Tracing Patterns and Attention: Humanoid Robot Cognition

Luiz-Marcos Garcia, Laboratory for Analysis and Architecture of Systems
Antonio A.F. Oliveira, Federal University of Rio de Janeiro
Roderic A. Grupen, David S. Wheeler, and Andy H. Fagg, University of Massachusetts, Amherst

HUMANOID ROBOTS PROMISE TO LEAD US TOWARD MORE EFFECTIVE AND INFORMATIVE INTERACTIONS BETWEEN HUMANS AND ROBOTIC DEVICES. THE AUTHORS INTRODUCE MECHANISMS FOR ATTENTION CONTROL AND PATTERN CATEGORIZATION AS THE BASIS FOR COGNITION IN A HUMANOID ROBOT.

DO ANDROIDS DREAM OF ELECTRIC sheep? Science fiction books, movies, and TV shows have long described humanoid robots interacting with—even passing for—real humans. But how close are we today to that sort of seamlessness between robots and humans?

In exploring that question, we have developed a methodology to provide the basic mechanisms that support humanoid robot cognition. Involved are mechanisms for attention control and pattern categorization (see the "Attention" sidebar), and constructing attentional maps of the environmental context that underlies behavior. Our system architecture supports several types of sensors, and we have experimented with many facets of the eventual integrated platform for learning control, acquired representations, and visual behavior.[1–4]

Our goal is to develop a robotic system capable of interacting with its environment and eventually with humans. We're also interested in the relationships between visual, haptic (touch-based), proprioceptive (simuli arising from within), and motor systems in humans. To this end, we constructed a robot, *Magilla*, consisting of a stereo head and two robotic arms (with attached graspers). To explore the potential for rich and varied interactions with the world, we connect our robot's cognitive organization and development to that of a human's—a methodology missing in the approaches the AI community usually employs. Moreover, by grounding our systems in the natural world (that humans share) and by providing flexible and redundant means of interacting within this domain, we postulate that our robot will develop cognitive structures more like those of humans and, as a consequence, will interact with humans more constructively.

Learning from children

Again, one of our main objectives in constructing an anthropomorphic torso is to study the relationship between visual and haptic sensory systems and their development in humans. The development of robot programs is an incremental search for strategies that exploit the intrinsic dynamics of the robot–world interaction. We interpret *intrinsic dynamics* fairly broadly as any kinematic, dynamic, perceptual, or motor synergy that produces characteristic and invariant temporal sequences of observable state variables.

Humanoid robots are simply too complex to use traditional approaches from robotics and computer vision. The range of interaction possible and the required kinds of perceptual distinction necessary challenge commonly used methodologies for control and programming. Consequently, we have adopted an incremental and automatic approach to programming, modeled after the sensorimotor development of human children in the first two years of life. Genetically mediated maturational mecha-

1094-7167/00/$10.00 © 2000 IEEE

Attention

In a general sense, we can define *attention* as the ability to select a topic of interest (a goal) for extracting information useful for a given task. A challenging problem in attentional control is the construction of an efficient mechanism with which to change attentional focus. Many facets of attention (top-down × bottom-up, covert × overt, and so on) are included in the task specification and influence the strategy adopted for changing attention. In our work context, the interest topics are real objects (or sometimes regions of interest) in a restricted domain. We employ attentional control and pattern categorization to provide interaction between the robot and the environment. We consider mainly using bottom-up and overt attention in a variety of monitoring tasks. The robot has to learn the characteristics of objects present in its environment, inserting a representation for each different type in its memory and dealing with new or known categories. Also, the robot has to learn how to construct the attentional maps. After it constructs such a world representation, the robotic agent can perform other, more specific tasks.

In relation to the work discussed in the "Background and related work" sidebar, we provide a more complete *working* model for robot cognition, including attention and pattern categorization behaviors. We consider an improved, practical set of features for both behaviors, extracted from real-time sequences of stereo images. This includes static spatial properties (such as intensity and texture), other temporal properties (such as motion), and stereo disparity features. Also, as the system is currently implemented in a robot platform that operates in a real environment, we include the robot's current functional state in the feature space.

nisms focus the infant on simple problems first and subsequently enrich these policies by including additional motor and perceptual systems.[5] Infants are constantly learning about the capabilities of their motor systems and adapting motor strategies in accord with their current level of sensory and motor control[6]—naive sensorimotor programs are not burdened with the full complexity of infant neuroanatomy. Instead, maturational mechanisms in the brain, co-contraction of distal degrees of freedom (DOF), and developmental neurological structure organize and direct evolving motor programming. Attentional mechanisms participate in this growth process—it is therefore critical to develop flexible means of directing attention in humanoid robots that can vary as a function of time. The methodology reported in this article is a first step in that direction.

In addition to basic contributions to creating computational systems and models of human growth and development, we expect to advance an important class of robot applications as well. Teleoperation of remote motor systems is taxing physically and mentally (due to technological and developmental limitations) and suffers from intrinsic temporal latencies. Robots must become capable of responding autonomously—this ultimately requires the kind of motor and sensory flexibility that humanoid-scale systems provide. Moreover, robots have traditionally been designed to complement humans in manufacturing tasks but this leads to special-purpose mechanisms and dedicated control. To cooperate in general tasks in space, terrestrial applications, and under water, a common and flexible configuration of resources that can adapt to many tasks in the context of human collaborators is attractive. Humanoid robots fit this general specification.

