SENSORIMOTOR COORDINATION IN A "BABY" ROBOT: LEARNING ABOUT OBJECTS THROUGH GRASPING

LORENZO NATALE
Italian Institute of Technology
via Morego, 30 16163 Genoa, Italy
Tel: +39 010 7178-1420
E-mail: lorenzo.natale@iit.it

FRANCESCO ORABONA
LIRA-Lab, DIST, University of Genova
viale Causa 13, 16145, Genova, Italy
Tel: +39 010 353-2946
Fax: +39 010 353-2948
E-mail: bremen@liralab.it

GIORGIO METTA
LIRA-Lab, DIST, University of Genova
viale Causa 13, 16145, Genova, Italy
AND
Italian Institute of Technology
via Morego, 30 16163 Genoa, Italy
Tel: +39 010 7178-1411
E-mail: pasa@liralab.it

GIULIO SANDINI
Italian Institute of Technology
via Morego, 30 16163 Genoa, Italy
Tel: +39 010 7178-1416
E-mail: giulio.sandini@iit.it

Contact author: GIORGIO METTA – E-mail: pasa@liralab.it

ABSTRACT

This paper describes a developmental approach to the design of a humanoid robot. The robot, equipped with initial perceptual and motor competencies, explores the "shape" of its own body before devoting its attention to the external environment. The initial form of sensorimotor coordination consists of a set of explorative motor behaviors coupled to visual routines providing a bottom-up sensory-driven attention system. Subsequently, development leads the robot from the construction of a "body schema" to the exploration of the world of objects. The "body schema" allows controlling the arm and hand to reach and touch objects within the robot's workspace. Eventually, the interaction between the environment and the robot's body is exploited to acquire a visual model of the objects the robot encounters which can then be used to guide a top-down attention system.

Keywords: development, humanoid robotics, body schema, top-down and bottom-up attention

(1) Introduction

In the past few years there has been significant technological advance in computer technology and robotics. Today computers are much more powerful than they used to be and they can be interconnected through fast networks, which allow efficient parallel computation. At the same time digital cameras have higher resolution, better quality and higher frame rate. This notwithstanding, we are still far from achieving the dream of artificial intelligence. Artificial systems (computer programs, expert systems or robots) are not able to face the challenges of the real world. We are still not capable of

building devices which are able to cope with the variability of the world where, on the other hand, even the simplest animal can thrive. Likewise there is a growing interest in the scientific community to the study of cognitive systems with the aim of implementing cognitive abilities in artificial systems. The study of cognition is still in the pre-paradigmatic stage and, indeed, little agreement can be found even in its definition (see (Clark, 2001) for a review). According to *cognitivism*, cognition is "a computational process carried out on a symbolic representation of the world". Symbols represent the world and can be shared across different entities (artificial or biological); they are a complete characterization of the world in which the entity is located, and as such are independent of the entity itself and its past experience. Somewhat at the other extreme, emergent approaches define cognition as the result of the interaction and co-development between the agent's body and the environment in which it lives (Maturana and Varela, 1998, Beer, 2000, Sandini et al., 2004).

Although the definitive answer is still to be found, the observation of biological systems provides hints to plausible solutions. Two aspects look crucial: i) the existence of a body (embodiment) and ii) the fact that the internal representation of the world is acquired by acting in the environment. The two requirements are obviously intertwined, as the interaction between the agent and the environment is possible only by means of a physical body. As a consequence, internal representations become function of the particular embodiment and, perhaps more importantly, of the history of experiences of the agent.

Subscribing to the emergent approach implies that internal representations cannot be built into the system "by design"; instead the cognitive system has to be able to create these representations by directly interacting with the environment or, indirectly,

with other agents. Through action, the embodiment and environment co-determine the resulting representations.

Motivated by these considerations, this paper proposes a developmental approach to the realization of a number of cognitive abilities in a humanoid robot. Although a fair amount of cognitivism is still present, especially in the realization of the visual system, learning permeates the implementation at various levels. Learning and a certain degree of adaptation is clearly the prerequisite to a fully emergent design, although not yet an end or a definite answer to the understanding of cognitive systems altogether.

We identified the minimum requirements for our robot as having an oculomotor system, an arm, and a hand. Although simplified this configuration suffices in allowing active manipulation of the world via reaching and grasping. The robot follows a developmental route that goes initially through the exploration of its body and terminates into the characterization of external objects (e.g. segmentation) by effect of grasping.

Conceptually this process can be divided in three phases. The first stage is devoted to learning the internal models of the body (we call it “learning the body-map”) which provides basic motor and perceptual skills like gaze control, eye-head coordination and reaching. Based on these abilities the interaction with the external world is investigated in the second phase where the robot discovers properties of objects and ways of handling them (learning to interact). The robot tries simple stereotyped actions like pushing/pulling and grasping of objects which allow to start the acquisition of information about the entities that populate its environment and simultaneously discover new more efficient ways of interaction (for example different grasp types). Finally the third stage concerns learning to understand and interpret events; the robot has associated

its actions with the resulting perceptual consequences. Interpretation is achieved by inverting this association; perceptions are projected into the corresponding actions which work as a reference frame to give meaning to what happens in the environment.

In our past work we have addressed some of the aspects related to this third phase (Natale et al., 2002, Fitzpatrick et al., 2003). In this paper we focus on the two first phases: learning a body-map and learning to interact.

We show how the robot can acquire an internal model of its hand which allows the robot to localize it and anticipate its position in the visual scene during action execution. The hand internal model is then used to learn to reach a point in space and to accommodate the position of the hand with respect to the object during grasping. The robot uses these abilities to build a visual model of the objects it grasps. Once an object is grasped, in fact, the robot can move and rotate it to build a statistical model of its visual appearance.

(1) Experimental Platform

The experiments reported in this paper were carried out on a robotic platform called Babybot (Figure 1). The Babybot is an upper torso humanoid robot which consists of a head, an arm and a hand. The head has 5 degrees of freedom, two of which control the neck in the pan and tilt direction, whereas the other three actuate the two eyes to pan independently and tilt on a common axis. The arm is a Unimate PUMA 260, an industrial manipulator with 6 degrees of freedom; it is mounted horizontally to better mimic the human kinematics. The hand has 5 fingers; each finger has three phalanges, the thumb has an additional degree of freedom which allows it to perform a rotation toward the palm. Overall the number of joints is 16 but for reasons of space and weight they are

controlled by using only six motors. Two motors are connected to the index fingers: they are linked to the first (proximal) and second phalanges. The distal (small) phalange is mechanically coupled to the preceding one so that the two bend together (see Figure 1). Two motors control the motion of middle, ring and little finger. As in the case of the index finger, the proximal phalanges are actuated by a single motor, while the second and third phalanges are actuated by a second motor. The mechanical coupling between the joints is realized by means of springs to allow a certain degree of adaptation in case of physical contact or impact with solid objects. For example, during a movement of flexion of the fingers toward the palm, if the middle finger were to be blocked by an obstacle the others would continue to bend up to the equilibrium of the torque generated by the motor and that of the spring (Figure 1 b) and c)). The same would happen in case the distal phalanges had hit the obstacle. The thumb is different as one motor controls the rotation around an axis parallel to the palm and a second motor is connected to the three phalanges, whose independent motion is permitted by elastic coupling as for the other fingers.

The sensory system of the Babybot consists of two cameras and two microphones for visual and auditory feedback. Tactile feedback is provided by 17 force sensing resistors mounted on the hand, five of which are placed on the palm and the remaining 12 evenly distributed on the thumb, index, middle and ring fingers. A JR3 6-axial force sensor provides torque and force feedback measured at the wrist. Further proprioceptive information is provided by encoders mounted on all motors and by a three-axis gyroscope mounted on the head. More details about the Babybot architecture can be found elsewhere (Natale, 2004).

[FIGURE 1 about here]

(1) Visual System

One of the first steps of any visual system is that of locating suitable interest points in the scene ("salient regions" or events) and eventually direct gaze toward these locations. Human beings and many animals do not have a uniform resolution view of the visual world but rather only a series of snapshots acquired through a small high-resolution sensor (e.g. our fovea). This leads to two questions: i) how to move the eyes efficiently to important locations in the visual scene, and ii) how to decide what is important and, as a consequence, where to look next.

The literature follows two different approaches in the attempt of accounting for these facts. On the one hand, the space-based attention theory holds that attention is allocated to a region of space, with processing carried out only within a certain spatial window. Attention in this case could be directed to a region of space even in absence of a real target (the most influential evidences for the spatial selection come from the experiments of Posner, Snyder and Davidson (Posner et al., 1980) and Downing and Pinker (Downing and Pinker, 1985)).

On the other hand, object-based attention theories argue that attention is directed to an object or a group of objects, and that the attention system processes properties of object(s), rather than regions of space. This object-based theory is supported by growing behavioral and neurophysiological evidence (Egly et al., 1994, Scholl, 2001). In other words, the visual system seems optimized for segmenting complex three-dimensional scenes into representations of (often partly occluded) objects for recognition and action.

Indeed, perceivers must interact with objects in the world and not with disembodied locations.

Finally, another classification can be made depending on which cues are actually used in modulating attention. One approach uses bottom-up information including basic features such as color, orientation, motion, depth, and conjunctions of features. A feature or a stimulus catches the attention of the system if it differs from its immediate surrounding in some dimensions and the surround is reasonably homogeneous in those same dimensions. However higher level mechanisms are involved as well; a bottom-up stimulus, for example, may be ignored if attention is already focused elsewhere. In this case attention is also influenced by top-down information relevant to a particular task.

