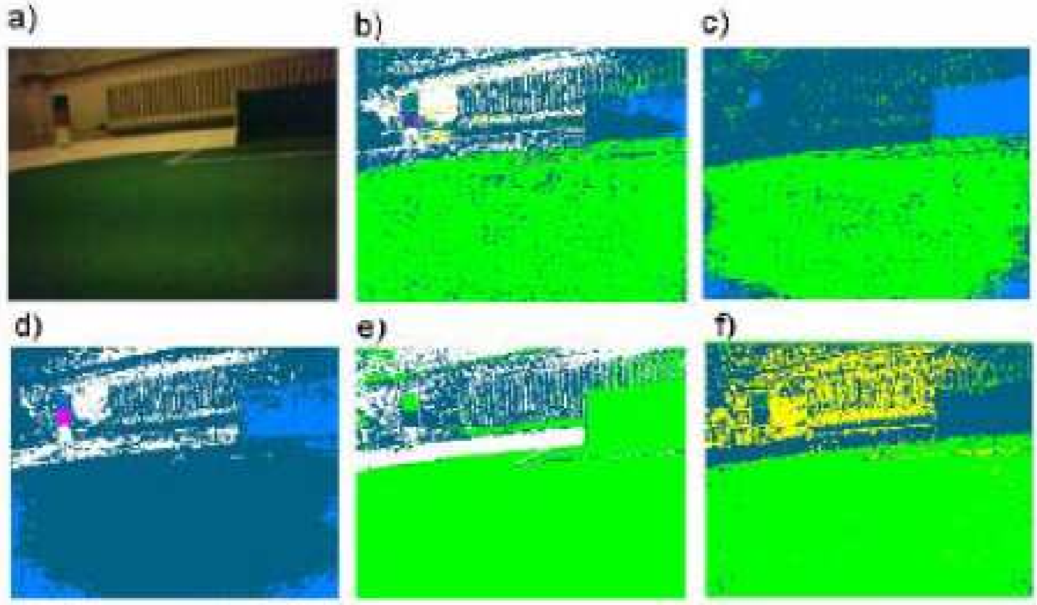

Figure 1: a) camera image (from [4]); b) segmented with the general color-table; c) segmented with the blue/green color-table for the detection of the blue goal; d) segmented with the blue/white/pink color-table for the detection of the blue flag; e) segmented with the green/white color-table for the detection of the field lines; f) segmented with the yellow/green color-table for the detection of the yellow goal.

Segmentation of the image is the primary step in the processing of the image, which makes it a main factor that influences the quality of the information resolved from the image. This explains the extensive research ([3],[7]) to optimize and automate the calibration of the color-table. The calibration has to be done with an extensive set of images, to guarantee that a segmentation that is optimal for a goalie is also optimal for a striker in the middle of the field. As can be seen from figure 1b, a general color-table is a carefully balanced compromise between the 10 color classes. For instance, the top of the blue flag is half blue / half green. The blue area is just big enough to categorize this object correctly later in the process. Extending the blue color class to segment a larger area as blue is possible, but will introduce blue pixels in the field. Mantz [4] has introduced the notion of object-specific image processing, where different, specialized, color-tables are used to detect objects like the blue goal, the blue flag, the field lines or the yellow goal (see figure 1c-f).

Also, in the next step, object-specific knowledge can enhance the image processing. In 2003 the German Team focused their object search to a region close to the horizon [5]. In 2004, both the German Team [6] and Mantz [4] came to the conclusion that it is valuable to differentiate the region of interest for different objects. The region of interest is defined by a grid of scan-lines.

The last step of the object recognition phase is the actual decision process, where an algorithm determines if the colored shapes at the right locations are really the objects relevant for playing soccer. This step has always used object-specific knowledge; the decision is i.e. based on the shape of the objects. The evolution of the three steps through the years can be illustrated with the following figure:

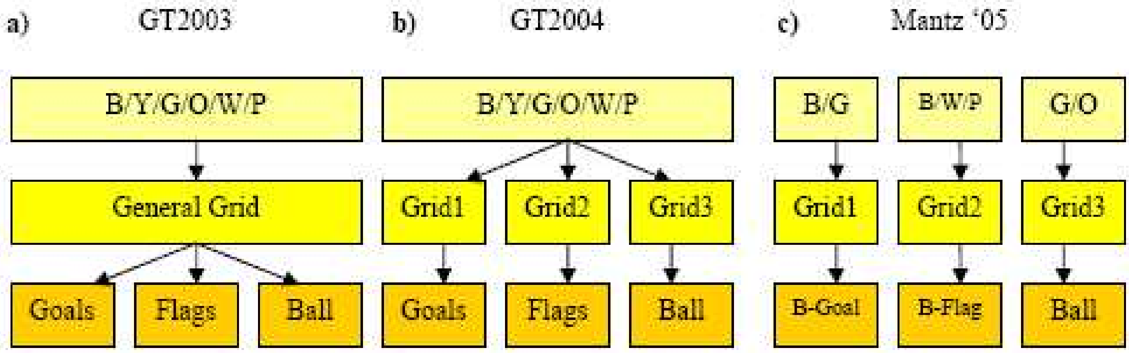

Figure 2: Generic versus object-specific steps in the object-recognition phase: a) architecture of the German Team 2003, only the shape evaluation (bottom) is object-specific; b) architecture of the German Team 2004, different scan-lines grids for different objects; c) architecture of Mantz [4]; different color-tables for different objects; B = Blue, Y = Yellow, G = Green, O = orange, W = White, P = Pink.

Mantz called his approach 'Behavior-Based Vision' [4]. The benefits of this approach are twofold. The first benefit is that stricter algorithms are possible in the last step of the object recognition. Because larger parts of the objects are correctly color-segmented, the shape evaluation can be much stricter which reduces the false acceptance rate [2]. The second advantage is that the color-tables with few colors can divide the color-space so coarse, that the segmentation is able to function in a wide range of lighting conditions [2].

2.2 Object modeling for soccer playing robots

The number of objects that are present on a soccer field is limited, so in principle it is possible that the robot is looking for all objects constantly in order to maintain a complete world-model. Yet, when a robot performs a certain behavior, only information about a subset of the objects is important, information about other objects can only interfere with the perception of the important objects and thus degrade classification performance. For instance, if a goalie is clearing the ball from the penalty area, its main concerns are the ball and the lines bordering the penalty area. All other objects are of secondary importance. Psychological researchers find for humans equivalent strategies of selective interpretation of visual scenes, including phenomena like inattentional blindness [10].

In [8] Torralba criticizes that models of visual attention predominantly focus on bottom-up approaches and proposes to make use of active vision which guides attention to the interesting regions of the image. Mantz' approach [4] nicely fits in his computational framework and seems to share the positive effects of using steered attention. The single difference is that Torralba's approach is based on guiding attention towards regions of interest whereas Mantz focuses attention on colors of interest.

Torralba's computational framework is centered around the evaluation of the probability $P()$ that an object O is present given the set of local measures v_L and context measures v_C: $P(O|v_L, v_C)$. Using Bayes this probability density function is decomposed to:

$$P(O|v_L, v_C) = \frac{1}{P(v_L|v_C)} P(v_L|O, v_C) P(O|v_C).$$

- **Saliency:**The first factor, $\frac{1}{P(v_L|v_C)}$, does not depend on the object searched or the task that is currently executed. Therefore this is a bottom-up factor. The general color table is the equivalent in Mantz' approach. The general color table (not to be confused with object specific color tables) is also not dependent on the currently executed task nor on the object searched.

- **Target-Driven Control of Attention:** The second factor, $P(v_L|O, v_C)$, represents the top-down knowledge of the target appearance. Regions of the image with features unlikely to belong to the target object are ignored and those similar to the searched object are enhanced. This is an exact match to what Mantz uses the object dependent color tables for.

- **Contextual Priors:** The last factor, $P(O|v_C)$, provides context-based priors on object features. This factor does not depend on local measurements and therefore modulates the saliency of local image properties in the search for a particular object O. Given that an object is defined by the tuple $O = [o, \overline{x}, t]$ where o denotes the object class, $\overline{x} = (x, y)$ the object's location and t its appearance parameters like color and shape, Torralba chooses to further split this probability density function by applying Bayes rule several times in order to acquire three factors that model different kinds of context priming on object search:

  $$P(O|v_C) = P(t|\overline{x}, v_C, o) P(\overline{x}|v_C, o) P(o, v_C).$$

  Here $P(t|\overline{x}, v_C, o)$ gives the likely (prototypical) features of objects of a particular class. In Mantz approach this is realized in the object-dependent detection algorithms which make use of specialized scanline grids and specific shape evaluations.

