A Biologically Inspired Adaptive Working Memory for Robots

Marjorie Skubic[1], David Noelle[2], Mitch Wilkes[2], Kazuhiko Kawamura[2] and James M. Keller[1]

[1]Electrical and Computer Engineering
University of Missouri-Columbia, Columbia, MO 65211
skubicm@missouri.edu / kellerj@missouri.edu

[2]Electrical Engineering and Computer Science
Vanderbilt University, Nashville, TN 37235
david.noelle@vanderbilt.edu / wilkes@vuse.vanderbilt.edu / kawamura@vuse.vanderbilt.edu

Abstract

In this paper, we discuss the motivation, approach, and status of a new NSF ITR project in which an adaptive working memory is investigated for robot control and learning. There is much evidence for the existence of such a memory structure in primates. Such memory is closely tied to the learning and execution of tasks, as it contributes to decision-making capabilities by focusing on essential task information and discarding distractions. We will integrate the adaptive working memory structure into a robot to explore the issues of task learning in a physical embodiment. This leads to a complex but realistic system involving perceptual systems, actuators, reasoning, and short-term and long-term memory structures. In the paper, we discuss also planned experiments intended to evaluate the utility of the adaptive working memory.

Introduction

There is much evidence for the existence of an adaptive working memory system involving prefrontal areas in the primate brain. Such memory is closely tied to the learning and execution of tasks, as it contributes to decision-making capabilities by focusing on essential task information and discarding distractions. Our goal is to develop such a memory for robots to support the following:

- Focus attention on the most relevant features of the current task.
- Support generalization of tasks without explicitly programming the robot.
- Guide perceptual processes by limiting the perceptual search space.
- Provide a focused short-term memory to prevent the robot from being confused by occlusions, i.e., to avoid the out of sight, out of mind, problem.
- Provide robust operation in the presence of distracting irrelevant events.

In this paper, we discuss the motivation, approach, and status of a new NSF ITR project in which an adaptive working memory is investigated for robot control and learning. In the project, we will integrate an adaptive working memory structure into a robot in order to provide the embodiment necessary for exploring the issues of task learning. In turn, this leads to a complex but realistic system involving perceptual systems, actuators, reasoning, and short-term and long-term memory structures.

Our investigations will provide insights into the utility of adaptive working memory systems for robot control. Issues in the design of such memory systems will be explored. But there is an additional potential benefit of this research in the domain of computational neuroscience. Some modern accounts of prefrontal function hinge on the ability of a reinforcement learning mechanism akin to temporal difference learning to scale to real-world environments and to real-world tasks. To date, these prefrontal cortex (PFC) models have only been validated against laboratory data. A successful implementation of a model of this kind in a functioning robot would provide some initial evidence that the proposed cognitive mechanisms can address the magnitude of challenges faced by biological organisms every day.

The rest of the paper is organized as follows. Sec. 2 covers background information on the role of working memory, and Sec. 3 discusses our initial efforts in creating a working memory system. Sec. 4 includes discussion on several components required for the robotics systems to run the planned experiments which are discussed in Sec. 5. Concluding remarks are given in Sec. 6.

The Role of Working Memory

Working memory is what allows you to remember the telephone number that the directory assistance operator just recited, retaining it only long enough to dial it yourself. Working memory is what allows you to stay focused on the search for a specific product on the shelves of a grocery store after looking up from your shopping list. Working memory is what allows you to keep track of the particular location in which you last had a clear view of a stalking wolf. Working memory systems are those that actively maintain transient information that is critical for successful decision-making in the current context.

Cognitive scientists have gathered evidence for a variety of memory systems in the mammalian brain (Squire 1987). One dichotomy, introduced early in the history of psychology (James 1890), has been between short-term memory and long-term memory, with the former being more prone to rapid forgetting. Extensive research has led to a more fractionated view of biological memory systems, with multiple dissociable cognitive processes being identified for both short-term and long-term retention. Long-term memory has been found to include somewhat segregated systems for semantic knowledge and for specific episodes (Tulving 1972), and memory in general has been found to have both implicit (i.e., inaccessible to consciousness) and explicit components (Warrington and Weiskrantz 1968). Separate short-term rehearsal mechanisms have been found for visual-spatial information and for speech-based information (Baddeley 1986). Working memory has been identified as a mechanism that protects a small number of informational "chunks" from interference and distraction and places them in a position to directly influence behavior (Goldman-Rakic 1987).