In the literature a number of attention models that use the first hypothesis have been proposed (Giefing et al., 1992, Milanese, 1993, Itti et al., 1998); most of them are derived from Treisman's Feature Integration Theory (FIT) (Treisman and Gelade, 1980). This model employs a separate set of low-level feature maps which are combined together by a spatial attention window operating in a master saliency map. An important alternative model is given by Sun and Fisher (Sun and Fisher, 2003), who proposed a combination of object- and feature-based theory (this model, unfortunately, requires hand-segmented images as input for training).

While it is known that the human visual system extracts basic information from images such as lines, edges, local orientation etc., vision not only represents visual features but also the items that such features characterize. But to segment a scene into items, objects, that is to group parts of the visual field as units, the concept of "object" must be known by the system. In particular, there is an intriguing discussion underway in vision science about reference to entities that have come to be known as "proto-objects"

or "pre-attentive objects" (Pylyshyn, 2001). These are steps up from mere localized features, and they have some but not all of the characteristics of objects.

The visual attention model we propose starts by considering the first stages of the human visual system, using then a concept of salience based on “proto-objects” defined as blob of uniform color in the images. Then, since the robot can act on the world, it can do something more: once an object is grasped the robot can move and rotate it to build a statistical model of the color blobs, thus effectively constructing a representation of the object in terms of proto-objects and their spatial relationships. This internal representation feeds then back to the attention system of the robot in a top-down way; as an example we show how the latter can be used to direct attention to spot one particular object among others that are visible on a table in front of the robot.

Our approach integrates bottom-up and top-down cues; in particular bottom-up information suggests/identifies possible regions in the image where attention could be directed, whereas top-down information works as a prime for those regions during the visual search task (i.e. when the robot seeks for a known object in the environment).

(2) Log-polar images

Figure 2 shows the block diagram of the first stage of the visual processing of the robot. The input data is a sequence of color log-polar images (Sandini and Tagliasco, 1980). The log-polar transformation models the mapping of the primate visual pathways from the retina to the visual cortex. The idea of employing space-variant vision is derived from the observation that the distribution of the cones, i.e. the photoreceptors of the retina involved in diurnal vision, is not uniform: cones have a higher density in the central region called fovea, while they are sparser in the periphery. Consequently the resolution

is higher and uniform in the center while it decreases in the periphery proportionally to the distance from the fovea.

The main advantage of log-polar sensors is computational, as they allow to acquire images with a small number of pixels and yet to maintain a large field of view and high resolution at the center (Sandini and Tagliasco, 1980). Moreover, this particular distribution of the receptors seems to influence the scan-paths of an observer (Wolfe and Gancarz, 1996), so it has to be taken into account to better model the overt visual attention.

[FIGURE 2 about here]

The radial symmetry of the distribution of the cones can be approximated by a polar distribution, whereas their projection to the primary visual cortex is well represented by a logarithmic-polar (log-polar) distribution mapped onto an approximately rectangular surface (the cortex). From the mathematical point of view the log-polar mapping can be expressed as a transformation between the polar plane (ρ,θ) (retinal plane), the log-polar plane (ξ,η) (cortical plane) and the Cartesian plane (x, y) (image plane), as follows (Sandini and Tagliasco, 1980):

$$\begin{cases} \eta = q \cdot \theta, \\ \xi = \log_a \dfrac{\rho}{\rho_0}. \end{cases} \tag{1}$$

where ρ_0 is the radius of the innermost circle, $1/q$ is the minimum angular resolution of the log-polar layout and (ρ,θ) are the polar co-ordinates.

Figure 3 illustrates the log-polar layout by showing a standard rectangular image and its log-polar counterpart. It is worth noting that the flower's petals, that have a polar structure, are mapped horizontally in the log-polar image. Circles, on the other hand, are mapped vertically. Furthermore, the stamens that lie in the center of the image of the flower, occupy about half of the corresponding log-polar image (the cortical magnification).

[FIGURE 3 about here]

(2) Visual attention

As a first step the input image is smoothed, by taking the average between the current frame and the output of the color quantization (see later) on the previous frame. Then the red, green, and blue channels of each image are separated, and the yellow channel is calculated as the mean of the red and green one. These four channels are combined to generate three color opponent channels, similar to those of the retina. Each of these channels, typically indicated as (R+G-, G+R-, B+Y-), has a center-surround receptive field (RF) with spectrally opponent color responses. That is, for example, a red input in the center of a particular RF increases the response of the channel R+G-, while a green one in the surrounding decreases its response. The spatial response profile of the RF is expressed by a Difference-of-Gaussians (DoG) function. Each pixel is considered as the center of a RF, so that the output of the RF filtering is simply obtained by a convolution of the whole image with a DoG kernel, generating an output image of the

same size of the input. This computation, considering for example the *R+G-* channel, is expressed by:

$$R^{+}G^{-}(\mathbf{x}) = a \cdot R(\mathbf{x}) \otimes \gamma_c(\mathbf{x}, \sigma_c) - b \cdot G(\mathbf{x}) \otimes \gamma_s(\mathbf{x}, \sigma_s). \quad (2)$$

The two Gaussian functions $\gamma_c(\mathbf{x}, \sigma_c)$ and $\gamma_s(\mathbf{x}, \sigma_s)$ are not balanced and the ratio *b/a* is 1.5, consistent with the study of Smirnakis et al. (Smirnakis et al., 1997) Similarly to what happens in the human retina (Billock, 1995) the unbalanced ratio implicitly code the achromatic information. It is worth noting that filtering the log-polar images with a standard space-invariant filter corresponds to a space-variant filtering in the original Cartesian image (von Seelen and Mallot, 1990).

Edges are then extracted on the three channels separately by employing a generalization of the Sobel filter due to Li et al. (Li et al., 2003). The resulting edge maps are combined together to generate a single map as follows:

$$E(\mathbf{x}) = \max\left\{abs\left(E_{RG}(\mathbf{x})\right), abs\left(E_{GR}(\mathbf{x})\right), abs\left(E_{BY}(\mathbf{x})\right)\right\}. \quad (3)$$

It has to be noted that the log-polar transform has the side effect of sharpening the edges near the fovea due to the already mentioned magnification factor. To compensate for this effect the edge map is multiplied by an exponential function, and normalized to a fixed range (0-255).

It has been speculated, that synchronizations of visual cortical neurons may serve as the carrier for the observed perceptual grouping phenomenon (Eckhorn et al., 1988, Gray et al., 1989). The differences in oscillator phase between spatially neighboring spiking cells could be used in principle to label different objects in the scene. We have used a watershed transform (rainfalling variant) (Vincent and Soille, 1991, Smet and Pires, 2000) on the edge map to simulate the result of this synchronization and to

generate the proto-objects. The activation is spread from the center of the image (in the edge map) until all spaces between edges are filled in. As a result the image is segmented into blobs with either uniform color or uniform gradient of color.

Each blob is then tagged with the mean color of the pixels within its internal area (this leads to a sort of quantized image). The result is blurred with a Gaussian filter and stored: it will be averaged with the next frame to obtain a temporal smoothing and reduce the effect of noise. After an initial startup delay of 4-5 frames, the number of blobs and their size stabilizes.

As discussed above, it is known that a feature or stimulus is salient if it differs from its immediate surrounding area. We chose to calculate the bottom-up salience as the Euclidean distance in the color opponent space between each blob and the average color in a ball surrounding it. The radius of the ball (the spot or focus of attention) is not fixed: it changes with the size of the objects in the scene. In the same way the definition of "immediate surrounding area" should be relative to the size of the focus of attention. For this reason the greater part of the visual attention models in the literature uses a multi-scale approach and filters the salience map with suitable filters, or "blob" detectors (Itti and Koch, 2001). These approaches lack continuity in the choice of the size of the attention focus. We propose instead to vary dynamically the region of interest depending on the size of the blobs. In other words, we compute the salience of each blob in relation to a neighborhood region whose size is proportional to that of the blob itself. In our implementation we use a rectangular region 3 times the size of the bounding box of the blob. The choice of a rectangular window is not incidental, it was chosen because filters over rectangular regions can be computed efficiently by employing the integral image as in (Viola and Jones, 2004). Blobs that are too small or too big are discarded from the

saliency computation and will not be considered as possible candidates to be part of objects (proto-objects).

The bottom-up saliency is computed as:

$$S_{bottom-up} = \frac{1}{\sqrt{3}} \sqrt{\left(\langle R^+G^- \rangle_{blob} - \langle R^+G^- \rangle_{surround} \right)^2 + \left(\langle G^+R^- \rangle_{blob} - \langle G^+R^- \rangle_{surround} \right)^2 + \left(\langle B^+Y^- \rangle_{blob} - \langle B^+Y^- \rangle_{surround} \right)^2}. \quad (4)$$

where $\langle \ \rangle$ indicates the average of the pixel values over a certain area (as in the subscripts).