  The subsequent factor, $P(\overline{x}|v_C, o)$, gives the most likely locations for the presence of an object given the context. This is a direct match with how Mantz positions scanlines in grids near the horizon in order to detect target objects.

  The third factor, $P(o, v_C)$, provides the probability density function for objects of a particular type in the scene. Mantz uses the current robot pose (location and orientation on the field) to determine which objects are likely in view.

In addition to this, Mantz implemented and experimented with some simple heuristic rules to further focus the attention of the goalie, which showed to improve the performance another 50% for the soccer domain [2]. Experiments are needed to extend this small rule base to a larger domain, for instance to the behaviors of the other players. Further, there does not seem to be a computational model to relocate the focus of attention. Our hypothesis is that when the information about the objects of interest is reliably perceived for a relatively long period, the attention should linger and a search for other objects should be included. In other words, our robot should get bored of its reliable world-model and broaden its view (see figure 3). We will design an experiment that will demonstrate the benefit of this approach.

Note that with this experimental scenario we limit ourselves to the creation and maintenance of the world model by a single robot. During a real soccer game the different robots can communicate, and exchange their estimations about their objects of interest the teammates. If there is a limited overlap between the objects of interest, the focus of attention leads indirectly to task-decomposition, where each robot is responsible for a subset of the shared world model. For this article we concentrate on a single, fully autonomous robot, namely the goalie.

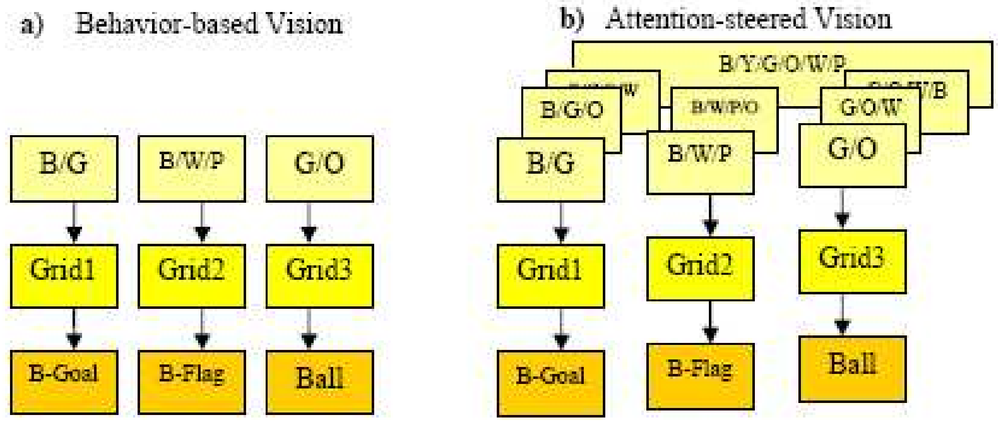

Figure 3: Behavior-based versus attention-steered vision: a) architecture of Behavior-based Vision [4]; different color-tables for different objects; b) architecture of Attention-steered Vision; multiple color-tables for each behaviors; B = Blue, Y = Yellow, G = Green, O = orange, W = White, P = Pink.

3 Experiment for attention steered vision

The goalie has a simple behavior schema (compared with the other players), which consists mainly of three behaviors:

- **Goalie-guard:** When no ball is near, the goalie stands in the centre. It moves its head around and is thus likely to see the field-lines of the penalty area very often and at least one of the two nearest flags once in a while. An image processing algorithm that uses a scan-lines grid above the horizon and a 3-color look-up table that rejects candidates with too small support, was used for detecting the own flags. In addition, an orange/white/green color-table was used for ball- and line detection. All these measurements heavily rely on the assumption that the goalie is in his goal. Therefore, a background process is needed that runs on a slower pace and keeps verifying if this assumption is still valid. For this, the average number of detected lines and flags is evaluated. If this value becomes too low, the behavior signals to the governing behavior that invoked him, that the pre-condition is false (he is likely to be located out-side the penalty area), and the governing behavior switches, e.g., to the behavior *goalie-return-to-goal.*

- **Goalie-clear-ball:** When the ball comes into the penalty area, the goalie will try to clear it, return to the center of his goal and return to the behavior *goalie-guard.* While walking to the ball, the head is aimed at the ball; when controlling the ball, the head is positioned over the ball. In these situations the quality of the sensor input tends to be low. The same image processing solution is used as for the *goalie-guard* behavior, detecting the lines and near flags. Flags and lines detected at far off angles or distances are totally ignored. The algorithm that detects whether the premises for "I am in the goal" are still valid, runs on a lower pace; so the robot works with the assumption that he is standing between ball and goal for a longer while, without actually verifying that this is true.