While contemporary theories of working memory are diverse (Miyake and Shah 1999), they are united in ascribing a few key properties to this cognitive system. A distinguishing characteristic of working memory is its highly limited capacity. Early estimates placed the capacity of working memory at "seven plus or minus two" items (Miller 1956), but more recent research suggests that this estimate is high (Cowan 2001). Working memory can be updated quickly, with its contents dynamically manipulated at sub-second rates, while many longer-term memory systems are much slower to adapt (Waugh and Norman 1965). Also, the contents of working memory are seen as readily available to deliberative and executive control processes (Norman and Shallice 1986).

These basic properties of working memory interact in interesting ways. Of specific interest is the limited capacity of working memory, which constrains the amount of information immediately available to explicit reasoning and executive control processes. This suggests that the working memory system must be specially tuned to select for retention only that information which is important for guiding controlled behavior in the current context.

There is extensive and converging evidence that the working memory systems of the PFC are capable of representing and actively maintaining different types of information, including information about spatial locations, about recently viewed objects, about sought or expected objects, and about rules for action. There is also experimental evidence for the encoding of verbal forms of information (Demb et al. 1995). Despite this diverse representational repertoire, the PFC working memory system is still limited in capacity. Thus, the contents of working memory must be carefully selected, and there is evidence that the activity of prefrontal cells specifically capture information that is critical to the performance of the current task (Rainer et al. 1998).

Initial Working Memory Efforts

A central focus of this project involves the incorporation of adaptive working memory mechanisms into robotic control systems, with the goal of establishing the utility of working memory for robot deliberation and learning. These mechanisms are inspired by the hypothesized contribution that the PFC of the human brain makes to working memory and cognitive control. A working memory system can be viewed as a relatively small cache of task relevant information, which is strategically positioned to efficiently influence behavior. The small size of this memory and its tight coupling with deliberation mechanisms alleviates the need for costly memory searches or retrievals. Information needed to fluently perform the current task is temporarily kept "handy" in the working memory store.

The primary challenge for such a working mmory system is determining whether a given chunk of information should be maintained in working memory or not. If the robot is to perform a relatively routine task which is well understood by the designers, it would be reasonable to hand code procedures for the identification of chunks needed. However, if the robot is expected to flexibly respond in novel task situations, or even acquire new tasks, then hand tuning becomes excessively laborious. In these cases, it would be beneficial for the robot to be equipped with a means to *learn* when to store a particular chunk of information in working memory.

The problem of learning when to update the contents of working memory has been explored by computational neuroscientists studying the working memory systems of the PFC. This brain area receives dense inputs from neurons in the midbrain that communicate using a neurotransmitter called dopamine. Electrophysiological research has recently indicated that these dopamine cells code for *changes in expected future reward*. This has led computational neuroscientists to posit that these dopamine neurons are involved in a kind of reinforcement learning called *temporal difference (TD) learning*, since this class of learning algorithms depends critically on a measure of change in expected future reward.

Temporal difference learning is a powerful method for learning to select actions based only on experiences that include reinforcement signals: scalar good/bad signals sporadically provided during behavior. Importantly, TD learning can also be used to select *covert* actions, such as deciding to retain a chunk of information in working memory. Computational PFC models that use TD learning mechanisms to capture the influence of dopamine cells have been successful at explaining human and non-human primate performance on a variety of working memory tasks (Braver & Cohen 2000; O'Reilly et al. 2002). The question at hand is if the same computational mechanisms can facilitate robot control and learning by providing a useful adaptive working memory.

The first step in addressing this question necessarily involves the construction of tools for building working

memory systems that integrate tightly with robot control mechanisms. We have designed and are in the process of implementing such software tools. The result will be a *working memory toolkit*: a software library supporting the creation and use of adaptive working memory systems within a variety of robot control frameworks. A primary design goal has been to make this toolkit as general and flexible as possible, facilitating its use in a wide variety of domains. In particular, no commitment is made concerning the contents or format of the chunks of information stored in working memory. The salient features of the robot's state (including the features of considered informational chunks) which are to be used to decide if a given chunk is worthy of retention are all specified by the user of the toolkit, rather than by the toolkit, itself. Reward signals are also provided by the user. In short, the potential contents of working memory and the reinforcement signals that allow the system to adapt are completely configured by the user, while the working memory toolkit proper provides the TD learning mechanisms needed to learn to remember some informational chunks and to ignore others.

The toolkit is being implemented in ANSI C++, offering an object-oriented organization to the code. The main working memory object is configured by the user to contain a fixed number of "chunks", each of which is implemented by the toolkit as an untyped pointer to a data structure provided by the user. In other words, each informational chunk maintained in the working memory can be any C++ data structure, of any type – the working memory simply maintains a pointer to each structure for rapid access to it.