Magilla—the humanoid robot

One aspect of Magilla's behavior involves the articulation of its stereo head to maintain an internally consistent spatial model of its

Background and related work

Many researchers have tried to reproduce or imitate biological systems and behaviors. In this direction, Pierre Van der Laar and his colleagues[1] and Laurent Itti and his colleagues[2] describe interesting computational models for the neurophysiology of attention. Both approaches perform multifeature extraction on stationary, monocular images to compute several feature maps. Then, mechanisms expressed in the form of function approximators gather information from the feature maps to construct a salience map that governs attention. Ilya Rybak and his colleagues[3] relate attention and identification using a simple model with stationary monocular images. Perception and cognition are treated as behavioral processes in which an attention window (with multiple image fragments processed in parallel) scans images sequentially. The two well-known "what" and "where" pathways are encoded using a neural network. Steve Kosslyn's work,[4] based on neuropsychology and neurobiology results, also suggests a good descriptive model to explain how identification and recognition happen. In his model, Kosslyn suggests combining features extracted from visual images with mental image completion for recognition purposes. Although Kosslyn suggests few practical mechanisms, the overall approach is intuitively appealing. Rajesh Rao and Dana Ballard[5] provide a set of operators based on Gaussian partial derivatives for feature extraction. Biological models motivate these operators.

Most of this related work considers using only stationary, monocular image frames, not temporal aspects such as motion or functional and behavioral aspects. These approaches also don't provide real-time feedback of environmental stimuli. Our work's main contribution lies in our approach to attentional control by computing features from an image filtered with Gaussian derivative operators combined with motion and stereo to generate salience maps. The next region of interest is simply given by the most salient region in those maps. For pattern categorization, we use appearance-based (semi-invariant) features abstracted from Gaussian operated images plus stereo and motion patterns as input to an associative memory that remembers addresses into a pattern-storage memory.

References

1. P. Van de Laar, T. Heskes, and C. Gielen, "Task-Dependent Learning of Attention," *Neural Networks*, Vol. 10, No. 6, Aug. 1997, pp. 981–992.
2. L. Itti et al., "A Model of Early Visual Processing," *Advances in Neural Information Processing Systems*, MIT Press, Cambridge, Mass., 1998, pp. 173–179.
3. I.A. Rybak et al., "A Model of Attention-Guided Visual Perception and Recognition," *Vision Research*, Vol. 38, Nos. 15–16, Aug. 1998, pp. 387–400.
4. S.M. Kosslyn, *Image and Brain: The Resolution of the Imagery Debate*, MIT Press, Cambridge, Mass., 1994.
5. R.P.N. Rao and D. Ballard, "An Active Vision Architecture Based on Iconic Representations," *Artificial Intelligence Magazine*, Vol. 78, Nos. 1–2, Oct. 1995, pp. 461–505.

Figure 1. Magilla's current configuration. Barrett Technologies designed and built the arms and hands.

neighboring visual landscape. This approach applies to a variety of visual monitoring tasks, including attentional control and pattern categorization for constructing attentional maps of the environment. To perform this bottom-up attentional task, our robot must be able to move its eyes to regions of interest (including foveation, vergence, and accommodation or focus), maintain attention on the region as needed, and shift its attentional focus when the current region is no longer of interest. The system maintains attentional maps of the environment (containing information about pattern representation, position, and orientation) and updates them incrementally, as needed. Initially, the system knows nothing about the environment (the attentional maps are empty), nor about the objects (the system memory has no pattern representations). It has to learn the characteristics of new objects detected in the environment. To do so, an active behavioral strategy dynamically interacts with a changing environment and discovers perceptual events that are worthy of attention. (Other tasks that the system can currently perform include visual surveillance, inspection, spatial orientation, and navigation. Future applications include tasks involving object manipulation by the two arms that can act based on visual and haptic information.)

The articulated stereo head provides Magilla's vision. A dedicated image-processing device provides data reduction and feature abstraction from the visual input data. Based on this perceptual buffer's features and on the current robot pose and functional state, the system can define its perceptual state. Magilla can make control decisions based on the information contained in the perceptual state, selecting the right actions in response to environmental stimuli. This approach is inherently reactive—choosing actions based on perceptions of the world at different temporal scales rather than by using a geometric model as traditional planning techniques do. Also, by reducing and abstracting data, the system performs fewer computations and substantially improves its performance. Finally, the attentional mechanism developed here can deal with dynamic environments and provide real-time feedback to stimuli of varying bandwidth. For these reasons, this system architecture is particularly relevant to humanoid robot systems.

Briefly describing the operation cycle, the system constructs salience maps bottom-up from attentional feature maps to direct attention to regions of interest. Magilla uses Saccadic movements—jerky eye movements from one point to another—(possibly including pan or tilt movements) to foveate on the selected region. The system then performs feature extraction, producing changes in the perceptual state. An associative memory maps the features into a pattern address, allowing recognition of existing perceptual categories or discovery of new categories. To complete its operating architecture, the system inserts information about the new or known pattern representations detected in the environment in the attentional maps. After the system visits all the regions, it initiates a guided search to detect any change that might occur in the environment. The heart of the system is the attentional mechanism, which controls a change in the attention focus. The attention mechanism is designed to visit all regions in the visible environment—it eventually returns to regions previously visited. The mechanisms that we describe here are a first step toward turning a reactive visual index into a richly associative and multimodal perceptual memory.

