Teaching a Robotic Child - Machine Learning Strategies for a Humanoid Robot from Social Interactions

Artur Arsenio
Massachussets Institute of Technology[1]
USA

1. Introduction

Children love toys. Human caregivers often employ learning aids, such as books, educational videos, drawing boards, musical or textured toys, to teach a child. These social interactions provide a rich plethora of information to a child, and hence they should be extrapolated to a humanoid robot as well (Arsenio, 2004d).

Inspired in infant development, we aim at developing a humanoid robot's perceptual system through the use of learning aids, cognitive artifacts, and educational activities, so that a robot learns about the world according to a child's developmental phases (Arsenio, 2004c). Of course, the human caregiver plays a very important role on a robot's learning process (as it is so with children), performing educational and play activities with the robot (such as drawing, painting or playing with a toy train on a railway), facilitating robot's perception and learning. The goal is for the humanoid robot Cog (Figure 1) to see the world through the caregiver's eyes.

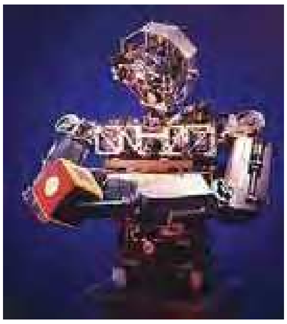


Fig. 1. The Humanoid Robot Cog.

This chapter will begin by addressing learning scenarios on section 2, which are employed to enable the robot to acquire input data in real-time to train learning algorithms. Tools developed to acquire such data, such as object / scene /texture segmentation (Arsenio,

[1] Research work was developed while the author was at MIT. The author is currently working at Siemens.

Source: Humanoid Robots, New Developments, Book edited by: Armando Carlos de Pina Filho
ISBN 978-3-902613-02-8, pp.582, I-Tech, Vienna, Austria, June 2007

2004e), sound / face segmentation (Arsenio, 2004f), object / face / hand tracking (Arsenio, 2004e;f) or even robot actuation for active perception (Arsenio, 2003; Metta & Fitzpatrick, 2003) are described in the literature.

We do not treat children as machines, i.e., automatons. But this automaton view is still widely employed in industry to build robots. Building robots involves indeed the hardware setup of sensors, actuators, metal parts, cables, processing boards, as well as software development. Such engineering might be viewed as the robot genotype. But equally important in a child is the developmental acquisition of information in a social and cultural context (Vigotsky, 1962).

Therefore, for a humanoid robot to interact effectively in its surrounding world, it must be able to learn. Section 3 presents learning strategies applied to a diverse set of problems, so that the robot learns information about objects, scenes, people and actions. Training data for the algorithms is generated on-line, in real-time, while the robot is in operation. We will describe the development of these learning mechanisms, presenting statistics from a large plethora of experimental results.

We aim at introducing robots into our society and treating them as us, using child development as a metaphor for developmental learning of a humanoid robot

2. Children-like Social Interactions as Learning Scenarios for a Humanoid

An autonomous robot needs to be able to acquire and incrementally assimilate new information, to be capable of developing and adapting to its environment. The field of machine learning offers many powerful algorithms, but these often require off-line, manually inserted training data to operate. Infant development research suggests ways to acquire such training data from simple contexts, and use these experiences to bootstrap to more complex contexts. We need to identify situations that enable the robot to temporarily reach beyond its current perceptual abilities (Figure 2 shows a set of such situations), giving the opportunity for development to occur (Arsenio, 2004c;d; Metta & Fitzpatrick, 2003).

This led us to create children-like learning scenarios for teaching a humanoid robot. These learning experiments are used for transmitting information to the humanoid robot Cog to learn about objects' multiple visual and auditory representations from books, other learning aids, musical instruments and educational activities such as drawing and painting.

Our strategy relies heavily in human-robot interactions. For instance, it is essential to have a human in the loop to introduce objects from a book to the robot (as a human caregiver does to a child). A more rich, complete human-robot communication interface results from adding other aiding tools to the robot's portfolio (which facilitate as well the children' learning process).

This is achieved by selectively attending to the human actuator (hand or finger). Indeed, primates have specific brain areas to process the hand visual appearance (Perrett et al., 1990). Inspired by human development studies, emphasis will be placed on facilitating perception through the action of a human instructor, and on developing machine-learning strategies that receive input from these human-robot interactions (Arsenio, 2004a,d).

Multi-modal object properties are learned using these children educational tools, and inserted into several recognition schemes, which are then applied to developmentally acquire new object representations (Arsenio, 2004c).

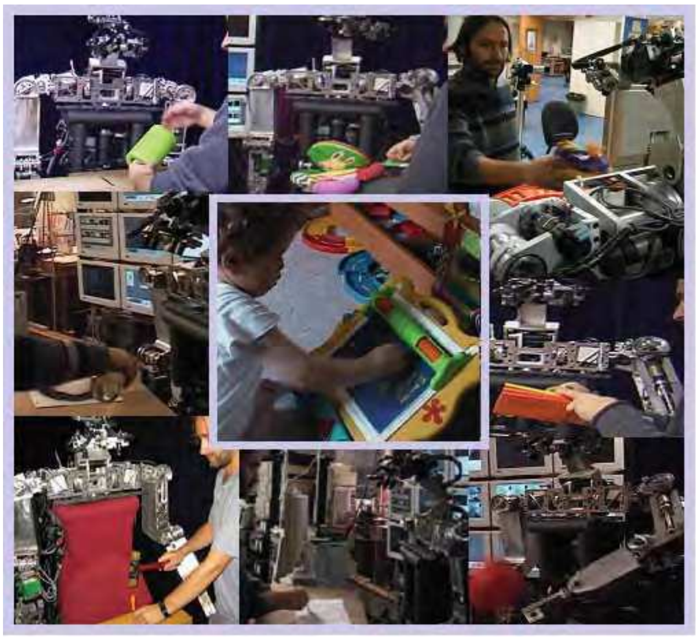


Fig. 2. Cog learning from several social interactions.

2.1 Robot Skill Augmentation through Cognitive Artifacts

A human caregiver can introduce a robot to a rich world of visual information concerning objects' visual appearance and shape. But cognitive artifacts, which enhance perception, can also be applied to improve perception over other perceptual modalities, such as auditory processing.