In order to evaluate chunks for potential retention, however, the working memory needs a characterization of the potentially relevant features of each chunk. A chunk will be retained if it contains features that predict a large future reward, given that the chunk is remembered. The ability to predict the likelihood that maintenance of a chunk with a given set of features will lead to future reward is learned by a component of the working memory object called the *adaptive critic*. The adaptive critic is an artificial neural network that learns, from reinforcement, to predict future reward based on an input real vector encoding potentially relevant features of the situation. In particular, the input to the adaptive critic includes the features of a chunk being considered for retention, the features of chunks already maintained in working memory, and other features of the current state of the robot that might be relevant for making this decision (e.g., the current goal of the robot).

Since the working memory toolkit does not limit the kinds of data structures that can be used as working memory chunks, the user must provide a function which extracts from each chunk a real vector encoding of the features of the chunk that suggests the utility of retention of that chunk. The user must also specify a function that offers relevant properties of the robot's current state to the adaptive critic in the form of a real vector. Given a function for extracting features from chunks and another for extracting features from the robot's current state, the working memory object constructs a useful real vector code for the current situation facing the working memory system, and it presents this real vector as input to the adaptive critic. The adaptive critic can then learn, using TD methods, to associate the retention of certain chunks in certain situations with expected future reward.

To assist the user with the process of designing the feature vector functions, the toolkit provides the classes *StateFeatureVector* and *ChunkFeatureVector* which are equipped with utility methods for translating common data types into real-valued feature vector elements.

The learning process embodied by the adaptive critic network requires the presentation of a scalar *reinforcement signal* with each situation, with a positive value indicating a good outcome, a negative value indicating a bad outcome, and a zero value indicating a neutral outcome. In most situations the reinforcement signal will be zero, as rewards (or the lack thereof) generally only arrive after a sequence of actions have been completed. It is the job of the adaptive critic to predict these sparse rewards. In order to inform the working memory of successes and failures, the user must provide a function, which provides the corresponding reinforcement signal for each situation. Typically, this function will translate a sensory event (e.g., the ringing of a bell) into an indication of reward.

The software objects that compose this toolkit are currently being implemented. As an initial test of these tools, we are also implementing a software simulation of a robotic version of the classic neuroscientific working memory task called the *delayed saccade* task. In this task, the robot will be expected to fixate its gaze on a central object (e.g., a crosshair on a computer screen) while another object (e.g., a bright dot) is briefly presented in the visual periphery. Once a "go signal"' is given (e.g., the crosshair vanishes), the robot is to shift its gaze to where the peripheral object was previously presented. This is a very simple task that requires the robot to maintain the location of the briefly presented target in working memory. Critically, the robot will *not* be programmed to perform this task, but will have to *learn* the sequence of behaviors that result in reward. This simple laboratory task will act as an initial test of the learning mechanisms provided by the implemented working memory toolkit.

Enabling Robotic Components

The adaptive working memory system will be integrated into both humanoid and mobile robots to test and evaluate its utility for robot control. In this section, we discuss several components included in this robotic embodiment that will facilitate interactive experiments: the central executive, the spatial reasoning system that supports spatial language, and two approaches to vision-based object recognition. These components provide a rich set of potential memory chunks that may be entered into

the working memory, such as objects, object locations, and actions.

Central Executive

Robots in the future must exhibit robust performance in a wide range of situations, requiring competencies ranging from efficient sensorimotor action control in routine situations to high-level cognitive control to handle new or difficult situations. Towards this goal, we have developed a multiagent-based cognitive robot architecture for our humanoid robot, ISAC (Kawamura et al. 2004) as shown in Fig. 1. Under the current NSF grant, we are developing the Central Executive (CE) within the Self Agent and the Working Memory System (WMS) .