Figure 1 shows Magilla's final configuration. It consists of two whole arm manipulators (WAMs), two multifingered hands, and a BiSight stereo head on top of its torso.

Each arm (Figure 2a) is a highly dexterous, back-drivable manipulator. It includes seven DOF, and its kinematics are similar to humans. A cabled transmission system provides motion for each joint—the cable drives permit low friction and smooth torque transmission from the actuator to the joints. Smooth and precise motions come from the motors and reducers placed at the joints, which maximize transmission rigidity and minimize cable-loading charges. This allows a large range of motion (with an average of 270 degrees, with zero backlash and a near-zero friction coefficient). The WAM can manipulate large and heavy objects as well as smaller objects with an attached gripper. These features provide a good way to simulate the human muscular system's characteristics.

Magilla's hands (Figure 2b) are two BH8-255 Barrett Hands, which are multifingered graspers. Each hand has three multijointed fingers and four independent DOF. Two of the fingers have an extra DOF with 180 degrees of synchronous lateral mobility, supporting different grasp types. The hand can grasp and hold target objects of different sizes, shapes, and orientations. Also, the fingertips have tactile force–torque sensors (ATI Nano17) to recover tactile information.

Figure 3 schematically shows the head's DOF. This platform consists of two video cameras mounted on a TRC BiSight head providing four mechanical DOF—pan, tilt, and independent left and right vergence. Zoom and focus, not used in this work, are also controllable by using the same motion

Figure 2. Magilla's (a) arm and (b) hand. Photographs courtesy of Barrett Technologies.

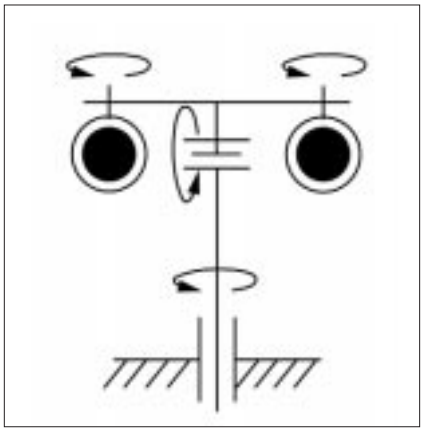

Figure 3. A schematic of the mechanical degrees of freedom in Magilla's head.

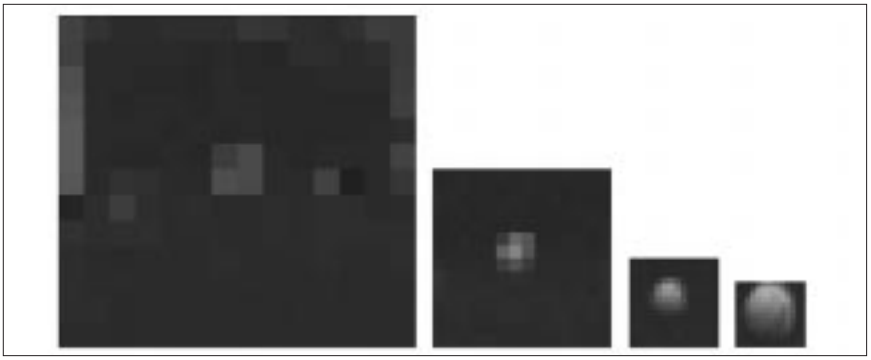

Figure 4. A multiresolution image of a sphere.

interface (PMAC from Delta TAU). Images from each camera input into a dedicated pipelined-array *image processor* (from Datacube). We use this IP device—which can perform image-processing operations in real time (up to 30 frames per second)—to reduce and abstract data. We did the work reported here entirely on the stereo head platform.

Data reduction and feature abstraction

We perform data reduction and feature abstraction inside the Datacube IP device. To accomplish data reduction, we compute a multiresolution (MR) representation in four levels, as in Figure 4. Each image is composed of 64 x 60 pixels. We compute two MR images for each eye (camera) to further extract intensity-based features and motion features. For MR intensity-image generation, we use the original 512 x 480 pixel stereo images as input. For MR motion images, we first compute the difference between two consecutive image frames ($I_t - I_{t-1}$). Then, we compute successive resolution levels for both the intensity and motion images by applying a mean filter in the neighborhood of each pixel and sampling with a level-dependent factor. The size of the neighborhood and the input image's affected region are also level-dependent. The result is MR intensity and motion images for each eye.

A *multifeature* representation provides abstraction to support attention and categorization behaviors. To obtain this MF representation, we apply Gaussian partial-derivative kernels, in two orthogonal orientations each, to the previous MR intensity and motion representations. An ideal approach for motion-feature generation effectively computes the motion field for the images. However, in our case, motion computation is unnecessary for attentional purposes and expensive for categorization purposes. We also perform a four-pixel sampling here, reducing the images to 16 x 15 pixels. So, the result is a *multiresolution–multifeature* (MRMF) representation, as in Figure 5. The first six column images are adapted Gaussian partial derivatives (order 0, 1, and 2) of gray-level images (referred to as Gaussian features) and the last two are derivatives of consecutive frames difference, our representation for motion. Considering the total number of pixels, original data decreases by a factor of 32. In the current implementation, the Datacube IP device generates the 64 feature images (32 for each eye) at 15 frames per second. This achieves reasonable performance for the real-time processing necessary in Magilla's active vision system.