We exploit repetition (rhythmic motion, repeated sounds) to achieve segmentation and recognition across multiple senses (Arsenio, 2004d;f). We are interested in detecting conditions that repeat with some roughly constant rate, where that rate is consistent with what a human can easily produce and perceive. This is not a very well defined range, but we will consider anything above 10Hz to be too fast, and anything below 0.1Hz to be too slow. Repetitive signals in this range are considered to be events in our system: waving a flag is an event, but the vibration of a violin string is not an event (too fast), and neither is the daily rise and fall of the sun (too slow). Such a restriction is related to the idea of natural kinds, where perception is based on the physical dimensions and practical interests of the observer.

Abrupt motions, such as a poking movement, which involve large variations of movement, are also used to extract percepts (Arsenio, 2003; Arsenio 2004c;d).

Teaching from Books

Learning aids are often used by human caregivers to introduce the child to a diverse set of (in)animate objects, exposing the latter to an outside world of colors, forms, shapes and contrasts, that otherwise might not be available to a child (such as images of whales and cows). Since these learning aids help to expand the child's knowledge of the world, they are a potentially useful tool for introducing new informative percepts to a robot.

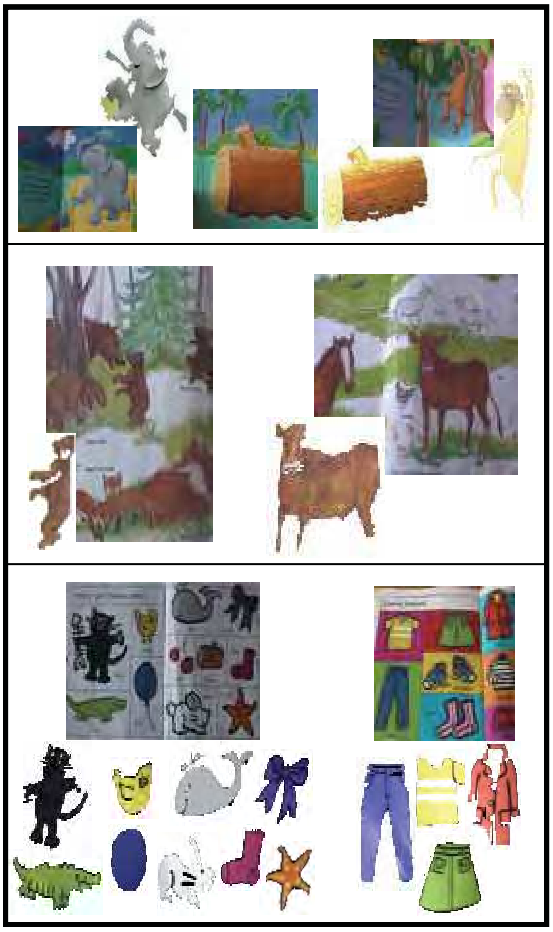


Fig. 3. Object templates extracted from books.

Children's learning is hence aided by the use of audiovisuals, and especially books, during social interactions with their mother or caregiver. Indeed, humans often paint, draw or just read books to children during their childhood. Books are also a useful tool to teach robots different object representations and to communicate properties of unknown objects to them.

Figure 3 shows images of object templates extracted from books using an active object segmentation algorithm – active in the sense that a human points to the object in a book with his/her finger (Arsenio, 2004e). This human aided perceptual grouping algorithm extracts informative percepts from picture books (fabric, foam or cardboard books), by correlating such information with a periodically moving human actuator (finger), resulting on signal samples for objects' image templates. Whenever the interacting human makes repetitive sounds simultaneously, object sound signatures, as well as cross-modal signatures, are segmented as well (Fitzpatrick & Arsenio, 2004). This data is employed afterwards as inputs for learning (section 3).

Matching Geometric Patterns: Drawings, Paintings, Pictures ...

Object descriptions may come in different formats - drawings, paintings, photos, *et cetera*. Hence, the link between an object representation in a book and real objects recognized from the surrounding world can be established through object recognition. Objects will be recognized using geometric hashing (section 3), a widely used recognition technique. The algorithm operates on three different set of features: chrominance and luminance topological regions, and shape (determined by an object's edges), as shown by Figure 4. Except for a description contained in a book, which was previously segmented, the robot had no other knowledge concerning the visual appearance or shape of such object.

Additional possibilities include linking different object descriptions in a book, such as a drawing, as demonstrated by two samples of the experimental results presented in Figure 4. A sketch of an object contains salient features concerning its shape, and therefore there are advantages in learning, and linking, these different representations. This framework is also a useful tool for linking other object descriptions in a book, such as a photo, a painting, or a printing (Arsenio, 2004a;d).

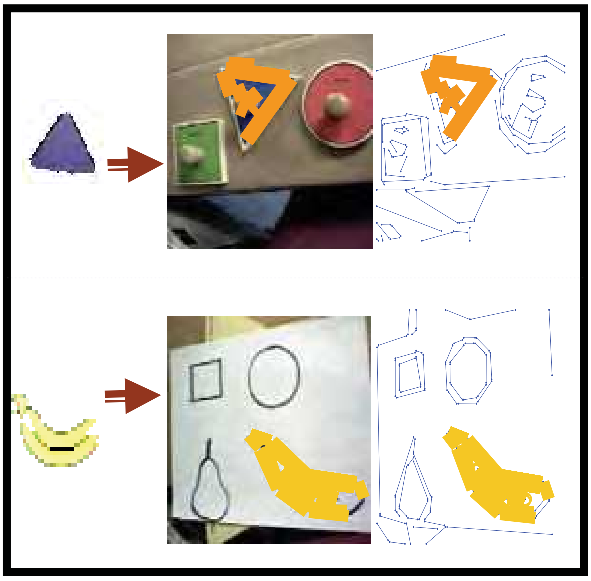


Fig. 4. Matching objects from books to real world objects and drawings.

Explorations into the World of Tools and Toys

A plethora of other educational tools and toys are widely used by educators to teach children, helping them to develop. Examples of such tools are toys (such as drawing boards), educational TV programs or educational videos. The Baby Einstein collection includes videos to introduce infants and toddlers to colors, music, literature and art. Famous painters and their artistic creations are displayed to children on the Baby Van Gogh video, from the mentioned collection. This inspired the design of learning experiments in which Cog is introduced to art using an artificial display (the computer monitor), as shown in Figure 5 (Arsenio, 2004d).