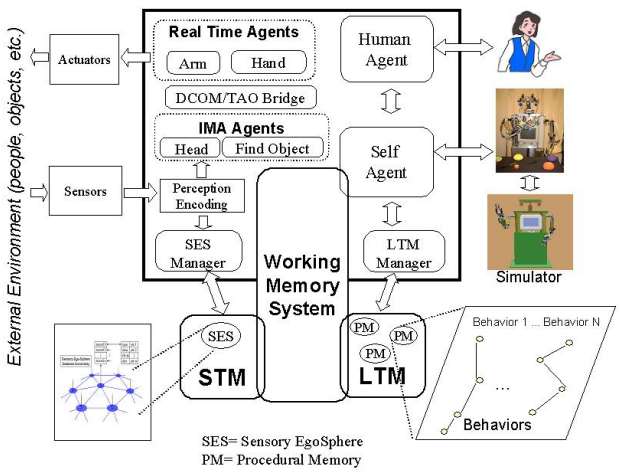


Fig. 1. MultiAgent-based Cognitive Robot Architecture

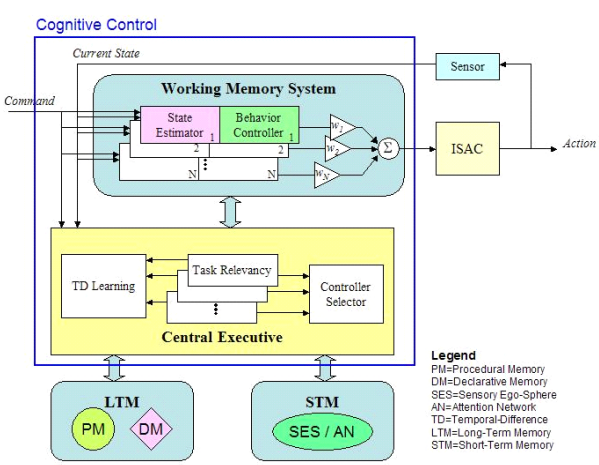


Fig. 2. Current Cognitive Control Implementation

The CE is responsible for high-level executive functions such as a goal-directed action selection process called *cognitive control*. Cognitive (or executive) control is the "ability of the brain to identify possible goals and figure out how to achieve them … and ignore the distractions and impulses that would derail our goal-directed efforts" (Miller 2003). In our case, it involves the control of behavior in situations in which the robot's routine sensorimotor actions fall short of meeting task demands. Within our multiagent cognitive robot architecture, the CE and WMS perform the role of cognitive control. Fig. 2 shows the current implementation.

All behaviors previously taught or learned are stored in the LTM (Erol 2003). When a new command or goal is given, appropriate behaviors are loaded into the WMS based on a decision made by a reinforcement learning algorithm. We plan to integrate the working memory toolkit and use its TD learning for this purpose. Within the WMS, the state estimator and behavior controller are assigned to each behavior. The CE then selects behavior based on *task relevancy* adapted from the original concept of modular controllers and their responsibilities introduced by Wolpert and Kawato (1998). That is, at current time *t*, the CE takes input from the state estimators as the estimated state of the arm $\hat{x}_t^i$, as well as the current state x_t and goal state x^*. Then task relevancy λ_i of the ith behavior that is loaded into the WMS is computed as

$$\lambda_i = \frac{e^{-|ec_i|^2/\sigma^2}}{2\sum_{j=1}^{N} e^{-|ec_j|^2/\sigma^2}} + \frac{e^{-|eg_i|^2/\sigma^2}}{2\sum_{j=1}^{N} e^{-|eg_j|^2/\sigma^2}}$$

where, $ec_i(t) = x_t - \hat{x}_t^i$ is the current state error, and $eg_i(t) = x^* - \hat{x}_t^i$ is the goal state error. The value of task relevancy suggests the level of involvement that each behavior must contribute to the action in the form of time-varying weights $w_i(t)$. The behavior controllers use the weights to control the level of command sent to the arm.

The role of the reinforcement learning system is to select the behaviors that will be appropriate for the task at hand. A new reward is calculated for each set of behaviors after attempting the task. The reward is equal to the average relevancies of behaviors within the set over the time it took to complete the task, discounted by the task completion time. Thus, the set of behaviors that complete the task most quickly with the most precision will be rewarded and are likely to be selected in the future. ISAC learns which set of behaviors to select for each task. New, similar tasks can be built on the learned values.

Spatial Reasoning

The spatial reasoning component includes the capability of modeling spatial relations to support innate spatial cognition as well as linguistic, human-robot communication based on this cognition (Skubic et al. 2003; 2004). Spatial reasoning is accomplished through the use of a spatial modeling component based on the histogram of forces (Matsakis and Wendling 1999). Force histograms are computed from a boundary representation of two objects to provide a qualitative model of the spatial

relationship between the objects. Here, the histograms are computed between an environment object (extracted from sensory data) and the robot to produce an egocentric model of the robot's environment. Features extracted from the histograms are fed into a system of rules (Matsakis et al. 2001) or used as parameters in algorithms to produce linguistic spatial terms.

Interactive Spatial Language

Our spatial language module has been integrated with a multi-modal interface developed by the Naval Research Lab (Perzanowski et al. 2001) that supports voice recognition and natural language processing. Robot spatial language will facilitate interactive experiments to test the working memory, e.g., "find the object to the right of the cup." This type of spatial language provides intuitive communication in many situations. Relative spatial terminology can be used to limit a search space by focusing attention in a specified region, as in "Look to the left of the telephone and find the cup." It can be used to issue robot commands, such as "Pick up the cup on the table." A sequential combination of such directives can be used to describe and issue a high level task, such as, "Find the telephone on the table in front of you. The cup is on the table to the left of the telephone. Pick it up and hand it to me." Finally, spatial language can also be used by the robot to describe its environment, such as, "There is a cup on the table to the left of the telephone."