Additional behavior control

Our system's desired attentional behavior is to focus attention on its environment's currently most salient region. This involves computing a salience map for each eye, taking the winning region and generating saccadic eye (and possibly pan or tilt) movements to foveate on that region.

Pre-attention and target definition. We compute salience maps with the data abstracted from the Datacube (MRMF), by interrogating the attentional maps and also by considering other task-dependent attentional features. Similar to the MRMF images, the salience maps have a pyramidal structure. The attentional feature matrices that we consider for salience-map generation are stereo disparity D_{ij}, Gaussian intensities $I_{ij}^{(0)}$, $I_{ij}^{(1)}$, $I_{ij}^{(2)}$, motion intensities M_{ij}, proximity P_{ij}, mapping T_{ij}, and interest E_{ij}. Gaussian and motion-intensity matrices are the square of magnitude of each image pair in Figure 5. The matrix T_{ij} tells whether a region was previously visited. In the matrix P_{ij}, each position's value is inversely proportional to its distance to the fovea. The value of each E_{ij} is set to zero when a region receives attention and increases slowly over time.

We compute stereo disparity by using a simple cascade correlation approach (see Figure 6) on the second-order Gaussian intensity $I_{ij}^{(2)}$ computed earlier. Because we have an MR representation, we use the results from one level to predict disparity for the next. For the initial level, the range of disparity is limited by vergence movement constraints. It is also constrained by the relative symmetry of the images with respect to the system's cyclopean axis. Also, disparity is computed only for the *x* direction because our system produces no *y* disparity. The cascade process sub-

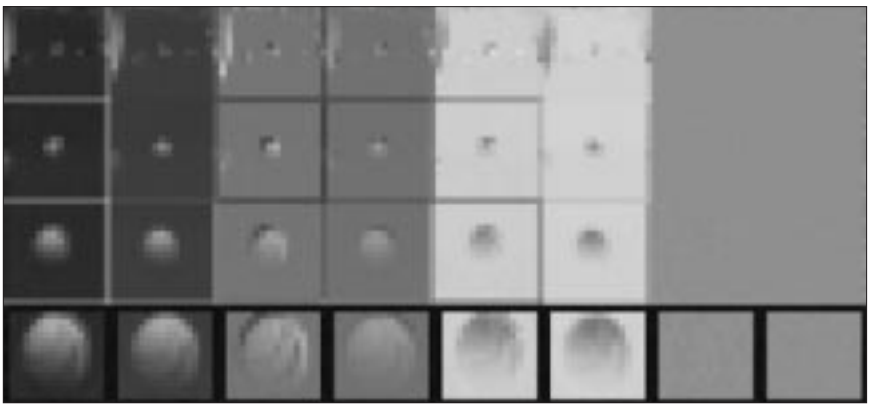

Figure 5. The multiresolution–multifeature matrices.

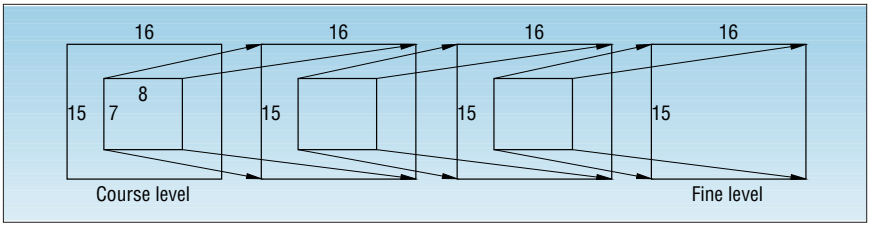

Figure 6. The cascade process used for stereo. (The numbers represent pixels.)

stantially reduces the computations necessary to find the best match for a point. The system computes the stereo process for both eyes, and it requires less than 0.1 sec.

After all attentional feature maps are computed, starting from the coarsest level, the system calculates an activation value for each position in the salience maps as a simple weighted summation of all attentional features:

$$S_{ij} = w_D D_{ij} + w_M M_{ij} + w_I^{(0)} I_{ij}^{(0)} + w_I^{(1)} I_{ij}^{(1)} + w_I^{(2)} I_{ij}^{(2)} + w_E E_{ij} + w_P P_{ij} + w_T T_{ij}.$$

The system can learn this function's task-dependent weights (w_0) using a neural network approach[7] or reinforcement learning.[8] In our monitoring task, we determined them experimentally. This simple function makes the system move its attention window from region to region, covering the whole scene but eventually returning to previously visited regions to detect possible changes. The result is a monitoring behavior in which the robot maintains a representation of the world (the attentional maps) consistent with reality. The robot's attention mechanism never visits a region twice in a row, but, depending on the weight function used to update the world-map activation values, it might revisit a region before visiting all other regions. Because we have a salience map for each eye, the winning region also determines the dominant eye (the eye whose salience map contains the most active region).

Shifting attention (saccades and vergence). Attention shifting involves taking the most active region over all levels in the salience maps and computing a coarse saccade movement to foveate each eye on the target. We use features near the target in the dominant eye to build a model to aid in later fine saccadic corrections. For the dominant eye, we determine the target position by taking the displacement from the current position to the winning one. For the nondominant eye, we compute the target by adding the stereo disparity to the dominant eye's target position. From these eye displacements, we compute the displacements for the stereo head's four DOF according to several constraints. If the cyclopean angle exceeds 15 degrees, Magilla initiates a pan movement to reduce the angle. If the eye axes diverge at more than parallel, or if they converge at more than 45 degrees, Magilla applies a correction to the nondominant eye vergence. We compute tilt motion directly from the dominant eye's vertical image displacement; the stereo head's geometry ensures both eyes have the same tilt. To complete this first phase of attentional process, the PMAC interface effectively produces physical movements—running its positional derivative (PD) servo controllers concurrently—to put the attentional goal in the fovea region.