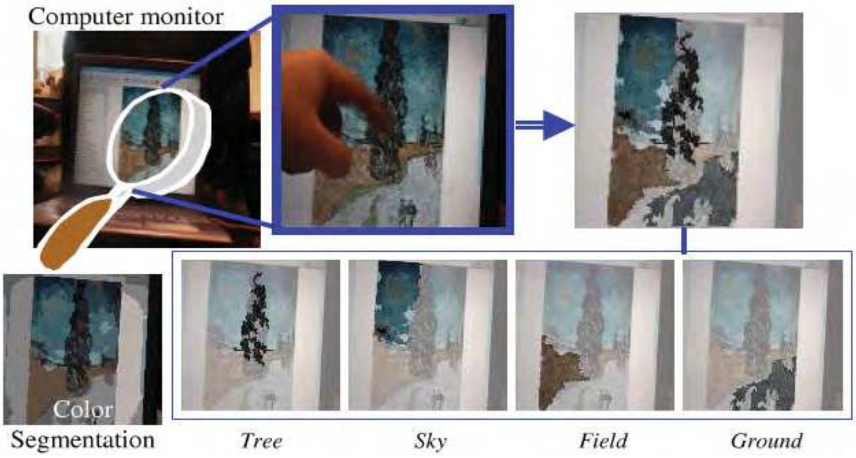


Fig. 5. The image of a painting by Vincent Van Gogh, Road with Cypress and Star, 1890 is displayed on a computer screen. Paintings are contextually different than pictures or photos, since the painter style changes the elements on the figure considerably. Van Gogh, a post-impressionist, painted with an aggressive use of brush strokes. But individual painting elements can still be grouped together by having a human actor tapping on their representation in the computer screen to group them together.

Learning First Words

Auditory processing is also integrated with visual processing to extract the name and properties of objects. However, hand visual trajectory properties and sound properties might be independent - while tapping on books, it is not the interacting human caregiver hand that generates sound, but the caregiver vocal system pronouncing sounds such as the object's name. Therefore, cross-modal events are associated together under a weak requirement: visual segmentations from periodic signals and sound segmentations are bound together if occurring temporally close (Fitzpatrick & Arsenio, 2004). This strategy is also well suited for sound patterns correlated with the hand visual trajectory (such as playing musical tones by shaking a rattle).

2.2 Educational, Learning Activities

A common pattern of early human-child interactive communication is through activities that stimulate the child's brain, such as drawing or painting. Children are able to extract information from such activities while they are being performed on-line. This capability motivated the implementation of three parallel processes which receive input data from three different sources: from an attentional tracker, which tracks the robot's attentional focus, and it is attracted to a new salient stimulus; from a multi-target tracking algorithm implemented to track simultaneously multiple targets; and from an algorithm that selectively attends to the human actuator (Arsenio, 2004d).

Learning Hand Gestures

Standard hand gesture recognition algorithms require an annotated database of hand gestures, built off-line. Common approaches, such as Space-Time Gestures (Darrel & Pentland, 1993), rely on dynamic programming. Others (Cutler & Turk, 1998) developed systems for children to interact with lifelike characters and play virtual instruments by classifying optical flow measurements. Other classification techniques include state machines, dynamic time warping or Hidden Markov Models (HMMs).

We follow a fundamentally different approach, being periodic hand trajectories mapped into geometric descriptions of objects, to classify simple circular or triangular movements, for instance (Arsenio, 2004a). Figure 6b reports an experiment in which a human draws repetitively a geometric shape on a sheet of paper with a pen. The robot learns what was drawn by matching one period of the hand gesture to the previously learned shape (the hand gesture is recognized as circular in the Figure). Hence, the geometry of periodic hand trajectories is recognized in real-time to the geometry of objects in an object database, instead of being mapped to a database of annotated gestures.

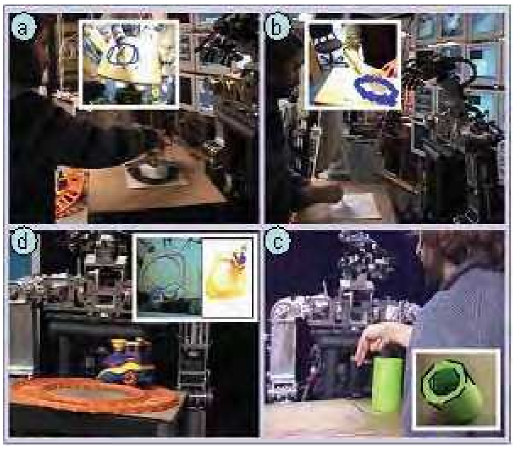


Fig. 6. Sample of experiments for object and shape recognition from hand gestures.

Object Recognition from Hand Gestures.

The problem of recognizing objects in a scene can be framed as the dual version of the hand gestures recognition problem. Instead of using previously learned object geometries to recognize hand gestures, hand gestures' trajectories are now applied to recover the geometric shape (set of lines computed by applying the Hough Transform) and appearance (given by an image template enclosing such lines) of a scene object (as seen by the robot). Visual geometries in a scene (such as circles) are recognized as such from hand gestures having the same geometry (as is the case of circular gestures). Figure 6a shows results for such task on an experiment in which an interacting human paints a circle. The robot learns what was painted (a circle) by matching the hand gesture to the shape defined by the ink on the paper. The algorithm is useful to identify shapes from drawing, painting or other educational activities (Arsenio, 2004d).

Shape from Human Cues

A very similar framework is applied to extract object boundaries from human cues. Indeed, human manipulation provides the robot with extra perceptual information concerning objects, by actively describing (using human arm/ hand/finger trajectories) object contours or the hollow parts of objects, such as a cup (see experiment with green cup in Figure 6c). Tactile perception of objects from the robot grasping activities has been actively pursued –

see for instance (Polana & Nelson, 1994; Rao et al., 1989). Although more precise, these techniques require hybrid position/ force control of the robot's manipulator end-effector so as not to damage or break objects.

Functional Constraints

Not only hand gestures can be used to detect interesting geometric shapes in the world as seen by the robot. For instance, certain toys, such as trains, move periodically on rail tracks, with a functional constraint fixed both in time and space. Therefore, one might obtain information concerning the rail tracks by observing the train's visual trajectory. To accomplish such goal, objects are visually tracked by an attentional tracker which is modulated by an attentional system (Arsenio, 2004d). The algorithm starts by masking the input world image to regions inside the moving object's visual trajectory (or outside but on a boundary neighborhood). Lines modeling the object's trajectory are then mapped into lines fitting the scene edges. The output is the geometry of the stationary object which is imposing the functional constraint on the moving object. Figure 6d shows as well an experiment for the specific case of extracting templates for rail tracks from the train's motion (which is constrained by the railway circular geometry).