Dialogs using spatial language can assist in recognizing and labeling objects to overcome limitations in object recognition. Once an object is labeled, the user can then issue additional commands using the spatial terms and referencing the named object, as shown below:

Human: "How many objects do you see?"
Robot: "I see 4 objects."
Human: "Where are they located?"
Robot: "There are two objects in front of me, and one object on my right."
Human: "The nearest object in front of you is a telephone. Place the cup to the left of the telephone."

Currently, commands using spatial references (e.g., go to the right of the table) assume an extrinsic reference frame of the object (table) and are based on the robot's viewing perspective to be consistent with Grabowski's "outside perspective" (1999). In the example, the robot would first turn to face the table and then determine a target point to the right of the table.

There is some rationale for using the robot's viewing perspective. In human-robot experiments, Moratz et al. (2001) investigated the spatial references used by human users to control a mobile robot. The test subjects consistently used the robot's perspective when issuing directives, in spite of the 180-degree rotation. In human to human experiments, Tversky et al. (1999) observed a similar result and found that speakers took the listener's perspective in tasks where the listener had a significantly higher cognitive load.

Spatial Representations

In our previous work, we have used both 2D horizontal planes (an evidence grid map, built with range data) and 2D vertical planes (from image data). Here, we combine them. To achieve the type of interaction described above, it is not necessary to build a full 3D, global representation of the environment. Rather, we obtain range information for a set of objects identified in the image plane. Human spatial language naturally separates the vertical and horizontal planes, e.g., the cup is on the table, vs. the cup is to the left of the telephone. Our linguistic combination utilizes both prepositional clauses, e.g., the cup is on the table to the left of the telephone. Processing the spatial information as two (roughly) orthogonal planes provides a better match with human spatial language.

The position of recognized objects will be stored in a robot-centric frame called the Sensory Ego Sphere (SES) (Peters et al. 2001). The SES is a database implementation of Albus's proposed egosphere (1991). This spatial database provides an egocentric view of the world that is consistent with the robot's viewing perspective. The SES structure is a geodesic dome with a default frequency of 13, yielding a resolution of about 5 degrees with 1680 hexagonally-connected triangles. Each vertex can be labeled with an object identifier; some objects span multiple vertices. Objects may be retrieved using azimuth and elevation angles as indices into the database. An azimuth and elevation may also define a starting point in a search, for example, to look for an object in a specified region. In this case, a breadth-first search is executed using the specified azimuth and elevation as the starting node (Peters et al. 2001).

As a database structure, sensory processes may be adding information to the database in the form of identified objects and ranges, while, at the same time, motor processes may be retrieving information for navigation or manipulation tasks. Because the SES is egocentric, sensory fusion and reasoning can be performed in the robot's inherent reference frame (Albus 1991).

To achieve the type of dialog described above, we will project environment objects onto horizontal and vertical planes. For example, phrases such as "look for the cup to the left of the telephone" will utilize a horizontal plane at the appropriate height. The telephone (stored in the SES with a known range) would be projected onto this horizontal plane and a region to the left of the telephone, also on the plane, would be computed. This left region is then transformed back into the SES to define a starting node for the cup search. In a similar way, a vertical plane can be used to form the relations above, below, and on top.

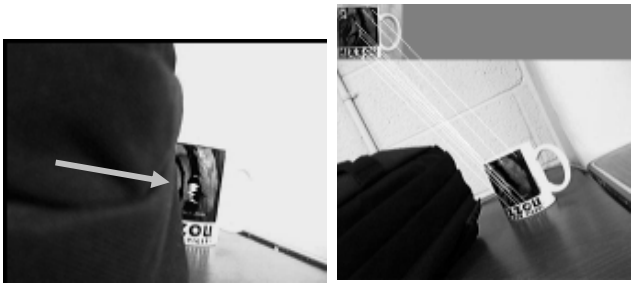
Vision-Based Object Recognition

There are currently two algorithms being studied for object recognition. The first is the Scale Invariant Feature Transform (SIFT) algorithm proposed by Lowe (2004). Lowe's method trains upon a single image and has high

recognition even with rotation and scale change in a test image. It will therefore be a good companion to the Morphological Shared Weight Neural Network (MSNN) developed by Won and Gader (1995; 1997).