After a coarse saccade, due to several types of error the goal might not be in the fovea. To correct that, the robot performs fine saccades iteratively at increasing levels of resolution to maximize the correlation between the target model and the dominant eye's image center. This process converges when the resolution level that determined the attention shift reaches a maximum correlation at (or very near) the image center. Simultaneously, the vergence algorithm, also iteratively, calculates the nondominant eye's displacements to maximize the correlation at the center of the two eye images. We use a threshold to avoid situations where there is no match inside the field of view due to occlusions. As a result of both iterative processes, the system will have both cameras foveated in the goal.

Categorization behavior

After the robot has its attention in focus, the system computes other feature set from the Datacube output (MRMF) for categorization behavior. Experiments using the MRMF representation directly as features for associative memory lookup gave good results, but are computationally expensive. In those experiments, we used a total of 3,840 features, therefore we used some abstraction to further reduce the input data. Our current approach uses locally normalized mean $\mu_{i,j}$ and variance $\sigma_{i,j}^2$ in the vicinity of only four positions—$(i,j) = (3,3),(3,8),(8,3),(8,8)$—for each one of a given level's 16 x 15 images. We compute the above statistical moments for each one of the six Gaussian images in Figure 5 and for the magnitudes of motion and stereo disparity computed in the attentional phase, which reduces the feature set to a total of 112 input features for each eye. Also, as a result of the averaging and variance, feature matching is more tolerant of scaling, rotation, and shift—experiments indicate rotations of up to 30 degrees are acceptable. Identification entails using an associative memory implemented by a multilayer perceptron trained with a back-propagation algorithm (BPNN).[9] The BPNN maps the features to an address in a long-term memory (LTM), which stores other information. We could use other classifiers, such as Teuvo Kohonen's self-organizing maps[10] or Paul Viola's belief networks,[11] but the BP network approach used here gives good results on identification and also returns the activation for a given index in the output layer (a normalized value between 0 and 1). This value can determine if a representation is new, and we can use it in top-down attentional tasks to keep attention in a given region.

Once the algorithm has classified a representation as new or identified, the system updates the current position in the attentional maps. It stores attentional features sufficient to detect online changes and updates mapping (T_{ij}) and interest (E_{ij}) feature values, allowing a shift of attention to another region. If the representation is new, the system automatically invokes supervised learning, inserting the new feature set into LTM and updating (creating new nodes in the hidden and output layers) and retraining the BPNN. (We can retrieve information in the LTM by using features from any resolution level.) In general, the resolution level depends on the task, available time, and image characteristics, and the attentional mechanism determines it.

Demonstrations and results

We performed several demonstrations involving attention and categorization behaviors for constructing attentional maps in a monitoring task. In the final experiments, we placed many instances of objects of various types on a table. We expect the robot to focus attention on all objects, learn their characteristics (inserting a representation for each in the BPNN), and incrementally update the attentional maps. The net result is a prioritized search (or monitoring behavior) of the task space.

Attentional behavior. In attentional behavior, we constructed three types of demonstrations. In the first type, we indicated the objects to the robot by touching or pointing to them

Figure 7. Experimenting with Magilla, using (a) motion cues and (b) attention without motion cues.

in a sequence (Figure 7a). Magilla used this motion cue (motion wins in the attentional process) to foveate on the objects. The stereo head went after the motion cues, detecting all objects on the table.

In the second demonstration, Magilla relied solely on intensity cues to visit all regions, not motion cues (Figure 7b). The attentional mechanism causes the robot to visit all regions, using mainly mapping T_{ij} and interest E_{ij} terms.

In the third demonstration, after we inserted all objects in the attentional maps, we either moved or removed an object. Magilla updated the attentional maps for the changed regions using mainly motion and intensity cues for changes (or movements) inside the field of view. If a change occurred outside the field of view, Magilla eventually returned attentional focus to that region by using T_{ij} and E_{ij} terms. Then, it detected the eventual change by comparing current attentional features with stored ones. Figure 8 shows a sequence where Magilla tracks an object by virtue of differences observed in the working attentional maps. The robot updates the maps continuously until the ball stops in the final position.

Categorization behavior. In the experiments involving categorization behavior, the system (without any prior knowledge) learned the objects' characteristics, inserting a representation for each new object type in its associative memory. A threshold determined whether a representation was new. In the experiments, the system could detect all new objects and rec-