The top-down influence on attention is, at the moment, calculated in relation to the visual search task. When the robot has acquired a model of the object and begins searching for it, it uses the visual information of the object to bias the saliency map. In practice, the top-down saliency map is computed as the distance between the average color of each blob and that of the target:

$$S_{top-down} = 255 - \frac{1}{\sqrt{3}} \sqrt{\left(\langle R^+G^- \rangle_{blob} - \langle R^+G^- \rangle_{object} \right)^2 + \left(\langle G^+R^- \rangle_{blob} - \langle G^+R^- \rangle_{object} \right)^2 + \left(\langle B^+Y^- \rangle_{blob} - \langle B^+Y^- \rangle_{object} \right)^2}. \quad (5)$$

The total salience is simply estimated as the linear combination of the two terms above:

$$S = \alpha \cdot S_{top-down} + \beta \cdot S_{bottom-up}. \quad (6)$$

The total salience map S is eventually normalized in the range 0-255, as a consequence the salience of each blob in the image is relative to the most salient one. The target of the next saccade is the center of mass of the most salient blob (this is in agreement with human behavior (Melcher and Kowler, 1999).

As a final note on efficiency, it is worth saying that the use of log-polar images allows to compute the saliency map in real-time (15 frames per second on a 2.8Ghz Pentium IV).

(2) IOR

Local inhibition is transiently activated in the salience map. This prevents the focus of attention to be redirected immediately to a location that was previously attended. Experiments in human psychophysics have demonstrated the existence of such an "inhibition of return" (IOR) coded in an allocentric reference frame (Posner and Cohen, 1984) and in an object-based coordinates (Tipper, 1994).

Our system implements a simple object-based IOR. The robot maintains a list of the last five positions (Wolfe, 2003) it has visited, coded in a body centered coordinate system. The color information of the relative blobs is also stored in the list which is updated with a First-In First-Out policy. When the robot moves its gaze – for example by moving the eyes or the head in coordination – it keeps memory of the blobs it has visited earlier. Inhibition occurs only if the blob presents the same color that is stored in the list; in case the object moves or its color changes the location becomes available for fixation.

(1) Learning about the Self

Internal models are thought to be available to the brain and responsible for formulating predictions about the world or simulating the body (Wolpert and Miall, 1996). In general the collection of the internal models required to represent the body is called the *body-schema*: it involves, for example, the relative positions of the limbs, and their weight and size. In humans and biological systems the internal representation of the

body is shaped during development and maintained adapted to the physical modification occurring in life. In artificial agents (where the body does not change with time) adaptation can spare the tedious operation of manually tuning the system's internal models and their calibration. The latter might be required to compensate changes in the visual appearance of the body or drift in the sensors (e.g. the motor encoders).

In infants this sense of the body is acquired during development and emerges a few months after birth (Rochat and Striano, 2000). This is a cause-effect problem because on the one hand the brain uses internal models to recognize the body whereas on the other it has to acquire the body-schema and maintain it up to date. To solve it, the brain needs a "bootstrapping" mechanism which allows the identification of the body and, in this way, the acquisition of the internal representation. To distinguish the body from the rest of the world the brain is thought to take advantage of extra information. For example, while a child waves the hand in front of his eyes, his brains "knows" what kind of motion is producing since it has exclusive access to the motor commands it sends to the muscles and the relative proprioceptive feedback (Rochat and Striano, 2000).

In robotics there have been attempts to replicate self-recognition mechanisms. Yoshikawa and colleagues (Yoshikawa et al., 2003) exploit the invariance of the body with respect to the external world to train a neural network to segment the arm of the robot. Their idea is that during learning, when the robot moves in the environment, the background changes, whereas the arms remain stationary with respect to the proprioceptive feedback.

Instead, the active behavior of the robot is used by Metta and Fitzpatrick (Metta and Fitzpatrick, 2003); in this case the robot identifies its body because it moves with respect to the background. Since motion alone is not sufficient to segment out external

objects that move in the environment, the system seeks similarities between proprioceptive and visual feedback. Among the others, periodic actions may add robustness because offer the possibility to exploit repeatability (Fitzpatrick and Arsenio, 2004).

(2) Segmentation of the hand

Repeated, self-generated actions were performed by the robot during the learning phase. In particular the robot was programmed to execute periodic movements of the wrist. The resulting motion of the hand was detected by computing the image difference between the current frame and an adaptive model of the background. The period of motion of each pixel in the resulting motion image was then computed with a zero-crossing algorithm; similar information was extracted from the proprioceptive feedback of each motor encoder. As a result, the hand of the robot was segmented by selecting, among the pixels that moved periodically, those whose period matched that of the wrist joints. Conversely non-periodic pixels or pixels moving with different periods were identified as being externally originated and discarded. Figure 4 shows an example of the detection for two different pixels whose motion was (a) correlated and (b) uncorrelated with that of the robot's hand. Low-pass filtering and a threshold was applied after the detection to obtain a dense segmented image (see Figure 5).

[FIGURE 4 about here]

This algorithm forces the robot to stop and wait until the periodic movement of the wrist is performed. For this reason it is not useful during action or to drive a feedback

control loop; it is instead ideally suited as a bootstrapping mechanism to acquire an internal model of the hand which can provide faster localization. In practice this was implemented with two neural networks: one trained to compute the position of the hand in the visual field given the current arm and head posture, and another to estimate the hand's shape and orientation (in this case the hand was represented as an ellipse). Indeed, these neural networks can also predict the expected location and the (simplified) appearance of the hand in the visual field given the current posture of the robot (its "felt" position). The approach we followed here to perform the segmentation of the hand is similar to the one of Metta and Fitzpatrick (Metta and Fitzpatrick, 2003); the main difference with our approach is the use of periodicity that allows the detection of the hand in real time at high resolution. The result is a dense segmentation from which it is possible to derive additional information like shape and orientation.

(2) The hand internal model, expectation and prediction

To gather the training data the robot moved the arm randomly and then waved the hand for a few seconds; for each spatial location the segmentation of the hand was performed as described in the previous section. For each trial the center of mass of the segmented area was computed along with the best fitting ellipse parameters. The complete algorithm is reported in Figure 6.

[FIGURE 5 about here]

The resulting (x, y) coordinates were used to train the first neural network whereas the ellipse parameters (orientation, major and minor axis) constituted the

training samples for the second neural networks. It is important to take into account that the position of the hand in the visual field depends both on the posture of the arm and head (other parameters like orientation and size of the hand are less influenced, if not at all). Unfortunately this enlarges the learning space and increases the time required for exploration (to collect the training set) and learning (higher dimensionality). For this reason the position of the hand was projected into an egocentric reference frame before being used to train the neural network. This last operation significantly reduced the dimensionality of the input space of the neural network. When needed, the output of the neural network is projected back to the retinocentric reference frame. Both projections (back and forth from egocentric and retinocentric reference frame) require knowledge of head inverse and direct kinematics. In the experiments reported here they were hardwired in the system, a possible procedure to learn a model of them is suggested by Arsenio (Fitzpatrick and Arsenio, 2004). Figure 7 reports the block diagrams of the two models.

[FIGURE 6 about here]

As learning module we employed a multi-layer perceptron network with sigmoidal units trained with backpropagation; learning was performed online by storing all new samples and performing batch learning every 100 new samples. The learning process was validated by testing the ability of the network to predict new samples; when a new sample was obtained the network was used to predict the output given the input. The resulting output was compared to the current sample and the error computed. The increasing ability of the network to predict new samples proved that learning was effective. Figure 8 (left) reports the plot of the error during an experiment (in this case the

error is computed in the image plane to simplify visualization of the results); the total time of this experiment was about two hours.

[FIGURE 7 about here]

At the end of the exploration phase the robot had trained an internal model of the hand by which it could i) localize its center of mass ii) estimate its orientation and approximate size. The output of these models is not based on actual visual feedback, but on the mere projection of the proprioceptive information about the hand: they represent the expectation the robot possesses about its body (in this case, the hand).

[FIGURE 8 about here]

These measures were used in numerous ways. The center of mass was employed to close a visual loop to direct gaze towards the hand (see Figure 8 right). For this task the internal model was addressed with the proprioceptive feedback of the arm. Another possibility was to address the model with the arm motor command (final joint position) to obtain the position of the hand at the end of the movement. In general this model offers a means of computing a prediction of the position, size and orientation of the hand from a given arm configuration or, in other words, of simulating a motor action. In the next section this will be used to learn the reaching map and estimate the visuomotor Jacobian matrix for a reaching task.

(2) Reaching

The solution we propose is based on the use of a direct mapping between the eye-head motor plant and the arm motor plant (Metta et al., 1999). Flanders and colleagues (Flanders et al., 1999) suggested that the information about gaze direction might be employed by the brain to establish a reference point for reaching. They analyzed the error when reaching in the dark and showed how this correlates to the error of the gaze (the gaze drifts away from the target in the dark). Accordingly one premise we make is that the position of the fixation point coincides with the object to be reached. In other words, reaching for an object starts by looking at it. Under this assumption, the fixation point can be considered as the "end-effector" of the eye-head system. The position of the eyes with respect to the head, determines uniquely the position of the fixation point in space relative to the shoulder. The arm motor command can be obtained by a transformation of the eye-head motor/positional variables. We called this approach "motor-motor coordination", because the coordinated action is obtained by mapping motor variables into motor variables:

$$\mathbf{q}_{arm} = \mathbf{f}(\mathbf{q}_{head}) \tag{7}$$

where $\mathbf{q}_{head}$ and $\mathbf{q}_{arm}$ are head and arm posture respectively (joint space).