The MSNN has been well-tested and its strengths are known. Being a neural net, it can memorize an object and find it in another image with great accuracy. It can even handle large amounts of occlusion of an object. But, it will only recognize what it has been trained on and will not do well on unpredictable changes to scale and rotation of a trained object. It can therefore be concluded that when the MSNN finds something in an image, it is very likely a proper detection. The problems with the MSNN are that it needs several images for training and a scan of an entire image is relatively slow.

As a tradeoff of the shortcomings of the MSNN, the SIFT algorithm is very quick and can be trained on a single image. As described above, the SIFT algorithm is rotation and scale invariant, going as far as finding a trained object oriented upside down in a test image. The algorithm operates by finding keypoints in a training image that are unique in scale space. The original image is progressively Gaussian blurred with a sigma ranging from 1 to 2. Each image is subtracted from its neighbor to produce a series of Difference of Gaussian images. Then, each pixel in each image is compared to its 8 neighbors and the 9 pixels in each of the two adjacent Difference of Gaussian images. If the pixel is smaller than or greater than all of its neighboring pixels, it is an extrema and is therefore a distinct keypoint in scale space. The orientation of the keypoints are then determined from the Gaussian images and stored in a database. Each image will contain hundreds of keypoints and a single database can contain keypoints from many different images. With as many as 100,000 keypoints, the algorithm can recognize keypoints with 70 percent accuracy in a few seconds.



(a) **(b)**

Fig. 3. (a) MSNN: target point is shown on the cup. (b) SIFT: matching feature points are shown for the cup.

At present, the MSNN is a working program with the ability for future modifications. The SIFT algorithm has been implemented to the point of finding keypoints with part of the orientation assignment also completed. Once completed, the SIFT implementation will be integrated into the system. Figure 3 shows examples of recognition using each algorithm.

Interactive Adaptive Vision System

We are also working on an interactive approach to gather, analyze, and present processed visual information to the user in the way of a dialog. Such an interaction enables the user to identify which landmarks the robot can reliably recognize in the environment, and then automatically design and implement detectors for these objects. The user and the robot working together in a dialog will explore the visual features that provide the best performance in object recognition. Since the features have semantic meaning, both the user and the robot have the same shared semantic context, and as a result, the user has an understanding of how the robot is identifying the objects. This knowledge can be used to refine the recognition methods, or may lead the user to decide that the current feature set is inadequate and more features must be added to the system. Such a system has significant advantages during the development process as well as supporting the training of the robot in the field as conditions change, which, in a real situation, is inevitable.

A description of how such a system might work is given as follows. First, the user teaches the robot about objects the user wants the robot to recognize, by showing the robot the various objects in a variety of images collected by the robot. The robot will examine these objects, compute visual features from them, and try to find discriminating features for identifying the objects. It is important that the chosen features used by the system be such that they have a semantic meaning that can be expressed in human language. Choosing features in this fashion provides a shared semantic context for subsequent dialog between the human and the robot. Currently, we are measuring a generic set of features that can be described as 250 different colors, two texture measures (roughness), one symmetry measure, and the presence of straight lines at 180 different angles. This yields 433 different visual features, each describable in clear language terms.

The training set is examined using decision trees, e.g., the C4.5 algorithm (Quinlan 1993). The result is a tree structure with the branches of the tree being created by different values of the features. Since the features have semantic meaning, traversing the tree produces a description of the features important for identifying different objects. We have formulated these descriptions in the form of fuzzy IF-THEN rules. The features producing the branches near the top of the tree have the greatest discriminatory power, and thus are expected to be appropriate for attention purposes. That is, we expect it will provide us with a small set of features that will be useful in guiding our attention when searching for certain kinds of objects. An iterative method of identifying the most powerful features, removing them from the training

data and then looking for the next most powerful features, has been developed and the results are encouraging.

Initial experiments on natural objects such as trees have also been very encouraging. Detectors for finding trees were automatically designed by the system and tested on images containing trees. The results are shown in Figure 4.

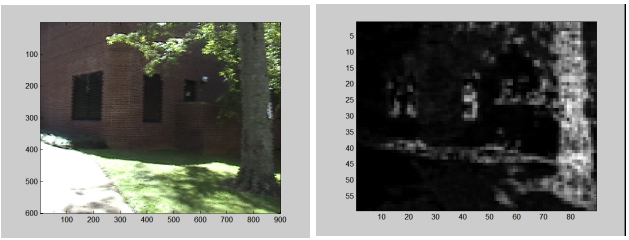


Fig. 4. Results from automatically designed detectors for trees.

Planned Experiments

The adaptive working memory will be tested using a series of evaluation tasks on both the humanoid and mobile robot systems. The first two experiments will be robotic analogs to some of the laboratory experiments that have been performed with monkeys; they provide a validation of basic functionality of the working memory component. After observing success in these simple tasks, we will test whether the system can generalize to more complex situations and more realistic robotic tasks. Results with the working memory component will be compared to results in a current baseline system of each robot (without working memory).