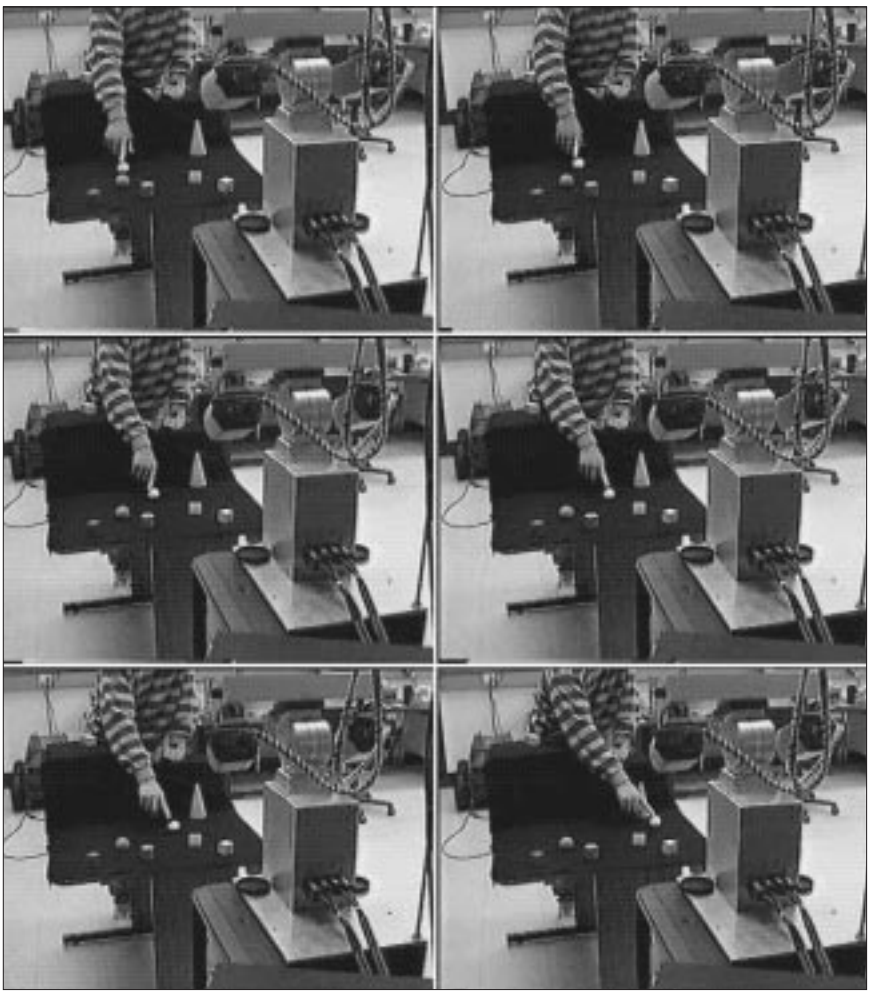

Figure 8. Updating attentional maps (tracking).

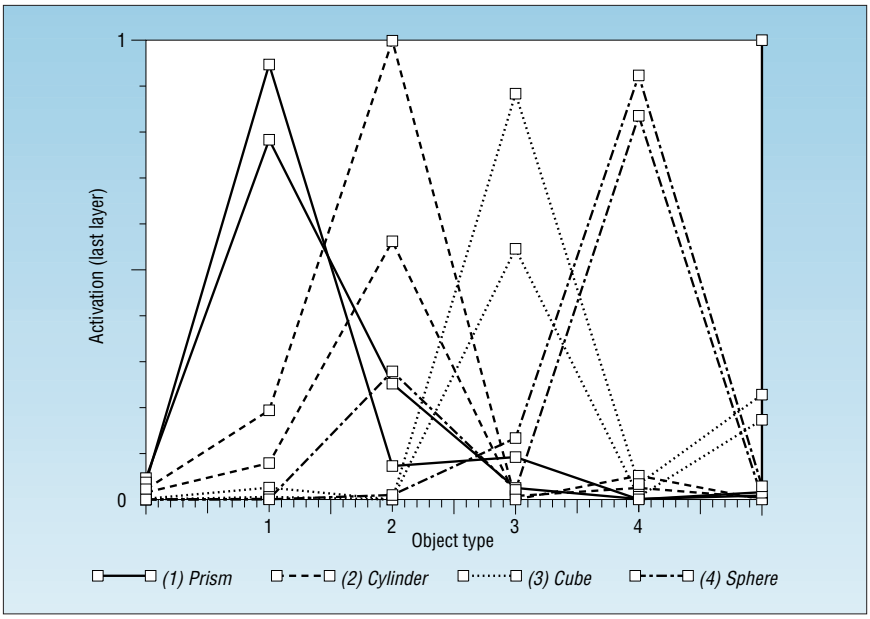

Figure 9. Activations for objects in BPNN.

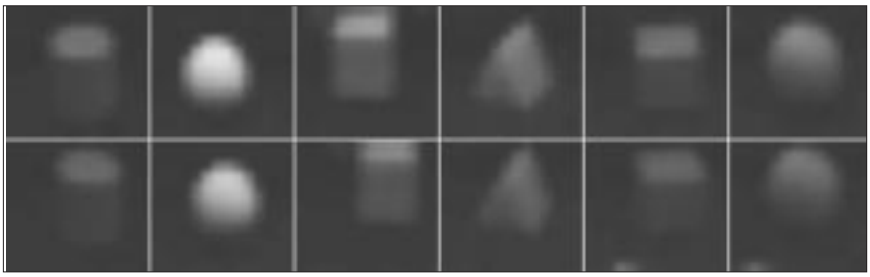

Figure 10. The different types of objects detected.

ognize all instances of already known objects.

Figure 9 shows BPNN confidence (simultaneous activations in its last layer), using several instances of four types of objects, placed on the table with controlled poses. We included variations in the object poses with rotations up to 30 degrees when viewed from the stereo head. For each instance, the upper line is the highest activation, experimented with the object in the learned pose. The next lower line is the lowest activation, experimented with the object pose degraded. We could verify that the system supported well these variations. For all objects, the activation errors (second lines) are still over the threshold, thus allowing categorization. Figure 10 shows only new object types discovered in one of the experiments. The figure also illustrates the vergence mechanism working correctly, become the same object is in the fovea for each pair of images.