- **Goalie-return-to-goal:** When the goalie is not in his penalty area, he has to return to it. The goalie walks around while scanning the horizon with its head. When his own position is determined, the goalie tries to walk straight back to goal, meanwhile facing his own goal. When the goalie is back in his goal, he returns (via the governing behavior) to the behavior *goalie-guard.*

The localization algorithm in this behavior mainly relies on the detection of the own goal and detected line-points. To overcome those situations where the own goal is not visible form some place in the field, the two own flags are also used for localization. Notice that in with this behavior the ball is completely ignored. The three behaviors of the goalie are illustrated in figure 4:

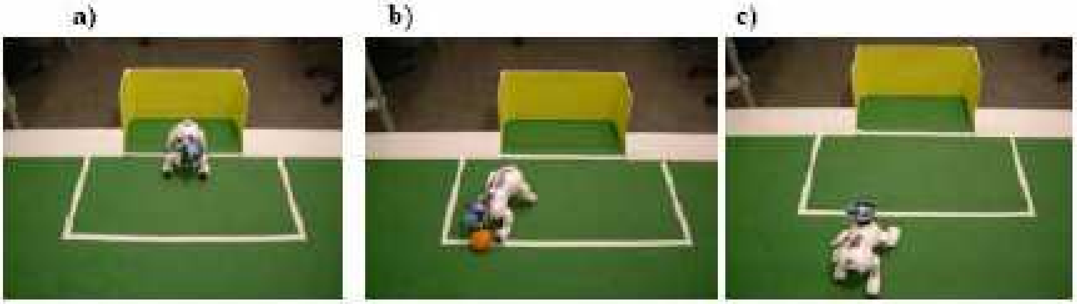

Figure 4: Snapshots of the basic goalie behaviors (from [4]): a) *goalie-guard*, the goalie stands in his goal; b) *goalie-clear-ball*, the goalie clears a ball from the penalty area; c) *goalie-return-to-goal*, the goalie finds its way back to its goal.

Previous experiments with the *goalie-return-to-goal* behavior have shown that in three minutes the robot can typically return from a corner at its own half of the field to the center of its goal six times. At the center of its goal the *goalie-guard* behavior becomes active, and the robot starts looking for the ball again.

In this article we extend this experimental scenario with an opponent striker, starting with the ball in the other corner on this half of the field. While the goalie is returning, the opponent dribbles to the goal, until it reaches a position where it can turn and shoot (see figure 5). For the pure behavior-based vision, the goalie focuses on finding the blue goal and ignores the ball until the center of the goal is reached. The result is that the goalie will be often too late to block the shoot. It should be beneficial to search for the ball earlier. Yet, from the corner it has no clear view on its own goal yet, so the goal (an important object of interest for the *goalie-return-to-goal* behavior) will be quite late perceived reliably for a relatively longer period of time. So, when the goalie starts to look for the ball too early, the chance that the goalie will not find its goal at all increases.