2.3 Learning about People

Faces in cluttered scenes are located by a computationally efficient algorithm (Viola & Jones, 2001), which is applied to each video frame (acquired by a foveal camera). If a face is detected, the algorithm estimates a window containing that face, as shown in Figure 7. The novelty here consists on acquiring a large amount of samples of training data in real-time using a multi-object tracking algorithm (Arsenio, 2004d), which allows to group several image templates together - from different views of the same tracked face – into the same group.

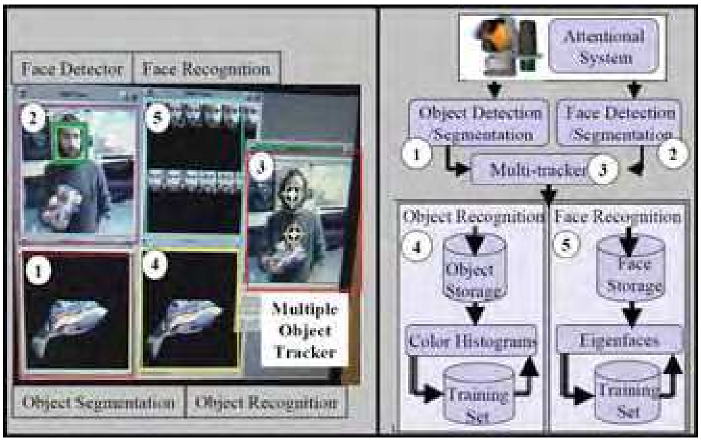


Fig. 7. Approach for segmenting and recognizing faces & objects. Training data for object/face recognition is extracted by keeping objects and others faces in memory for a while, generating a collection of training samples consisting of multiple segmentations of objects and faces. (left) on-line experiment on Cog (right) schematic organization. 1. Object segmentation 2. Face detection and segmentation 3. Multiple object tracking 4. Object Recognition 5. Face Recognition.

3. Machine Learning Algorithms

This section presents a collection of machine learning algorithms and methodologies implemented to emulate different cognitive capabilities on the humanoid robot Cog. This framework is effectively applied to a collection of AI, computer vision, and signal processing problems. It is shown to solve a broad spectrum of machine learning problems along a large categorical scope: actions, objects, scenes and people, using real-time data acquired from human-robot interactions as described by last section's learning scenarios.

Most of the learning algorithms introduced receive as training inputs information produced by other modules, such as object segmentation (Arsenio, 2004e). But all this training data is automatically annotated on-line, in real-time, instead of the standard off-line, manual annotation. This is accomplished by having human caregivers introducing new percepts to the robot (Arsenio, 2004a). Hence, we are motivated by cognitive development of human infants, which is bootstrapped by the helping hand that a human caregiver (and especially the infant's mother) provides to the infant (Arsenio, 2004c).

This chapter does not intend to propose new learning algorithms. Instead, the focus is placed on using existing learning algorithms to solve a wide range of problems.

One essential capability for a humanoid robot to achieve convincing levels of competency is object recognition. Therefore, a learning algorithm operating on object color histograms is first presented. A more robust algorithm employing geometric hashing techniques is also described. These algorithms are not however appropriate to tackle the face recognition problem, which is solved using the eigenfaces method. A similar eigenobjects based algorithm will be applied as well for sound recognition, using eigensounds.

Recognition of scenes is especially important for robot localization. An approach based on a contextual description of the scene envelope is also described. Contextual descriptions of a scene are modeled by a Mixture of Gaussians, being the parameters of such mixture estimated iteratively using the Expectation-Maximization algorithm.

The processing of cross-modal information, among different sensorial capabilities, leads to an innovative cross-modal object recognition scheme using a Dynamic Programming approach. Contextual features provide another source of information very useful to recognize objects – we apply a method similar to a mixture of experts: weighted cluster modeling. Another technique employed is Back-propagation Neural Networks for activities' identification (and for identifying the function of an object within an activity). This learning method is also shown to increase sound recognition rates compared to the eigensounds method. Both qualitative and quantitative experimental results are evaluated and discussed for each algorithm. Section 3 ends by referring briefly other learning strategies employed by the humanoid robot Cog, applied not only for perception but also for robot control by learning the underlying dynamic models.

3.1 Color Histograms

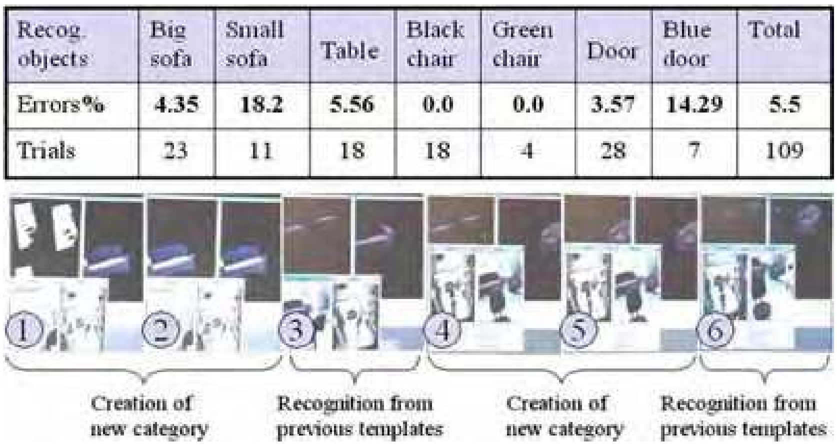
The object recognition algorithm needs to cluster object templates by classes according to their identity. Such task was implemented through color histograms – objects are classified based on the relative distribution of their color pixels. Since object masks are available from segmentation (Arsenio, 2004e), external global features do not affect recognition, and hence color histograms are appropriate. A multi-target tracking algorithm (Arsenio, 2004d) keeps track of object locations as the visual percepts change due to movement of the robot's active head. Ideally, a human actor should expose the robot to several views of the object being tracked (if the object appearance is view-dependent), in order to link them to the same object. This way, a collection of object views becomes available as input.

Recognition works as follows. Quantization of each of the three color channels originates 8^3 groups G_i of similar colors. The number of image pixels n_{Gi} indexed to a group is stored as a percentage of the total number of pixels. The first 20 color histograms of an object category are saved into memory and updated thereafter. New object templates are classified according to their similarity with other object templates previously recognized for all object categories, by computing:

$$p = \sum_{i=1}^{8^3} \text{minimum}\,(n_{Gi}, n_{G'i})$$

If p < th (th set to 0,7) for all of the 20 histograms in an object category, then the object does not belong to that category. If this happens for all categories, then it is a new object. If p ≥ th, then a match occurs, and the object is assigned to the category with maximum p.