Task 1: Using ISAC, place a green block on the table. ISAC focuses on the green block. Put a red block on the table to serve as a distracting event. Remove the red block; wait a while, then remove the green block. ISAC should then saccade to the previous location of the red block. This is analogous to delayed saccade experiments.

Task 2: Three blocks of different colors are placed on the table. ISAC learns to identify all three blocks by color. A block is then covered and ISAC must reason using the WMS to locate the covered block.

Task 3: Test the mobile robot in a navigation task using landmarks. The robot uses working memory to maintain focus on the few relevant objects for the task. It should ignore other distracting objects, colors, people moving around, etc.

Task 4: Exercise the PFC model while maintaining a controlled environment. ISAC can sense and attend to several different visual events (e.g., motion, color, flashing light, strong edges) but its task will only require attention to a subset of these. The system must learn through experience which events and sensations to attend to.

Task 5: Define a class of "attention-grabbing events" which include such things as the presentation of a verbal instruction, identification of a pointing finger, the presentation of reward, the occlusion of a tracked object, and visual target acquisition. These events are encoded as an unspecified list of sensory features, potentially including a location in space. When an "attention-grabbing event" occurs, the working memory system assesses it for retention. Over time, it learns to determine if the event should be stored, and it learns what features of the event to store, as some may be irrelevant. It also re-evaluates each chunk currently held in memory. (Thus, a goal-attainment event, signaled by reward, can cause memory to be flushed.)

Task 6: Test the mobile robot in a set of experiments with increasing levels of difficulty: (1) Find a known object (e.g., the cup) in a cluttered environment and move to it. Other obstacles, including dynamic obstacles, may present occlusions and distractions while the robot attempts to perform the task. (2) Find an object that looks like "this" (the robot is shown an example object) and move to it (with occlusions and distractions). (3) Find the object to the right of the "cup" and move to it (with occlusions, etc.). (4) Find the object (any of the above), move to it and then come back (again, with occlusions and distractions). Metrics include success or failure within an allotted time period, distance traveled, and time to perform the various parts of the tasks (time spent scanning for the target object, time to reach the object, time to return, etc.).

Concluding Remarks

In this paper, we have discussed our initial efforts towards integrating an adaptable working memory into robotic systems that have the ability to interactive physically with their environment. Using a model of working memory based on temporal difference learning, as inspired by evidence in cognitive neuroscience, we are implementing a working memory structure for robotic systems. The utility of the working memory model will be tested in a set of experiments, some of which are modeled after experiments in cognitive science. If effective, the working memory system has the potential to significantly improve the intelligence of robots. At the same time, the work will also test how well the existing cognitive models scale up in realistic settings using the robotic systems and, thus, has potential to contribute to research in computational neuroscience.

Acknowledgements

This work is supported by NSF under grant EIA-0325641. Contributing students include Sam Blisard and Bob Luke at UM-C and Will Dodd, Palis Ratanaswasd, and Josh Phillips at Vanderbilt.

References

Albus, J.S. 1991. Outline for a theory of intelligence. *IEEE Trans. SMC*, 21(3): 473-509.

Baddeley, A. 1986. *Working Memory*, 11. Oxford Psychology Series. Oxford: Clarendon Press.

Braver, T. S. & Cohen, J. D. 2000. On the control of control: The role of dopamine in regulating prefrontal function and working memory. In S. Monsell & J. Driver, eds., *Control of Cognitive Processes: Attention and Performance XVIII*, 713-738. MIT Press.

Cowan, N. 2001. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, 24(1): 87–185.

Demb, J.B., J.E. Desmond, A.D. Wagner, C.J. Vaidya, G.H. Glover and J.D.E. Gabrieli. 1995. Semantic encoding and retrieval in the left inferior prefrontal cortex: A functional MRI study of task difficulty and process specificity. *J. of Neuroscience*, 15: 5870–5878.

Erol, D., Park, J., Turkay, E., Kawamura, K., Jenkins, O.C. and Mataric, M.J. 2003. Motion Generation for Humanoid Robots with Automatically Derived Behaviors, *Proc. IEEE Intl. Conf. SMC,* Washington, DC, 1816-1821.

Goldman-Rakic, P.S. 1987. Circuitry of the prefrontal cortex and the regulation of behavior by representational knowledge. In F. Plum and V. Mountcastle, eds, *Handbook of Physiology*, 373–417, Bethesda, MD: American Physiological Society.