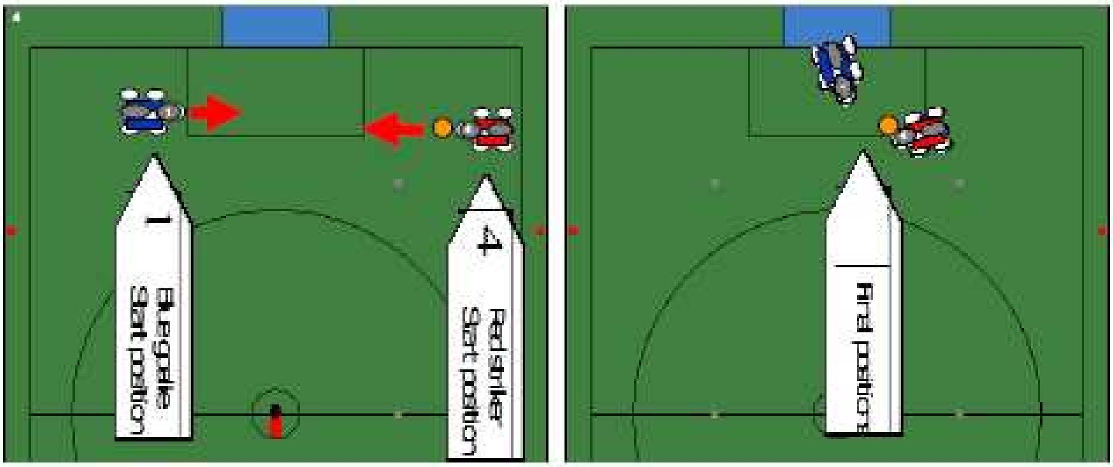

Figure 5: An overview of the proposed experiment: left the initial, right the final positions.

This experiment will make it possible to find the appropriate moments to relocate the focus of attention based on the reliability of the perceived information over a time period. A comparison can be made between a naive approach which relocates the focus on fixed moments and a more sophisticated approach which relocates the focus when the reliability reaches certain thresholds.

4 Conclusion

In this article we played with the idea of selective interpretation of a scene by soccer playing robots. In previous research, the benefits of the phenomenon corresponding with the focus of attention are indicated. In this article we have concentrated more on the dangers of this focus of attention. We propose to loosen the focus after a period of reliably perceiving all objects of interest, and have designed an experiment to find the characteristics of such a period in our domain, and the benefit that can be gained from such method.

Acknowledgements

We would like to thank all partners of the Dutch Aibo Team, who make it possible to do such a research project in close co-operation with the Delft University of Technology. We especially like to thank Paul van Rossum for the initiating discussions about behavior-based vision. We like to thank Niek Wijngaards for his comments on an initial version of this article. We like to thank the students of the Utrecht University for providing us with such a stable software environment. We like to thank Jürgen Sturm for his deep knowledge of this environment and his efforts to create a reliable experimental environment by solving the problems with calibration and localization.

References

[1] R.A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1), March 1986.

[2] W. Caarls F.A. Mantz, P.P. Jonker. Thinking in behaviors, not in tasks; a behaviour-based vision system on a legged robot. In *Proceedings of the Robocup 2005 Symposium, Osaka, Japan*, July 2005.

[3] M. Jüngel. Using layered color precision for a self-calibrating vision system. In *Proceedings of the 8th International Workshop on RoboCup 2004*, Lecture Notes on Artificial Intelligence. Springer Verlag, 2005.

[4] F.A. Mantz. A behavior-based vision system on a legged robot. Master's thesis, Delft University of Technology, February 2005.

[5] T. Röfer, H.-D. Burkhard, U. Düffert, J. Hoffmann, D. Göhring, M. Jüngel, M. Lötzsch, O. v. Stryk, R. Brunn, M. Kallnik, M. Kunz, S. Petters, M. Risler, M. Stelzer, I. Dahm, M. Wachter, K. Engel, and A. Oster. *GermanTeam RoboCup 2003*. Online, 199 pages, 2003.

[6] T. Röfer, T. Laue, H.-D. Burkhard, J. Hoffmann, M. Jüngel, D. Göhring, M. Lötzsch, U. Düffert, M. Spranger, B. Altmeyer, V. Goetzke, O. v. Stryk, R. Brunn, M. Dassler, M. Kunz, M. Risler, M. Stelzer, D. Thomas, S. Uhrig, U. Schwiegelshohn, I. Dahm, M. Hebbel, W. Nisticó, C. Schumann, and M. Wachter. *GermanTeam RoboCup 2004*. Online, 299 pages, 2004.

[7] D. Schulz and D. Foz. Bayesian color estimation for adaptive vision-based robot localization. In *Proceedings of IROS*, 2004.

[8] A. Torralba. A robust layered control system for a mobile robot. *Journal of the Optical Society of America A*, 20(7):1407–1418, 2003.

[9] N. Wijngaards, F. Dignum, P.Jonker, T. de Ridder, A. Visser, S. Leijnen, J. Sturm, and S. van Weers. Dutch aibo team at robocup 2005. In *Proceedings CD RoboCup 2005, Osaka, Japan*, July 2005.

[10] S. Wood. Representation and purposeful autonomous agents. *Robotics and Autonomous Systems*, 49(1-2), 2004.