Whenever an object is recognized into a given category, the average color histogram which originated a better match is updated. Given an average histogram which is the result of averaging m color histograms, the updating consists of computing the weighted average between this histogram (weight m) and the new color histograms (unit weight). This has the advantage that color histograms evolve as more samples are obtained to represent different views of an object.



Recog. objects	Big sofa	Small sofa	Table	Black chair	Green chair	Door	Blue door	Total
Errors%	**4.35**	**18.2**	**5.56**	**0.0**	**0.0**	**3.57**	**14.29**	**5.5**
Trials	23	11	18	18	4	28	7	109

1
2
3
4
5
6
Creation of new category
Recognition from previous templates
Creation of new category
Recognition from previous templates

Fig. 8. (top) Recognition errors. Matches evaluated from a total of 11 scenes (objects are segmented and recognized more than once per scene). (bottom) Sequence from an on-line experiment of several minutes on the humanoid robot Cog: (1) The robot detects and segments a new object – a sofa; (2) New object is correctly assigned to a new category; (3) Object, not being tracked, is recognized from previous templates (as shown by the two sofa templates mapped to it); (4-5-6) Same sequence for a different, smaller sofa.

Experimental Results for Template Matching

Figure 8 presents quantitative performance statistics. It shows a sample of the system running on the humanoid robot Cog, while recognizing previously learned objects. Incorrect matches occurred due to color similarity among different objects (such as a big and a small sofa). Errors arising from labeling an object in the database as a new object are chiefly due to drastic variations in light sources. Qualitative results from an on-line experiment of several minutes for object segmentation, tracking and recognition of new objects on the humanoid robot are also shown.

Out of around 100 samples from on-line experiments, recognition accuracy average was of 95%. Several experiments have shown however the algorithm not capable to differentiate among people's faces, although it differentiated correctly between faces and other objects.

3.2 Geometric Hashing

This object recognition algorithm consists of three independent algorithms. Each recognizer operates along orthogonal directions to the others over the input space (Arsenio, 2004b). This approach offers the possibility of priming specific information such as searching for a specific object feature (color, shape or luminance) independently of the others. The set of input features are:

- Color: groups of connected regions with similar color
- Luminance: groups of connected regions with similar luminance
- Shape. A Hough transform is applied to a contour image (from a Canny edge detector). Line orientation is determined using Sobel masks. Pairs of oriented lines are then used as input features

Geometric hashing (Wolfson & Rigoutsos, 1997) is a rather useful technique for high-speed performance. In this method, quasi-invariants are computed from training data in model images, and then stored in hash tables. Recognition consists of accessing and counting the contents of hash buckets. Recognition of objects has to occur over a variety of scene contexts. An adaptive Hash table (a hash table with variable-size buckets) algorithm was implemented to store affine color, luminance and shape invariants (which are view independent for small perspective deformations). Figure 4 displays two results from applying this algorithm using shape features (Arsenio, 2004d).

3.3 Eigenobjects – Principal Component Analysis

Component Analysis (PCA) is an efficient method to describe a collection of images. The corresponding eigenvectors are denominated eigenobjects.

Let the training set of M images from an object n be $\{\phi_1, \phi_2, . . . \phi_M\}$ (see Figure 9). The average image of this set is defined by

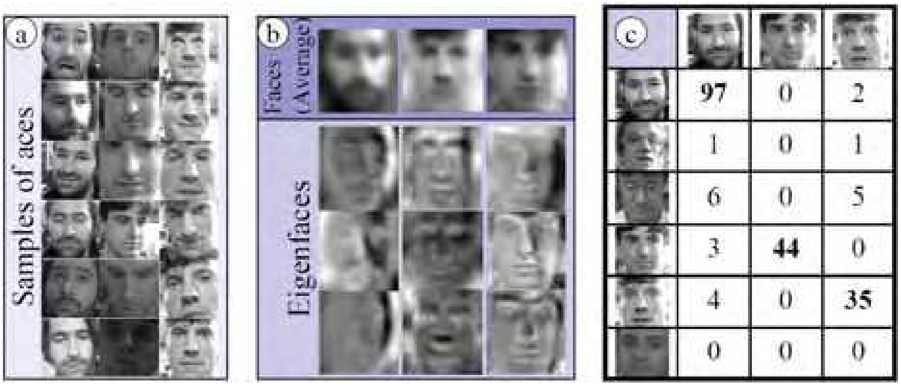
$\psi = 1/M \sum_{i=1}^{M} \phi_i$. The covariance matrix for the set of training objects is thus given by (1):

$$C\phi_n = \sum_{i=1}^{M} \Gamma_i \Gamma_i^T = AA^T \quad (1)$$

being $\Gamma_n = \phi_n - \psi$ the difference of each image from the mean, and $A = [\Gamma_1, \Gamma_2, . . . , \Gamma_M]$.

Cropped faces are first rescaled to 128 × 128 images (size $S = 128^2$). Determining the eigenvectors and eigenvalues of the S^2 size covariance matrix C is untractable. However, C rank does not exceed $M - 1$. For $M < S^2$ there are only $M - 1$ eigenvectors associated to non-zero eigenvalues, rather than S^2. Let v_i be the eigenvectors of the M × M matrix $A^T A$. The eigenfaces μ_i are given by:

$$\mu_i = \sum_{k=1}^{M} v_{ik}\Gamma_k \qquad i = 1, \ldots M \tag{2}$$



	97	0	2
	1	0	1
	6	0	5
	3	44	0
	4	0	35
	0	0	0

Fig. 9. a) Face image samples are shown for each of three people out of a database of six; b) Average face image for three people on the database, together with three eigenfaces for each one; c) Confusion table with face recognition results.

The number of basis functions is further reduced from M to M′ by selecting only the most meaningful M′ eigenvectors (with the largest associated eigenvalues), and ignoring all the others. Classification of the image of object φ consists of projecting it into the eigenobject components, by correlating the eigenvectors with it, for obtaining the coefficients $w_i=\mu_i(\phi-\psi)$, i= 1,..., M′ of this projection. The weights w_i form a vector $\Omega = \{w_1, w_2, \ldots, w_{M'}\}$. An object is then classified by selecting the minimum L_2 distance to each object's coefficients in the database $\varepsilon_\phi = \left\| \Omega - \Omega_k \right\|$ where Ω_k describes the kth object class in the database. If ε_ϕ is below a threshold, then it corresponds to a new object.