What is interesting in this approach is not equation (7) per se, which, after all, implements the inverse kinematics of the arm, but the mechanisms used to learn it. In fact, this mapping can be easily learnt when the tracking behavior described in the previous section is active. The robot explored the workspace by moving the arm randomly, while simultaneously, it tracked its hand; whenever the eyes fixated the hand a new sample consisting of the arm and head joint angles was acquired and used to train a

neural network approximating equation (7). In this case learning was performed online by using the Schaal at al. model (Schaal and Atkeson, 1998). The exploration was conducted in two ways. A first movement of the arm was performed by sampling a random uniform distribution within the part of the arm workspace in front of the robot. Small subsequent movements were performed randomly with Gaussian distribution with zero mean and standard deviation equal to 5 degrees. This last step while not strictly required sped up learning by sampling quickly large portions of the arm's workspace: i.e. for small movements of the order of 5 degrees the arm fixation was achieved rapidly and thus a new sample was added to the training set. When a sufficient number of samples were acquired, the robot started using the motor-motor map to actively reach for visually identified objects while learning could continue.

Learning can be further improved by reducing the dimensionality of the input vector $\mathbf{q}_{head}$. In fact, only three variables are needed to code the position of the fixation point; for this purpose we decided to use azimuth, elevation, and distance – in substitution for the five angles of the head joints. This transformation is motivated by practical reasons, but it is also biologically plausible (Lacquaniti and Caminiti, 1998).

[FIGURE 9 about here]

Similarly to the previous section, learning was tested by comparing every new sample to the output of the network (see text for details). The graph of the error during an experiment is reported in Figure 9 (left) for each sample (dotted line) and the moving window average over 20 samples (the total time of the experiment was about one hour and a half). From the first plot it is hard to determine a real increment of performance as

several samples at the end of the learning session present relatively large errors. This is due to noise in the training data, which affects not only learning, but also the measure of performance. In particular noise is higher in those configurations of the arm where the hand is closer to the head and the system fails to control the angle of vergence between the eyes. In these situations the error is large because the position of the fixation points varies significantly (from very far to very close). The average error, however, has a distinguishable uniform trend. Figure 9 (right) shows a sequence of images taken from the robot left eye during an exemplar reaching action.

It is worth mentioning that there is no need to separate the exploration/training phase and reaching (exploitation). An initial "reflex" can be employed as substitute for the reaching map at the very beginning; this simple behavior could, for example, populate the robot workspace with three positions (left, center and right). Exploration in this case would still be guaranteed by a random procedure, similar to the one described earlier. This approach was followed in (Metta et al., 1999, Metta, 2000).

The reaching problem can also be solved in the image plane. Consider the planar case (i.e. no 3D information is available and one of the arm joints is maintained to a fixed position) and suppose to measure the position of the end point in the image plane $\mathbf{x}_{hand}$. We want to control the arm to reach a target point $\mathbf{x}^{*}_{hand}$. If the robot is not in a singular configuration we can solve the problem by following a standard visual servoing approach (Espiau et al., 1992, Hutchinson et al., 1996):

$$\dot{\mathbf{q}} = -k \cdot \mathbf{J}^{-1}\left(\mathbf{q}_{arm}\right) \cdot \Delta\mathbf{x}\,, \tag{8}$$

where:

$$\Delta\mathbf{x} = \mathbf{x}_{hand} - \mathbf{x}^{*}_{hand}\,, \tag{9}$$

$k > 0$ is a scalar and $\mathbf{J}(\mathbf{q}_{arm})$ is the Jacobian of the transformation between the image plane and the arm joint space. $\mathbf{J}^{-1}(\mathbf{q}_{arm})$ is 2 by 2 matrix whose elements are a non-linear function of the arm joint angles. Given $\mathbf{J}^{-1}(\mathbf{q}_{arm})$ it is possible to drive the endpoint toward any point in the image plane. At least locally $\mathbf{J}^{-1}$ can be approximated by a constant matrix:

$$J^{-1}(\mathbf{q}_{arm}) \approx \hat{\mathbf{J}}^{-1}(\mathbf{q}_{arm}) = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}. \quad (10)$$

Since following the procedure described in the previous section the robot has learnt a direct transformation between the arm joint angles and the image plane (see for example Figure 7), it can now recover the position of the endpoint from a given joint configuration:

$$\mathbf{x}_{hand} = \mathbf{f}(\mathbf{q}_{arm}). \quad (11)$$

Indeed, to compute a local approximation of $\mathbf{J}^{-1}$, a random sampling of the arm joint space around a given point $(\bar{\mathbf{x}}, \bar{\mathbf{q}})$ can be performed:

$$\mathbf{q_i} = \bar{\mathbf{q}} + \Delta\mathbf{q_i} \quad (12)$$

with

$$\Delta\mathbf{q_i} = \boldsymbol{\eta}(\mathbf{0}, \boldsymbol{\sigma}) \quad (13)$$

and where $\boldsymbol{\eta}(\mathbf{0}, \boldsymbol{\sigma})$ follows a normal distribution of zero mean and standard deviation of 5 degrees.

For each sample, by applying equation (11) we obtain a new value $\mathbf{x}_i = \mathbf{x} + \Delta\mathbf{x}_i$ that can be used to estimate $\mathbf{J}^{-1}$ around $\bar{\mathbf{q}}$ with a least squares procedure:

$$\Delta \mathbf{q}_i = \begin{bmatrix} \Delta \mathbf{x}_i^T & \mathbf{0} \\ \mathbf{0} & \Delta \mathbf{x}_i^T \end{bmatrix} . \begin{bmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \end{bmatrix} \tag{14}$$

$\hat{\mathbf{J}}^{-1}(\bar{\mathbf{q}})$ can then be used in the closed loop controller to drive the arm toward a specific position in the image plane. However, there is no need to close the loop with the actual visual feedback. By using the map in equation (11), in fact, we can substitute the actual visual feedback with the internal simulation provided by the model. From the output of the closed loop controller we can estimate the position of the arm at the next step, by assuming a pure kinematic model of the arm; in this way the procedure can be iterated several times to obtain the joint motor commands required to perform a reaching movement. The flowchart below explains this procedure.

In principle the inverse Jacobian could be learnt by using the visual feedback of the hand. In practice however this is often impractical because continuous visual feedback from the hand is rarely available. The approach we propose here requires only knowledge of the forward kinematics (as estimated in the previous section); the estimation of the inverse Jacobian with the approach we described is fast and can be easily performed online. Note also that the inverse Jacobian could have been computed analytically by taking the first derivative of equation (11). By selecting a least square solution, in our case, we added an extra smoothing factor that is beneficial in considering a control application. Also, in theory, our approach is more flexible since it does not require the knowledge of the number of units and structure of the neural network employed to approximate equation (11) and can be completely automatic.

[FIGURE 10 about here]

The main limitation of this approach is that we do not make use of three-dimensional visual information; while this is a clear limitation of this implementation, the same approach can be easily extended to the full 3D case. The implementation is consistent with the hand internal model which provides the position of the hand in the image plane of one of the eyes only (left). Since in the Babybot the hand position is uniquely described by three degrees of freedom (the first three joints of the Puma arm), this technique was used to control only two of them (arm and forearm). Given the kinematics of the Puma arm this allowed to perform movements on the plane defined by the shoulder joint. Another point worth discussing is that the closed loop controller does not use real visual feedback, and, therefore, its accuracy depends on the precision of the hand internal model. To achieve better performances, actual visual feedback might be required.

[FIGURE 11 about here]

Let us summarize what we have described in this section. We have introduced two approaches to solving the inverse kinematics of the manipulator. The first method uses a mapping between the posture of the head (whose fixation point implicitly identifies the target) and the arm motor commands; it allows controlling the arm to reach any point fixated by the robota. The second approach uses the hand internal model to compute a piecewise constant approximation of the inverse Jacobian and simulate small movements of the arm in the neighborhood of the desired target. The procedure is iterated

[a] During the learning of the motor-motor map, the robot tracks the palm of the hand.

several times to compute the motor command required for reaching the target. Reaching in this case is planned in the image plane; however, since the internal model is two dimensional, the approach is limited to the plane identified by the shoulder. For these reasons, the two methods were mixed in the experiment reported in the next section. The motor-motor mapping is employed to plan a first gross movement to approach the target, whereas the "closed-loop approach" allows a finer positioning of the fingers on the target. This second part of the movement is planned by considering the point of the ellipse at maximum distance from the robot's body (which corresponds to the fingers) as the arm endpoint (Figure 11). This strategy proved successful because it substantially increased the probability to grasp the objects on the table.

Once the robot has computed the final arm posture, planning of the actual movement is still required. This was done with a simple linear interpolation between the current and final arm configuration. The trajectory was divided in steps which were then effected by the low level controller; to this purpose we employed a low-stiffness PD controller with gravity compensation. The gravity load term for each joint was learnt online as described in Natale (Natale, 2004).

(1) Learning about Objects

In this section we describe a method for building a model of the object the robot grasps. We assume for a moment that the robot has already grasped an object; this can happen because a collaborative human has given the object to the robot (as we describe in the next section) or because the robot has autonomously grasped the object. In this case the robot may spot a region of interest in the visual scene and apply a stereotyped action involving the arm and hand to catch it. Both solutions are valid bootstrapping behaviors

for the acquisition of an internal model of the object. When the robot holds the object it can be explored through movements of the arm and rotations of the wrist.

In short, the idea is to represent objects as collections of blobs generated by the visual attention system and their relative positions (neighboring relations). The model is created statistically by looking at the same object several times from different points of view (see Figure 12). At the same time the system estimates the probability that each blob belongs to the object by counting the number of times each blob appears during the exploration.