Grabowski, J. 1999. A Uniform Anthropomorphological Approach to the Human Conception of Dimensional Relations. In *Spatial Cognition and Computation*, 1: 349-363.

James, W. 1890. *The Principles of Psychology*. New York: Holt, Rinehart and Winston.

Kawamura, K., Peters II, R.A., Bodenheimer, R., Sarkar, N., Park, J. and Spratley, A. 2004. Multiagent-based Cognitive Robot Architecture and its Realization. *Intl. J. of Humanoid Robotics*, 1:1.

Lowe, D. 2004. Distinct Image Features from Scale-Invariant Keypoints. In *Intl. J. of Computer Vision*.

Matsakis, P. and Wendling, L. 1999. A New Way to Represent the Relative Position of Areal Objects. In *IEEE Pattern Analysis and Machine Intelligence*. 21(7): 634-643.

Matsakis, P., Keller, J., Wendling, L., Marjamaa, J. and Sjahputera, O. 2001. Linguistic Description of Relative Positions of Objects in Images. In *IEEE Trans. SMC*, 31(4): 573-588.

Miller, E.K. 2003. Cognitive Control: Understanding the brain's executive, in Fundamentals of the Brain and Mind, Lecture 8, June 11-13, MIT.

Miller, G.A. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63: 81–97.

Miyake, A. and P. Shah, eds. 1999. *Models of Working Memory: Mechanisms of active maintenance and executive control*, Cambridge: Cambridge Univ. Press.

Moratz, R., Fischer, K. and Tenbrink, T. 2001. Cognitive Modeling of Spatial Reference for Human-Robot Interaction. In *Intl. J. on Artificial Intelligence Tools*, 10(4): 589-611.

Norman, D.A. and Shallice, T. 1986. Attention to action: Willed and automatic control of behavior. in R.J. Davidson, G.E. Schwartz and D. Shapiro, eds, *Consciousness and Self-Regulation: Advances in research and theory*, 4: 1–18. New York: Plenum Press.

O'Reilly, R. C., Noelle, D. C., Braver, T. S., & Cohen, J. D. 2002. Prefrontal cortex and dynamic categorization tasks: Representational organization and neuromodulatory control. *Cerebral Cortex*, 12:246-257.

Peters II, R.A., Hambuchen, K., Kawamura, K. and Wilkes, D.M. 2001. The Sensory Ego-Sphere as a Short-Term Memory for Humanoids. In *Proc. IEEE-RAS Intl. Conf. on Humanoid Robots*.

Perzanowski, D., Schultz, A.C., Adams, W., Marsh, E., Bugajska, M. 2001. Building a Multimodal Human-Robot Interface. *IEEE Intelligent Systems*, 16-20.

Quinlan, J.R. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

Rainer, G., Asaad, W.F. and Miller, E.K. 1998. Selective representation of relevant information by neurons in the primate prefrontal cortex. *Nature*, 393: 577–579.

Skubic, M., Perzanowski, D., Blisard, S., Schultz, A., Adams, W., Bugajska, M. and Brock, D. 2004. Spatial Language for Human-Robot Dialogs. In *IEEE Trans. SMC, Part C*. 34(2):154-167.

Skubic, M., Matsakis, P., Chronis, G. and Keller, J. 2003. Generating Multi-Level Linguistic Spatial Descriptions from Range Sensor Readings Using the Histogram of Forces. In *Autonomous Robots*, 14(1): 51-69.

Squire, L.R. 1987. *Memory and Brain*. New York: Oxford University Press.

Tulving, E. 1972. Episodic and semantic memory. in E. Tulving and W. Donaldson, eds., *Organization of Memory*, 381-403. New York: Academic Press.

Tversky, B., Lee, P and Mainwaring, S. 1999. Why Do Speakers Mix Perspective? In *Spatial Cognition and Computation*, 1: 399-412.

Warrington, E.K. and Weiskrantz. L. 1968. New methods of testing long-term retention with special reference to amnesic patients, *Nature*, 217: 972–974.

Waugh, N.C. and D.A. Norman. 1965. Primary memory. *Psychological Review*, 72: 89–104.

Wolpert, D. M. and M. Kawato. 1998. Multiple paired forward and inverse models for motor control. *Neural Networks*, 11: 1317-1329.

Won, Y. 1995. *Nonlinear Correlation Filter and Morphology Neural Networks for Image Pattern and Automatic Target Recognition*, Ph. D. Dissertation, Univ. of Missouri – Columbia.

Won, Y., Gader, P. and Coffield, P. 1997. Morphological Shared-Weight Neural Networks with Applications to Automatic Target Recognition. *IEEE Trans. Neural Networks*, 8: 1195-1203.