Eigenfaces: Experimental Results

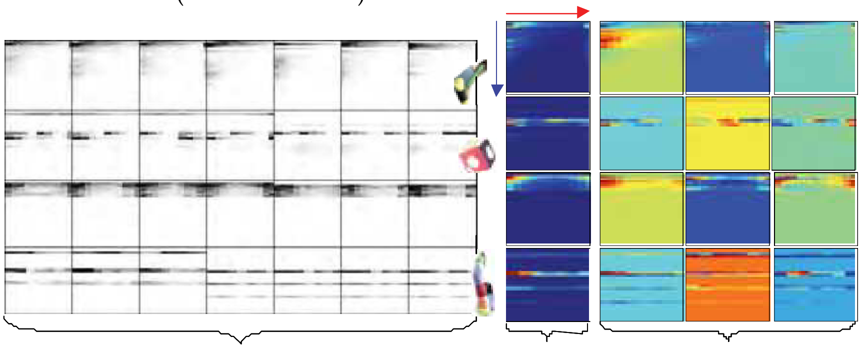
The eigenvectors are now denominated eigenfaces (Turk & Pentland, 1991), because they are face-like in appearance (see Figure 9 - the confusion table in this Figure presents results for recognizing three different people, being the average recognition accuracy of 88.9%). The training data set contains a lot of variation. Validation data corresponds to a random 20% of all the data.

Eigensounds for Sound Recognition

A collection of annotated acoustic signatures for each object are used as input data (see Figure 10) for a sound recognition algorithm by applying the eigenobjects method. A sound image is represented as a linear combination of base sound signatures (or *eigensounds*). Classification consists of projecting novel sounds to this space, determining the coefficients of this projection, computing the L_2 distance to each object's coefficients in the database, and selecting the class corresponding to the minimum distance.

Cross-modal information aids the acquisition and learning of unimodal percepts and consequent categorization in a child's early infancy. Similarly, visual data is employed here to guide the annotation of auditory data to implement a sound recognition algorithm. Training samples for the sound recognition algorithm are classified into different categories by the visual object recognition system or from information from the visual object tracking system. This enables the system, after training, to classify the sounds of objects not visible.

The system was evaluated quantitatively by random selection of 10% of the segmented data for validation, and the remaining data for training. This process was randomly repeated three times. It is worth noticing that even samples received within a short time of each other often do not look too similar, due to background acoustic noise, noise on the segmentation process, other objects' sounds during experiments, and variability on how objects are moved and presented to the robot. For example, the car object is heard both alone and with a rattle (either visible or hidden). The recognition rate for the three runs averaged to 82% (86.7%, 80% and 80%). Recognition rates by object category were: 67% for the car, 91.7% for the cube rattle, 77.8% for the snake rattle and 83.3% for the hammer. Most errors arise from mismatches between (car and hammer) sounds.



1) 7 random sound samples for each of 4 objects. From top to bottom: hammer, cube rattle, car and snake rattle, respectively

2) Average sound images

3) Eigenobjects corresponding to the three highest eigenvalues

Fig. 10. Sound recognition. Acoustic signatures for four objects are shown along the rows. (1) Seven sound segmentation samples are shown for each object, from a total of 28 (car), 49 (cube rattle), 23 (snake rattle) and 34 (hammer) samples. (2) Average acoustic signatures. The vertical axis corresponds to the frequency bands and the horizontal axis to time normalized by the period. (3) Eigensounds corresponding to the three highest eigenvalues. The repetitive nature of the sound generated by an object under periodic motion can be analyzed to extract an acoustic signature for that object. We search for repetition in a set of frequency bands, collecting those whose energies oscillate together with a similar period (Fitzpatrick & Arsenio, 2004).

3.4 Mixture of Gaussians for Scene Recognition

Wavelets (Strang & Nguyen, 1996) are employed to extract contextual features. Processing is applied iteratively through the low frequency branch of the wavelet transform over T=5 scales, while higher frequencies along the vertical, horizontal and diagonal orientations are stored (due to signal polarity, this corresponds to a compact representation of six orientations in three images). The input is thus represented by v(x, y) $= v(\vec{p})$ =$\{v_k(x, y), k=1,..., N\}$, with N=3T=15. Each wavelet component at the ith level has dimensions $256/2^i \times 256/2^i$, and is down-sampled to an 8 × 8 image:

$$\bar{v}(x,y) = \sum_{i,j} \mathrm{v}(\mathrm{i},\mathrm{j})\mathrm{h}(\mathrm{i}-\mathrm{x},\mathrm{j}-\mathrm{y}) \qquad (3)$$

where h(x,y) is a Gaussian window. Thus, $\bar{v}(x, y)$ has dimension 960. Similarly to other approaches (Torralba, 2003), the dimensionality problem is reduced to become tractable by applying Principal Component Analysis (PCA). The image features $\bar{v}(\vec{p})$ are decomposed into the basis functions given by the PCA:

$$v(\vec{p}) = \sum_{i=1}^{D} c_i \varphi_k^i(\vec{p}) \text{ , } c_i = \sum_{\vec{p},k} v_k(\vec{p}) \varphi_k^i(\vec{p}) \tag{4}$$

where the functions $\varphi_k^i(\vec{p})$ are the eigenfunctions of the covariance operator given by $v_k(\vec{p})$. These functions incorporate both spatial and spectral information. The decomposition coefficients are obtained by projecting the image features $v_k(\vec{p})$ into the principal components c$_i$, used hereafter as input context features.

The vector $\vec{c}$ ={c$_i$, i= 1, . . . , D} denotes the resulting D-dimensional input vector, with D=E$_m$, 2≤ D≤ Th$_o$, where m denotes a class, Th$_o$ an upper threshold and E$_m$ denotes the number of eigenvalues within 5% of the maximum eigenvalue. These features can be viewed as a scene's holistic (Oliva & Torralba, 2001) representation since all the regions of the image contribute to all the coefficients, as objects are not encoded individually. The effect of neglecting local features is reduced by mapping the foveal camera (which grabs data for the object recognition scheme based on local features) into the image from the peripheral view camera, where the weight of the local features $\vec{v}_I$ is strongly attenuated. The vector $\vec{p}$ is thus given in wide field of view retinal coordinates.