In the following, we use the probabilistic framework proposed by Schiele and Crowley (Schiele and Crowley, 1996a, Schiele and Crowley, 1996b). We want to calculate the probability of the object O given a certain local measurement M. This probability P(O|M) can be calculated using Bayes' formula:

$$P(O \mid M) = \frac{P(M \mid O)\,P(O)}{P(M)}. \tag{15}$$

where P(O) is the a priori probability of the object O, P(M) the a priori probability of the local measurement M, and P(M|O) is the probability of the local measurement M when the object O is fixated. In the following experiments we carried out only a single detection experiment, there are consequently only two classes, one representing the object and another representing the background. For lack of better estimations we set P(O) and P(~O) to 0.5 (this is equivalent to doing a maximum likelihood estimation).

Since a single blob is not discriminative enough, we considered the probabilities of observing pairs of blobs; the local measurement M becomes the event of observing both a central (i.e. fixated) and surrounding blobs:

$$P(M \mid O) = P\left(B_i \mid B_c \text{ and } \left(B_i \text{ adiacent } B_c\right)\right). \tag{16}$$

where B_i is the i^{th} blob surrounding the central blob B_c which belongs to the object O. That is, we exploit the fact the robot is fixating the object and assume B_c to be constant across fixations of the same object – this is guaranteed by the fact the object is being hold by the hand. In practice this corresponds to estimating the probability that all blobs B_i adjacent to B_c (which we take as a reference) belong to the object. Moreover the color of the central blob B_c will be stored to be used during visual search to bias the salience map. This procedure, although requiring the "active participation" of the robot (through gazing) is less computationally expensive compared to the estimation of all probabilities for all possible pairs of blobs of the fixated object. Estimation of the full joint probabilities would require a larger training set than the one we used in our experiments. For the same reason we assumed statistical independence of the blobs of the objects; under this assumption the total probability P(M1,…,MN|O) can be factorized in the product of the probabilities P(Mi|O). The probabilities P(M|~O) are estimated during the exploration phase with the blobs not adjacent to the central blob. An object is detected if the probability P(O|M1,…,MN) is greater than a fixed threshold.

Our requirement was that of building the object model with the shortest possible exploration procedure. Unfortunately, the small training set might give histograms P(M|*) with many empty bins zero counts bins. To overcome this problem a probability smoothing method was used. A popular method of zero smoothing is Lidstone's law of succession: (Lidstone, 1920)

$$P(M \mid O) = \frac{count(M \wedge O) + \lambda}{count(O) + v\lambda}. \tag{17}$$

for a v valued problem. With λ=1 and a two valued problem (v=2), we obtain the well-known Laplace's law of succession. Following the results of Kohavi et al. (Kohavi et al.,

1997) we choose $\lambda=1/n$ where n is equal to the number of frames utilized during the training. The model of an object is trained in real-time; the duration of the training is determined by the time required by the robot to rotate and move the object with the hand (currently about 30 seconds).

When an object is detected after visual search, a possible figure-ground segmentation is attempted, using the information gathered during the exploration phase. Each blob is segmented from the background if it is adjacent to the central blob and if its probability to belong to the object is greater than 0.5. This probability is approximated using the estimated probability as follows:

$$P(B_i \in O \mid B_c \text{and} (B_i \text{ adiacent } B_c)) \cong P(B_i \mid B_c \text{and} (B_i \text{ adiacent } B_c)). \quad (18)$$

As an example Figure 13 shows the result of the segmentation procedure. These results could be further improved by adding some hypothesis about the regularity of the object boundary. However for the purpose of this paper (object identification for the manipulation task) these refinements were not necessary.

In table 1, results are shown of using a toy car and a toy airplane as target objects; 50 training sessions were performed for each object. The first column shows the recognition rate, the second the average number of saccades (mean ± standard deviations) it takes the robot to locate the target in case of successful recognition. The recognition rate of the toy airplane is lower than the one of the toy car because the former is more similar (by virtue of its color and number of blobs) to the background.

[TABLE 1 about here]

[FIGURE 12 about here]

[FIGURE 13 about here]

(1) Grasping Behavior

The modules described in the previous sections can be integrated to achieve an autonomous grasping behavior. Figure 14 can be used as a reference for the following discussion. The action starts when an object is placed in the robot's hand and the robot detects pressure in the palm (frame 1). This elicits a clutching action of the fingers; the hand follows a preprogrammed trajectory, the fingers bend around the object toward the palm. If the object is of some appropriate size, the intrinsic elasticity of the hand facilitates the action and the grasping of the object. The robot moves the arm to bring the object close to the cameras and begins its exploration. The object is placed in four positions with different orientations and background (frames between 2 and 6). During the exploration, the robot tracks the hand/object; when the object is stationary and fixation is achieved, a few frames are acquired and the model of the object trained as explained above. At the end of the exploration the object is released (frame 4). At this point the robot has acquired the visual model of the object and starts searching for it in the visual scene. To do this, it selects the blob whose features better match those of the object's main blob and perform a saccade. After the saccade the model of the object is matched against the blob that is being fixated and its surrounding. If the match is not positive the search continues with another blob, otherwise grasping starts (frames 7-8-9). At the end of the grasp the robot uses haptic information to detect whether it is holding the object or rather the action failed. In this process the weight of the object and its

consistence in the hand is checked (the shape of the fingers holding the object). If the action is successful the robot waits for another object, otherwise it performs another trial (search and reach).

It is fair to say that part of the controller was preprogrammed. The hand was controlled with stereotyped motor commands. Three primitives were used: one to close the hand after pressure was detected, and two during the grasping to pre-shape the hand and actually clasp the object. The robot relied on the elasticity of the hand to achieve the correct grasping. To facilitate grasping, the trajectory of the arm was also programmed beforehand; waypoints relative to the final position of the arm were included in the joint space to approach the object from the top.

[FIGURE 14 about here]

(1) Discussion and Conclusions

In this paper we have presented a developmental approach to the realization of cognitive abilities in a humanoid robot which starts from the exploration of the body and unfolds by eventually exploring the external world. The robot starts from a limited set of initial motor and perceptual competencies and autonomously develops more sophisticated ways to interact with the environment. This knowledge is used to begin the exploration of the environment and to build a visual model of the objects that are grasped.

We have presented an implementation of a visual attention system properly taking into account top-down and bottom-up information. The top-down system divides the visual scene into color blobs; each blob is assigned a saliency depending on the ratio

between its color and the color of the area surrounding it. The robot actively explores the visual appearance of the objects it grasps: every time an object is placed on the palm a statistical model of the blobs that are part of it is constructed. This information is subsequently fed to the attention system as a bottom-up primer to control the visual search of the same object. Thus the robot experience allows it to build a representation of the object with which it interacts while, at the same time, modulates the visual attention system. The robot's ability to act is used together with the body internal model to drive the exploration of the environment. This facilitates learning in different ways. Firstly it helps the robot to focus attention both in space and in time. During the acquisition of the object visual model, in fact, the robot can track the object because it knows the position of the hand from its proprioceptive feedback. The latter is also useful to detect when the acquisition of the model can be initiated because the object does not move and the eyes have acquired a stable fixation on it. Finally, the fact that the object is being held by the hand guarantees the link between different sensory modalities (for example the sight of the object and the kinesthetic information from the hand). The object model makes use of visual information; in (Natale et al., 2004) we show how it is possible to build a model of the objects based only on haptic information. In the future we would like to investigate the integration of the two approaches.

We support the enactive view of cognition in showing how much the body and the ability to build the representation of the external world through the interaction between the body and the environment can be useful for an autonomous agent. Even a simple set of behaviors (such as the one initially provided to the robot) is sufficient to begin the exploration of the environment and acquire an internal representation of it. On the other hand it is fair to say that much of the system presented in this paper is still

"cognitivist" and more or less carefully handcrafted into the robot. For practical reasons, our implementation lays in between a full emergent and a cognitivist approach although biologically informed choices were made when possible.

We have also shown how this initial body-environment interaction is sufficient to start linking actions with their resulting consequences to form prediction about the behavior of the robot. Very often prospective control is required to plan a successful action. During grasping, for example, the correct timing of preshaping and closure of the fingers is required; the lags in the sensory streams (visual and tactile) typical of artificial and natural systems make feedback control ineffective. To be able to anticipate the impact of the hand with the object, the robot is required to control the timing between preshaping and actual grasping; clearly this cannot be based only on visual and tactile feedback. Prospective control, however, is not only important for action. It gives an agent the possibility to create expectations on which to base the interpretation of the world and the actions performed by others. By means of the interaction with the world the agent builds a model of the behavior of external entities (objects, people, etc.) and the associated sensory feedback. This link can be used afterward to anticipate the consequences of a similar action and, eventually, to compare them with the real feedback. In the same way new situations can be interpreted by matching them against the robot's past experience. For example, the event of a ball that falls on the floor (and the resulting visual and auditory sensations) can be associated to the action of dropping it. Anticipation and predictions enhance the agents' ability to understand and interact with the environment and, for this reason, are important aspects of cognition. The results of this paper represent the first steps into the implementation of cognitive abilities in an artificial system. It is difficult to think, at least from an emergent perspective, of a

shortcut that prescinds from sensorimotor coordination in achieving cognitive skills to be used in the real world.