We did several experiments to test the associative memory's (BPNN's) performance. Figure 11 shows two graphs illustrating performance as the number of objects increases for the BPNN training procedure. The graphs show an apparently soft parabolic function, which is a characteristic of the adopted BP model. In practice, this issue does not compromise system performance—a model of a short-term, working memory with 10 objects seems quite acceptable. As Figure 11b shows, the system can learn a reasonable set of objects (more than 20) in about 20 seconds of training time. We can improve this time by applying local training. Here, we also reinitialize the network for each running experiment (it starts with zero objects) to test the automatic supervised learning procedure. In a practical situation, the system would save the synaptic weights and network configuration (also the attentional cues) for future use. In this situation, the probability of finding a new object would be very small—it wouldn't interfere with system performance. Moreover, as an example of further application, we can propagate the acquired knowledge (the learned BPNN weights) into multiple entities (other humanoids). Of course, this somewhat depends on the environment and task domain.

During some experiments, we also collected system performance data. Table 1 shows the time required for each of the processes involved in the attention and categorization behaviors. The first column identifies the process, and the remaining columns show the minimum, maximum, and average times required, sampled over several hundred control cycles. The first phase occurs entirely inside the Datacube. Then, after transfer to the host computer, the pre-attention phase computes all attentional features described earlier. This phase includes computing stereo dispar-

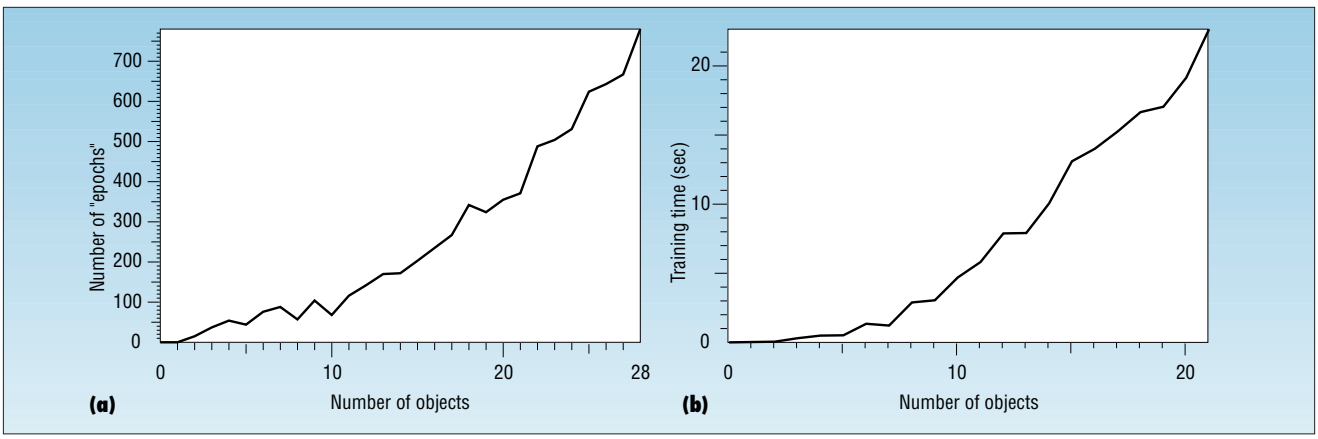

Figure 11. Training performance in (a) number of epochs and (b) time in seconds.

Table 1. Processing time required in each phase.

Phase or process	Min(sec)	Max(sec)	*M*(sec)
Computing MRMF	0.145	0.189	0.166
Pre-attention	0.139	0.205	0.149
Salience maps	0.067	0.134	0.075
Total attention	0.324	0.395	0.334
Total saccade	0.466	0.903	0.485
Features for match	0.135	0.158	0.150
Memory match	0.012	0.028	0.019
Total identification	0.323	0.353	0.333

ity and intensity (square of magnitude) of Gaussian and motion features. The statistical feature calculation for categorization (features for match), also occuring in the host computer, is another expensive phase. We have since replaced the host computer (a 40-MHz Sun Sparc 10) with a 300-MHz Sun Ultra 10 board that shares the Datacube bus for improved data transfer rates. So, we can currently achieve a processing rate of 10 to 15 frames per second (about 70 to 100 milliseconds per object) for both behaviors. We can also improve saccade speed by adjusting PD controller gains in the stereo head PMAC interface, making Magilla's saccade movements as fast as a human's.

ALTHOUGH WE MAINLY USED visual information, our system is more generally applicable and can easily incorporate haptic, auditory, or other sensory maps to provide a more discriminative feature set. We have developed here the basic architecture integrating an attentional mechanism and a neural network classifier. The controller-oriented approach lets us integrate other processes into this architecture. We believe that the two abilities (attention and categorization) are a basis not only for this simple monitoring task but also for other more complex tasks involved in a humanoid robot. The two behaviors are interrelated, and a behaviorally active system needs both to perform other tasks.