A collection of images is automatically annotated by the robot (Arsenio, 2004b;c) and used as training data. Mixture models are applied to find interesting places to put a bounded number of local kernels that can model large neighborhoods. In D-dimensions a mixture model is denoted by density factorization over multivariate Gaussians (spherical Gaussians were selected for faster processing times), for each object class n:

$$p(\vec{c} \mid o_n) = \sum_{m=1}^{M} b_m G(\vec{c}, \vec{\mu}_{m,n}, C_{m,n})$$

where G$_m$ refers to the mth Gaussian with mean $\vec{\mu}_m$ and covariance matrix C$_m$, M is the number of Gaussian clusters, and b$_m$= p(G$_m$) are the weights of the local models. The estimation of the parameters will follow the EM algorithm (Gershenfeld, 1999):

- **E-step for** k-**iteration**: From the observed data $\vec{c}$, compute the a-posteriori cluster probability $e_{m,n}^k(l)$:

$$e_{m,n}^k(l) = p(c_{m,n} \mid \vec{c}) = \frac{b_{m,n}^k G(\vec{c}, \vec{\mu}_{m,n}^k, C_{m,n}^k)}{\sum_{m=1}^{M} b_{m,n}^k G(\vec{c}, \vec{\mu}_{m,n}^k, C_{m,n}^k)} \tag{5}$$

- **M-step for** k-**iteration**: cluster parameters are estimated according to the maximization of the join likelihood of the L training data samples

$$b_{m,n}^{k+1} = \sum_{l=1}^{L} e_{m,n}^{k} \text{ , } \vec{\mu}_{m,n}^{k+1} = \frac{\sum_{l=1}^{L} e_{m,n}^{k}(l)\vec{c}_l}{\sum_{l=1}^{L} e_{m,n}^{k}(l)} \tag{6}$$

$$C_{m,n}^{k+1} = \frac{\sum_{l=1}^{L} e_{m,n}^{k}(l)(\vec{c}_l - \vec{\mu}_{m,n}^{k+1})(\vec{c}_l - \vec{\mu}_{m,n}^{k+1})^T}{\sum_{l=1}^{L} e_{m,n}^{k}(l)} \tag{7}$$

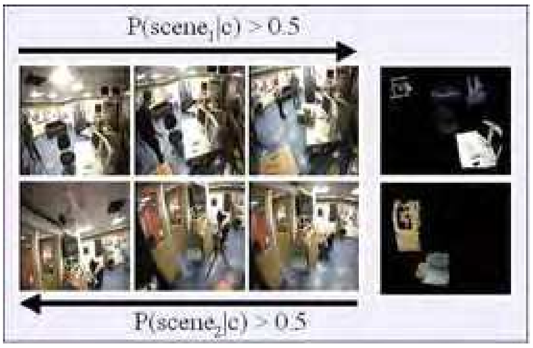


Fig. 11 Test images (wide field of view) organized with respect to $p(o_n \mid \vec{c})$. Top row: o$_n$=scene$_1$, $p(scene_1 \mid \vec{c}) > 0.5$; Bottom row: o$_n$=scene$_2$, $p(scene_2 \mid \vec{c}) > 0.5$. Scene descriptions shown in the right column are built on-line, automatically (Arsenio, 2004b;c).

The EM algorithm converges as soon as the cost gradient is small enough or a maximum number of iterations is reached. The probability density function (PDF) for an object n is then given by Bayes' rule $p(o_n \mid \vec{c}) = p(\vec{c} \mid o_n) p(o_n) / p(o_n)$ where $p(\vec{c}) = p(\vec{c} \mid o_n) p(o_n) + p(\vec{c} \mid \neg o_n) p(\neg o_n)$.

The same method applies for the out-of-class PDF $p(\vec{c} \mid \neg o_n)$ which represents the statistical feature distribution for the input data in which o$_n$ is not present.

Finally, it is necessary to select the number M of gaussian clusters. This number can be selected as the one that maximizes the join likelihood of the data. An agglomerative clustering approach based on the Rissanen Minimum Description Length (MDL) order identification criterion (Rissanen, 1983) was implemented to automatically estimate M (Figure 11 shows algorithm results for classifying two scenes).

3.5 Dynamic Programming for Recognition from Cross-Modal Cues

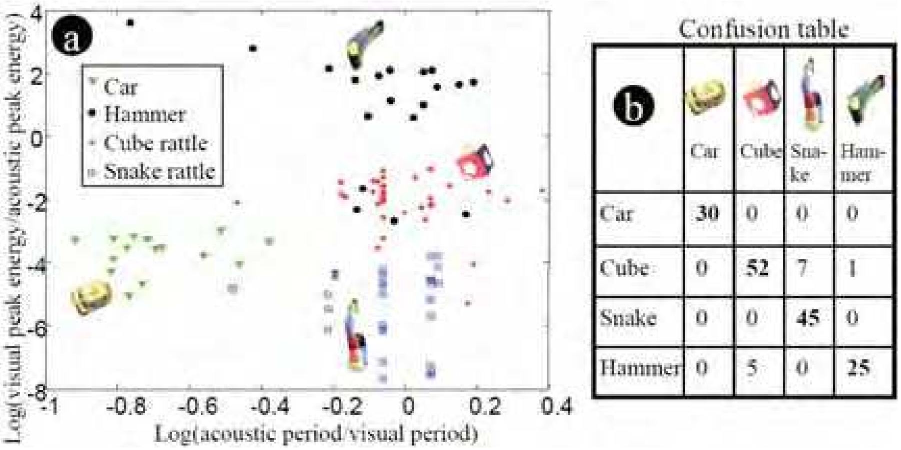
Different objects have distinct acoustic-visual patterns which are a rich source of information for object recognition, if we can recover them. The relationship between object motion and the sound generated varies in an object-specific way. A hammer causes sound after striking an object. A toy truck causes sound while moving rapidly with wheels spinning; it is quiet when changing direction. These statements are truly cross-modal in nature. Features extracted from the visual and acoustic segmentations are what is needed to build an object recognition system (Fitzpatrick & Arsenio, 2004). The feature space for recognition consists of:

- Sound/Visual period ratios – the sound energy of a hammer peaks once per visual period, while the sound energy of a car peaks twice.
- Visual/Sound peak energy ratios – the hammer upon impact creates high peaks of sound energy relative to the amplitude of the visual trajectory.