To conclude, we would like to comment on the effort required to build a complete robotic platform on the one hand, and the software architecture on the other. Presently the Babybot is an integrated robotic platform where it is extremely easy for software modules controlling different subparts (arm, head or hand to mention just a few) to exchange information and coordinate with each other (Metta et al., 2006). This is not very common, as usually in the literature papers report single experiments where the robotic platform is specifically programmed to perform the desired task, but care is not taken to realize a system which can grow in complexity as new modules are added. The experiment reported in the last section does not only show the integration between the visual attention system and the motor system but also the complexity of the system as a whole. We believe that this is a necessary prerequisite to carry out research in humanoid robotics as the complexity and number of skills increase.

Acknowledgments

This work is funded by the European Commission's Cognition Unit, Directorate-General Information Society, as part of project no. IST-2004-004370: RobotCub — ROBotic Open-architecture Technology for Cognition, Understanding, and Behaviour.

References

Beer, R., D. (2000) Dynamical approaches to cognitive science. *Trends in Cognitive Sciences*, 4**,** no. 3, pp. 91-99.

Billock, V. A. (1995) Cortical Simple Cells Can Extract Achromatic Information from the Multiplexed Chromatic and Achromatic Signals in the Parvocellular Pathway. *Vision Research*, 35**,** no. 16, pp. 2359-2369.

Clark, A. (2001) *Mindware: an introduction to the philosophy of cognitive science*, Oxford, UK, Oxford University Press.

Downing, C. and Pinker, S. (1985) The spatial structure of visual attention. In Posner, M. and Marin, O. S. M. (Eds.), *Attention and Performance*. Vol. XI, Lawrence Erlbaum Associates, Hillsdale, New Jersey, USA, pp. 171-187.

Eckhorn, R., Bauer, R., Jordan, W., Brosch, M., Kruse, M., Munk, W. and Reitboeck., H. J. (1988) Coherent oscillations: a mechanism of feature linking in the visual cortex? *Biological Cybernetics*, 60**,** no. 8, pp. 121-130.

Egly, R., Driver, J. and Rafal, R. (1994) Shifting visual attention between objects and locations: evidence for normal and parietal subjects. *Journal of Experimental Psychology: General*, 123**,** no. 2, pp. 161-177.

Espiau, B., Chaumette, F. and Rives, P. (1992) A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, 8**,** no. 3, pp. 313-326.

Fitzpatrick, P. and Arsenio, A. (2004). Feel the beat: using cross-modal rhythm to integrate perception of objects, others and self. In *Proc. of the Fourth International Workshop on Epigenetic Robotics*, August, 25-27, 2004, Genoa, Italy.

Fitzpatrick, P., Metta, G., Natale, L., Rao, S. and Sandini, G. (2003). Learning About Objects Through Action: Initial Steps Towards Artificial Cognition. In *Proc. of the IEEE International Conference on Robotics and Automation*, May 12-17, 2003, Taipei, Taiwan.

Flanders, M., Daghestani, L. and Berthoz, A. (1999) Reaching beyond reach. *Experimental Brain Research*, 126**,** no. 1, pp. 19-30.

Giefing, G. J., Janssen, H. and Mallot, H. A. (1992). Saccadic object recognition with an active vision system. In *Proc. of the International Conference on Pattern Recognition*.

Gray, C. M., König, P., Engel, A. K. and Singer, W. (1989) Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature*, 338**,** pp. 334-336.

Hutchinson, S., Hager, G. and Corke, P. (1996) A Tutorial on Visual Servo Control. *IEEE Transactions on Robotics and Automation*, 12**,** no. 5, pp. 651-670.

Itti, L. and Koch, C. (2001) Computational modelling of visual attention. *Nature Reviews Neuroscience*, 2**,** no. 3, pp. 194-203.

Itti, L., Koch, C. and Niebur, E. (1998) A Model of Saliency-Based Visual Attention for Rapid Scene Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20**,** no. 11, pp. 1254-1259.

Kohavi, R., Becker, B. and Sommerfield, D. (1997). Improving simple Bayes. In *Proc. of the European Conference on Machine Learning*.

Lacquaniti, F. and Caminiti, R. (1998) Visuo-motor transformations for arm reaching. *European Journal of Neuroscience*, 10**,** no. 1, pp. 195-203.

Li, X., Yuan, T., Yu, N. and Yuan, Y. (2003) Adaptive color quantization based on perceptive edge protection. *Pattern Recognition Letters*, 24**,** no. 16, pp. 3165-3176.

Lidstone, G. (1920) Note on the general case of the Bayes-Laplace formula for inductive or a posteriori probabilities. *Transactions of the Faculty of Actuaries,* 8**,** pp. 182-192.

Maturana, H., R. and Varela, F., J. (1998) *The tree of knowledge, the biological roots of human understanding*, Boston & London, Shambhala Publications, Inc.

Melcher, D. and Kowler, E. (1999) Shapes, surfaces and saccades. *Vision Research,* 39**,** no. 17, pp. 2929-2946.

Metta, G. (2000). *Baby$_{ro}$bot: A Study into Sensori-motor Development.* PhD Thesis, University of Genoa, Genoa, Italy.

Metta, G. and Fitzpatrick, P. (2003) Early Integration of Vision and Manipulation. *Adaptive Behavior,* 11**,** no. 2, pp. 109-128.

Metta, G., Fitzpatrick, P. and Natale, L. (2006) YARP: Yet Another Robot Platform. *International Journal on Advanced Robotic Systems, special issue on Software Development and Integration in Robotics,* 3**,** no. 1, pp. 43-48.

Metta, G., Sandini, G. and Konczak, J. (1999) A Developmental Approach to Visually-Guided Reaching in Artificial Systems. *Neural Networks,* 12**,** no. 10, pp. 1413-1427.

Milanese, R. (1993). *Detecting Salient Regions in an Image: From Biological Evidence to Computer Implementation.* PhD Thesis, University of Geneva, Geneva, Switzerland.

Natale, L. (2004). *Linking action to perception in a humanoid robot: a developmental approach to grasping.* PhD Thesis, University of Genoa, Genoa, Italy.

Natale, L., Metta, G. and Sandini, G. (2004). Learning haptic representation of objects. In *Proc. of the International Conference on Intelligent Manipulation and Grasping*, July 1-2, 2004, Genoa, Italy.

Natale, L., Rao S. and Sandini, G. (2002). Learning to act on objects. In *Proc. of the Second International Workshop, BMCV 2002*, November 22-24, 2002, Tubingen, Germany.

Posner, M. I. and Cohen, Y. (1984) Components of visual orienting. In Bouma, H. and Bouwhuis, D. G. (Eds.), *Attention and Performance.* Vol. X, Erlbaum, Hillsdale, NJ, pp. 531-556.

Posner, M. I., Snyder, C. R. R. and Davidson, B. J. (1980) Attention and the detection of signals. *Journal of Experimental Psychology: General,* 109**,** no. 2, pp. 160-174.

Pylyshyn, Z. (2001) Visual indexes, preconceptual object, and situated vision. *Cognition,* 80**,** no. 1-2, pp. 127-158.

Rochat, P. and Striano, T. (2000) Perceived self in infancy. *Infant Behavior and Development,* 23**,** no. 3-4, pp. 513-530.

Sandini, G., Metta, G. and Vernon, D. (2004). RobotCub: An Open Framework for Research in Embodied Cognition. In *Proc. of the IEEE-RAS/RJS International Conference on Humanoid Robotics*, November 10-12, 2004, Santa Monica, California, USA.

Sandini, G. and Tagliasco, V. (1980) An Anthropomorphic Retina-like Structure for Scene Analysis. *Computer Vision, Graphics and Image Processing,* 14**,** no. 3, pp. 365-372.

Schaal, S. and Atkeson, C. G. (1998) Constructive Incremental Learning from Only Local Information. *Neural Computation,* 10**,** no. 8, pp. 2047-2084.

Schiele, B. and Crowley, J. L. (1996a). Probabilistic object recognition using multidimensional receptive field histograms. In *Proc. of the 13th International Conference on Pattern Recognition*, August, 1996, Vienna, Austria.

Schiele, B. and Crowley, J. L. (1996b). Where to look next and what to look for. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Decmber, 1996, Osaka, Japan.

Scholl, B. J. (2001) Objects and attention: the state of the art. *Cognition,* 80**,** no. 1-2, pp. 1-46.

Smet, P. D. and Pires, R. (2000). Implementation and analysis of an optimized rainfalling watershed algorithm. In *Proc. of the IS&T/SPIE's 12th Annual Symposium Electronic Imaging 2000: Science and Technology*, 23-28 January, San Jose, California, USA.

Smirnakis, S. M., Berry, M. J., Warland, D. K., Bialek, W. and Meister, M. (1997) Adaptation of retinal processing to image contrast and spatial scale. *Nature*, 386**,** pp. 69-73.

Sun, Y. and Fisher, R. (2003) Object-based visual attention for computer vision. *Artificial Intelligence*, 146**,** no. 1, pp. 77-123.

Tipper, S. P. (1994) Object-based and environment-based inhibition of return of visual attention. *Journal of Experimental Psychology: Human Perception and Performance*, 20**,** no. 3, pp. 478-499.

Treisman, A. M. and Gelade, G. (1980) A feature integration theory of attention. *Cognitive Psychology*, 12**,** no. 1, pp. 97-136.

Vincent, L. and Soille, P. (1991) Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13**,** no. 6, pp. 583-598.

Viola, P. and Jones, M. J. (2004) Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57**,** no. 2, pp. 137-154.

von Seelen, W. and Mallot, H. A. (1990). Neural Mapping and Space-Variant Image Processing. In *Proc. of the International Joint Conference on Neural Networks (IJCNN)*, San Diego, California, USA.