The immediate application that comes to mind to test our architecture is a monitoring task. In such a task, the system incrementally constructs and dynamically updates maps of the environment. Other applications—such as automatic interpretation of agent behaviors, studying patterns of activity in dynamic environments, perimeter security, and policing—can use such an approach. A more advanced task that could integrate the arms would be to interact with a human in a collaborative task. We also plan to experiment with reaching and grasping tasks, using visual and haptic information. ■

Acknowledgments

We performed this work at the Laboratory for Perceptual Robotics, University of Massachusetts (www-robotics.cs.umass.edu). The National Science Foundation funded this work under RI-9704530 and CDA-9703217 and DARPA funded it under AFOSR FA9620-97-1-0485. The National Research Council (CNPq) and the Resesarch Support Foundation of the State of Rio de Janeiro (PAPERJ), both from Brazil, also funded and supported this work.

References

1. M. Huber and R.A. Grupen, "Learning to Coordinate Controllers: Reinforcement Learning on a Control Basis," *Proc. 15th Int'l Joint Conf. on Artificial Intelligence*, AAAI Press, Menlo Park, Calif., 1997, pp. 1366–1371.
2. J.A. Coelho, Jr. and R.A. Grupen, "A Control Basis for Learning Multifingered Grasps," *J. Robotics Sys.*, Vol. 14, No. 7, 1997, pp. 545–557.
3. J.H. Piater and R.A. Grupen, "A Framework for Learning Visual Discrimination," *Proc. 12th Int'l FLAIRS Conf.*, AAAI Press, Menlo Park, Calif., 1999, pp. 84–88.
4. L.M.G. Gonçalves, A.A.F. Oliveira, and R.A. Grupen, "Multi-Modal Stereognosis," *Proc. Third Int'l Conf. on Autonomous Agents*, ACM Press, New York, 1999, pp. 337–338.
5. R. Grupen et al., "Structure and Growth: Computational Model of Robot Development," presented at the NSF/DARPA Workshop on Development and Learning, 2000; ftp://www-robotics.cs.umass.edu/pub/papers/DARPA00.ps.gz (current July 2000).
6. N.E. Berthier, "Learning to Reach: A Mathematical Model," *Developmental Psychology*, Vol. 32, No. 5, Sept. 1996, pp. 811–823.
7. P. Van de Laar, T. Heskes, and C. Gielen, "Task-Dependent Learning of Attention," *Neural Networks*, Vol. 10, No. 6, Aug. 1997, pp. 981–992.
8. R.S. Sutton and A.G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, Mass., 1998.
9. M. Riedmiller and H. Braun, "A Direct Adaptive Method for Faster Backpropagation Learning: The Rprop Algorithm," *Proc. Int'l Conf. on Neural Networks*, ICNN '93, IEEE Computer Soc. Press, Los Alamitos, Calif., 1993, pp. 123–134.
10. T. Kohonen, *Self-Organizing Maps*, Springer Verlag, New York, 1997.
11. P.A. Viola, *Complex Feature Recognition: A Bayesian Approach for Learning to Recognize Objects*, AI Memo 1591, MIT Artificial Intelligence Laboratory, Cambridge, Mass., Nov. 1996.

Luiz Marcos Garcia is currently a contractual researcher at the Robotics and Artificial Intelligence group at LAAS-CNRS, in France. His main technical interests include computer vision, robotics, and computer graphics. He received his BSc in computer science from the State University of Rio de Janeiro, Brazil. He has an MSc and a DSc from the Federal University of Rio de Janeiro. He is a member of the IEEE Computer Society. Contact him at LAAS/RIA-CNRS, 7 Ave. du Colonel Roche, 31077 Toulouse Cedex 04 France; lmgarcia@laas.fr; lmarcos@computer.org.

Roderic A. Grupen is an associate professor of the Department of Computer Science at the University of Massachusetts, Amherst. His main interests include computing, operations research, and control theory as a means of modeling intelligent systems—natural and artificial. He received a BA in physics from Franklin and Marshall College, a BS in mechanical engineering from Washington University, an MS in mechanical engineering from Pennsylvania State University, and a PhD in computer science from the University of Utah. Contact him at the Univ. of Massachusetts, 140 Govenor's Dr., Computer Sciences Building, Amherst, MA 01003; grupen@cs.umass.edu.

Antonio A. F. Oliveira is an associate professor in the Laboratory for Computer Graphics and Vision at the Federal University of Rio de Janeiro. His technical interests include computer vision, computational geometry, image processing, and computer graphics. He received his BSc in electrical engineering from the Federal University of Rio de Janeiro, Brazil. He has an MSc and a DSc in systems engineering and computer science from the Federal University of Rio de Janeiro. Contact him at COPPE/UFRJ Progama de Sustemas, CP 68511, 21945-970, Rio de Janeiro, Brasil; oliveira@lcg.ufrj.br.

David S. Wheeler is a graduate student and research assistant in the Department of Computer Science at the University of Massachusetts, Amherst. His interests include machine learning and robotics. He has a BS in computer systems engineering from the University of Massachusetts. Contact him at the Univ. of Massachusetts, 140 Govenor's Dr., Computer Sciences Building, Amherst, MA 01003; dwheeler@cs.umass.edu.

Andrew H. Fagg is a research assistant professor in the Department of Computer Science at the University of Massachusetts, Amherst. His primary interests include computational neuroscience, machine learning, autonomous robotics, and wearable computing. He has a BS in applied mathematics and computer science from Carnegie Mellon University, and an MS and a PhD in computer science from the University of Southern California. Contact him at the Univ. of Massachusetts, 140 Govenor's Dr., Computer Sciences Building, Amherst, MA 01003; fagg@cs.umass.edu.