Wolfe, J. M. (2003) Moving towards solutions to some enduring controversies in visual search. *Trends in Cognitive Sciences*, 7**,** no. 2, pp. 70-76.

Wolfe, J. M. and Gancarz, G. (1996) Guided Search 3.0. In Lakshminarayanan, V. (Eds.), *Basic and Clinical Applications of Vision Science*. Kluwer Academic, Dordrecht, Netherlands, pp. 189-192.

Wolpert, D. M. and Miall, R. C. (1996) Forward models for physiological motor control. *Neural Networks*, 9**,** no. 8, pp. 1265-1279.

Yoshikawa, Y., Hosoda, K. and Asada, M. (2003). Does the invariance in multi-modalities represent the body scheme ? - a case study with vision and proprioception -. In *Proc. of the 2nd Intelligent Symposium on Adaptive Motion of Animals and Machines*, Kyoto, Japan.

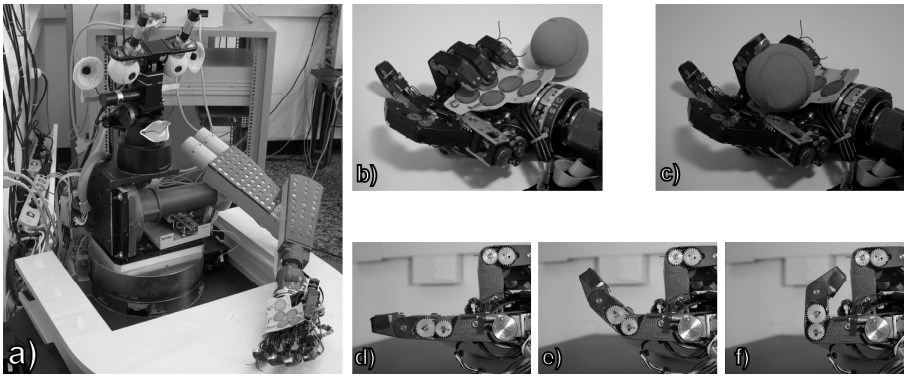

Figure 1. a) The experimental setup, the Babybot. Left: details of the hand. b) and c): elastic compliance. d)-f): mechanical coupling between phalanges.

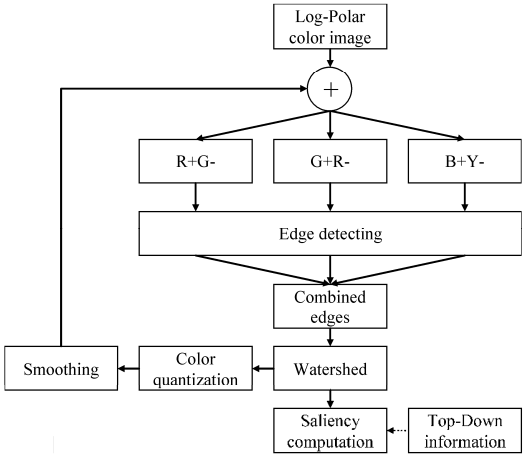

Figure 2. The visual attention system: block diagram (see text for details).

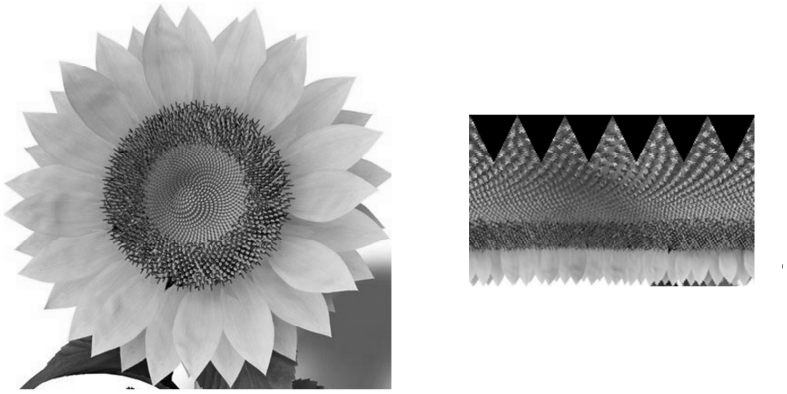

Figure 3. Log-polar mapping. The original image (left) and the result of the log-polar mapping in the cortical plane (right).

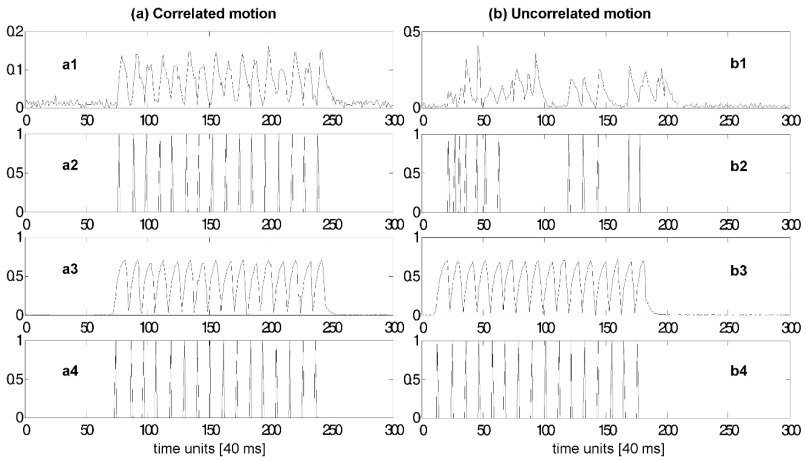

Figure 4. Correlated versus uncorrelated motion, an example. The plots represents the time course of the variables involved in the detection procedure for two exemplar pixels whose motion matched (a) and did not match (b) that of the hand. (a1) and (b1) show the value of the motion for the pixel (normalized between 0 and 1). The result of the zero-crossing algorithm is reported in (a2) and (b2). The same procedure is replicated for the wrist proprioceptive feedback: (a3) and (b3) show the speed of the joint (normalized arbitrary scale), whereas (a4) and (b4) show the result of the zero-crossing algorithm. Compare (a2) to (a4) and (b2) to (b4).

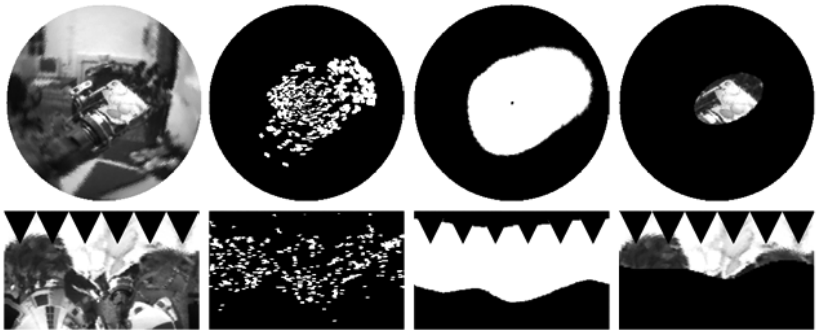

Figure 5. An example of the detection procedure. From left to right: the original image at the beginning of the procedure, the result of the detection (that is the pixels whose motion was correlated with that of the hand), the result of the low-pass filtering, the segmentation after the ellipse fitting. Notice that the ellipse tends to collapse towards the center, because the log-polar transformation gives more weight to the pixels close to the fovea.

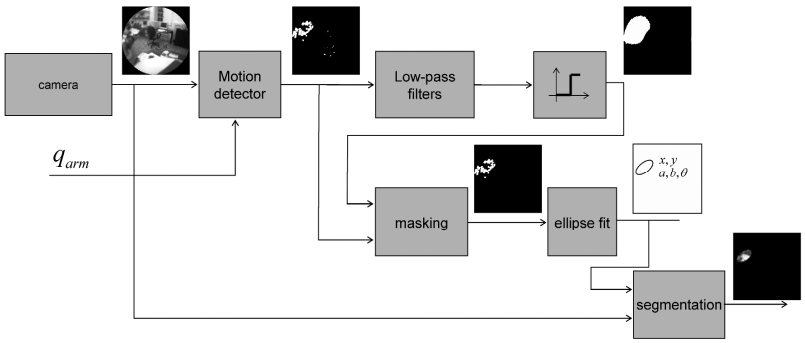

Figure 6. Detection algorithm, block schema. Images are captured from the camera. The "motion detector" block compares the motion in the image with the proprioceptive feedback from the arm (the wrist). A series of low-pass filters identify the blob which contains the hand. The blob is used to mask the result of the "motion detector" to remove possible outliers. An ellipse shape is fitted on the remaining pixels and, eventually, the hand is segmented.

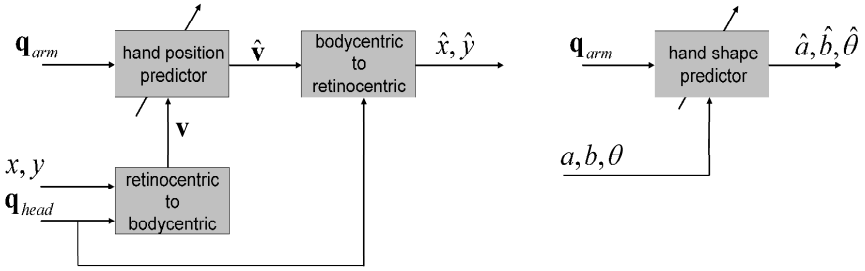

Figure 7. Left: hand position predictor. Right: hand shape predictor. In the experiments reported in this paper the learning modules were multi-layer perceptrons with a hidden layer and sigmoidal units.

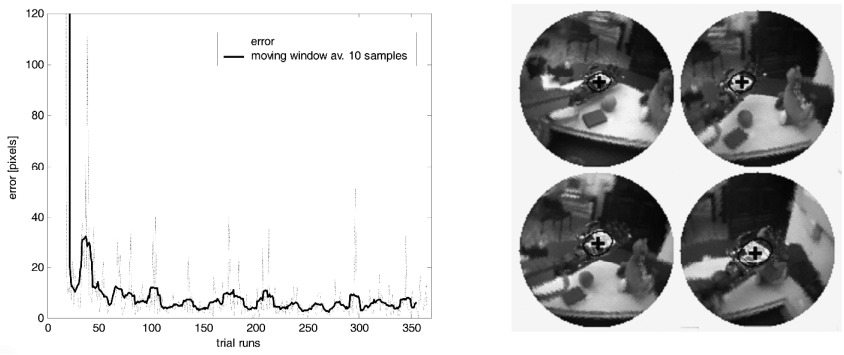

Figure 8. Hand localization error trend (left). As new examples are presented to the network the performance improves. Example of the localization after learning (right). The cross corresponds to the position of the hand, whereas the ellipse represents its approximate shape and orientation. The size of the network was 20 units in the hidden layer, the total time of this experiment was about two hours.

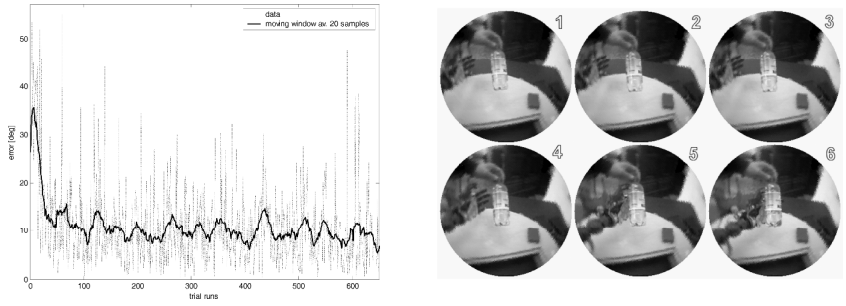

Figure 9. Reaching error (left). As new examples are gathered and presented to the network the performance increases. This improvement is less remarkable; we believe this is due to noise in the training data which affects not only learning, but also the measure of performance. An exemplar sequence of a reaching action after the learning is reported on the right. The number of units of the network after the learning was 12, the total time required to perform this experiment was about one hour and a half.

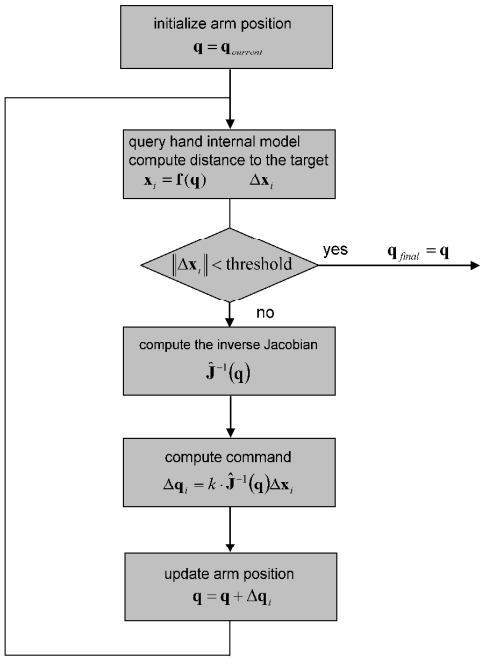

Figure 10. Closed-loop approach to reaching, flowchart. See text for further details.

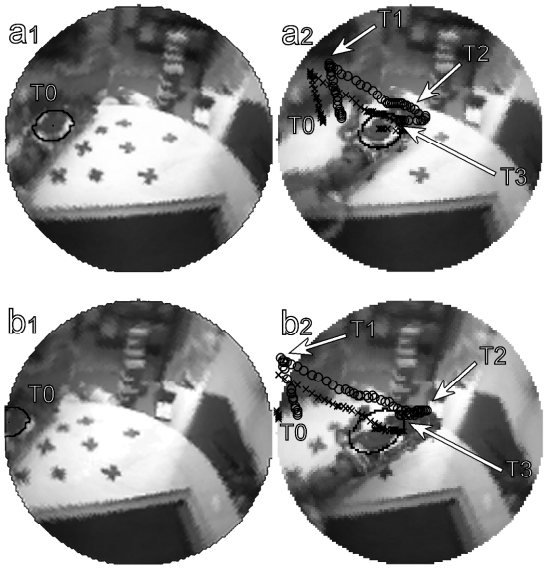

Figure 11. Arm trajectories for two reaching actions (a) and (b). T0 marks the position of the hand at the beginning of the action. Crosses correspond to the position of the palm; circles show the position of the fingers. The action is divided in three phases. From T0 to T1 arm prepositioning. From T1 to T2, reaching: in this case the motor-motor map is used to move the palm towards the center of the visual field (the target). A small adjustment with the arm Jacobian is performed to position the fingers on the target (T2 to T3).

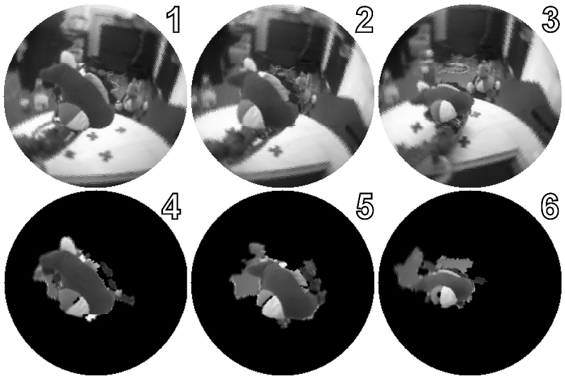

Figure 12. Object exploration and corresponding blobs (1-3 and 4-6 respectively). The blobs used in training the object model are the central and the adjacent ones. An example of the resulting segmentation is reported in Figure 13. Notice that fixation is maintained on the object by using the hand localization module (see text).

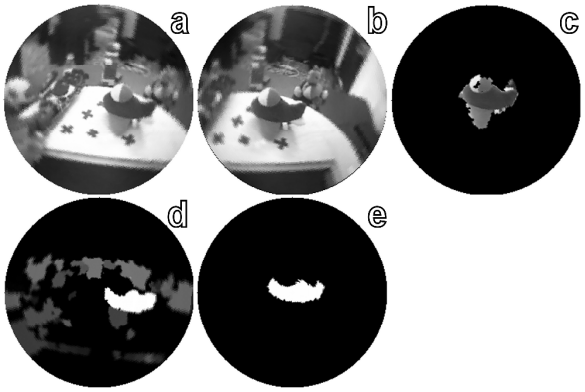

Figure 13. Visual search. The robot has acquired a model of the airplane toy during an exploration phase (not shown); this information primes the attention system. The blue blob at the center of the airplane is selected and a saccade performed. (a) and (b) show the visual scene before and after the saccade. (d) and (e) show the output of the visual attention system synchronized with (a) and (b) respectively. The result of the segmentation after the saccade is in (c).

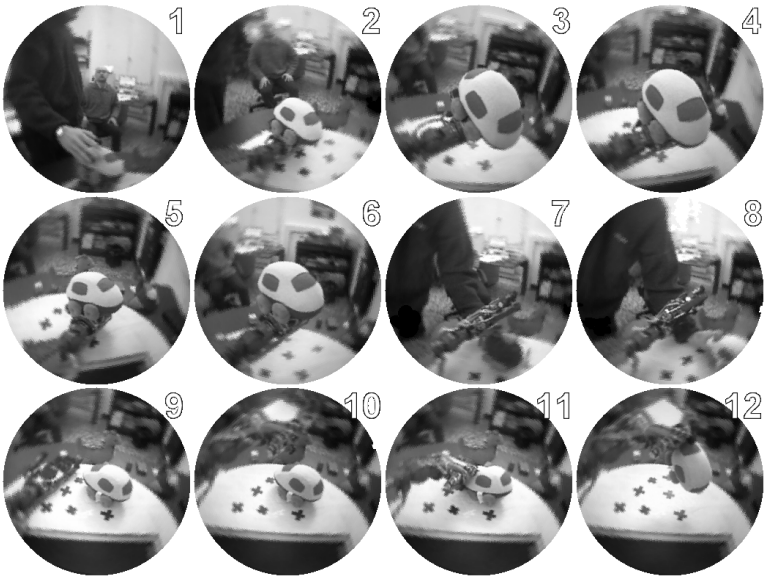

Figure 14. A sequence of the robot grasping an object. The action starts when an object is placed on the palm (1). The robot grasps the object and moves the eyes to fixate the hand (2). The exploration starts in (3) when the robot brings the object close to the camera. The object is moved in four different positions while maintaining fixation; at the same time the object model is trained (3-6). The robot drops the object and starts searching for it (7). The object is identified and a saccade performed (7-9). The robot eventually grasps the toy (10-12).

Table 1. Performance of the recognition system measured from a set of 50 trials.

Object	Recognition rate	Number of saccades when recognized
Toy car	94%	3.19±2.17
Toy airplane	88%	3.02±2.84