Dynamic programming is applied to match the sound energy to the visual trajectory signal. Formally, let S = (S_1, . . . , S_n) and V= (V_1, . . . , V_m) be sequences of sound and visual trajectory energies segmented from n and m periods of the sound and visual trajectory signals, respectively. Due to noise, n may be different to m. If the estimated sound period is half the visual one, then V corresponds to energies segmented with 2m half periods (given by the distance between maximum and minimum peaks). A matching path P = (P_1, . . . , P_l) defines an alignment between S and M, where max(m, n) ≤ l ≤ m + n − 1, and P_k=(i, j), a match *k* between sound cluster *j* and visual cluster *i*. The matching constraints are set by:

- **The boundary conditions:** P_1= (1, 1) and P_l= (m, n).
- **Temporal continuity:** P_{k+1}∈ {(i+1, j+1), (i+1, j), (i, j+1)}. Steps are adjacent elements of P.

The function cost $c_{i,j}$ is given by the square difference between V_i and S_j periods. The best matching path W can be found efficiently using dynamic programming, by incrementally building an m × n table caching the optimum cost at each table cell, together with the link corresponding to that optimum. The binding W will then result by tracing back through these links, as in the Viterbi algorithm.



b	Car	Cube	Snake	Hammer
Car	30	0	0	0
Cube	0	52	7	1
Snake	0	0	45	0
Hammer	0	5	0	25

Fig. 12. Object recognition from cross-modal clues. The feature space consists of period and peak energy ratios. The confusion matrix for a four-class recognition experiment is shown. The period ratio is enough to separate the cluster of the car object from all the others. Similarly, the snake rattle is very distinct, since it requires large visual trajectories for producing soft sounds. Errors for categorizing a hammer originated exclusively from erroneous matches with the cube rattle, because hammering is characterized by high energy ratios, and very soft bangs are hard to identify correctly. The cube rattle generates higher energy ratios than the snake rattle. False cube recognitions resulted mostly from samples with low energy ratios being mistaken for the snake.

Experimental Results: Figure 12 shows cross-modal features for a set of four objects. It would be hard to cluster automatically such data into groups for classification. But as in the sound recognition algorithm, training data is automatically annotated by visual recognition and tracking. After training, objects can be categorized from cross-modal cues alone. The system was evaluated by selecting randomly 10% of the data for validation, and the remaining data for training. This process was randomly repeated 15 times. The recognition rate averaged over all these runs were, by object category: 86.7%for the cube rattle, 100% for both the car and the snake rattle, and 83% for the hammer. The overall recognition rate was 92.1%. Such results demonstrate the potential for recognition using cross-modal cues.

3.6 Weighted Cluster Modeling

Objects in the world are situated, in the sense that they usually appear in specific places. Children are pretty good at learning the relative probability distribution of objects in a scene – for instance, chairs are most probable in front of desks, but not in a ceiling. The scene context puts a very important constraint on the type of places in which a certain object might be found. From a humanoid point of view, contextual selection of the attentional focus is very important both to constrain the search space for identifying or locating objects (optimizes computational resources) and also to determine common places on a scene to drop or store objects.

Therefore, it is important to develop a model for the contextual control of the attentional focus (location and orientation), scale selection and depth inference. The output space is defined by the 6-dimensional vector $\vec{x} = (\vec{p}, d, \vec{s}, \phi)$, where $\vec{p}$ is a 2D position vector, d is the object's depth (Arsenio, 2004b;c), $\vec{s}$ = (w, h) is a vector containing the principal components of the ellipse that models the 2D retinal size of the object, and ϕ is the orientation of such ellipse. Given the context $\vec{c}$, we need to evaluate the PDF $p(\vec{x} \mid o_n, \vec{c})$ from a mixture of (spherical) Gaussians (Gershenfeld, 1999),

$$p(\vec{x}, \vec{c} \mid o_n) = \sum_{m=1}^{M} b_{m,n} G(\vec{x}, \vec{\eta}_{m,n}, X_{m,n}) G(\vec{c}, \vec{\mu}^{k}_{m,n}, C^{k}_{m,n}) \tag{8}$$

The mean of the new Gaussian $G(\vec{x}, \vec{\eta}_{m,n}, X_{m,n})$ is now a function $\vec{\eta} = f(\vec{c}, \beta_{m,n})$, that depends on $\vec{c}$ and on a set of parameters $\beta_{m,n}$. A locally affine model was chosen for *f*, with $\{\beta_{m,n} = (\vec{a}_{m,n}, A_{i,n}) : \eta_{m,n} = \vec{a}_{m,n} + A^T \vec{c}\}$. The learning equations become now (Gershenfeld, 1999):

- **E-step for** k**-iteration**: From the observed data $\vec{c}$ and $\vec{x}$, compute the a-posteriori probabilities of the clusters:

$$e^{k}_{m,n}(l) = \frac{b^{k}_{m,n} G(\vec{x}, \vec{\eta}^{k}_{m,n}, X^{k}_{m,n}) G(\vec{c}, \vec{\mu}^{k}_{m,n}, C^{k}_{m,n})}{\sum_{m=1}^{M} b^{k}_{m,n} G(\vec{x}, \vec{\eta}^{k}_{m,n}, X^{k}_{m,n}) G(\vec{c}, \vec{\mu}^{k}_{m,n}, C^{k}_{m,n})}$$

- **M-step for** k**-iteration**: cluster parameters are estimated according to (where m indexes the M clusters, and l indexes the number L of samples):

$$C^{k+1}_{m,n} = < (\vec{c}_l - \vec{\mu}^{k+1}_{m,n})(\vec{c}_l - \vec{\mu}^{k+1}_{m,n})^T >_m \tag{9}$$

$$A_{m,n}^{k+1} = (C_{m,n}^{k+1})^{-1} < (\vec{c} - \vec{\mu})(\vec{x} - \vec{\mu})^T >_m \tag{10}$$

$$a_{m,n}^{k+1} =< (\vec{x} - (A_{m,n}^{k+1})^T \vec{c}) >_m \tag{11}$$

$$X_{m,n}^{k+1} =< (\vec{x} - \vec{a}_{m,n}^{k+1} - (A_{m,n}^{k+1})^T \vec{c})(\vec{x} - \vec{a}_{m,n}^{k+1} - (A_{m,n}^{k+1})^T \vec{c})^T >_m \tag{12}$$

All vectors are column vectors and <>m in (9) represents the weighted average with respect to the posterior probabilities of cluster m.

The parameters $b_{m,n}^k$ and means $\vec{\mu}_{m,n}^{k+1}$ are estimated as before. The conditional probability follows then from the joint PDF of the presence of an object o_n, at the spatial location *p*, with pose ϕ, size $\vec{s}$ and depth *d*, given a set of contextual image measurements $\vec{c}$:

$$p(\vec{x} \mid o_n, \vec{c}) = \frac{b_{m,n}^k G(\vec{x}, \vec{\eta}_{m,n}^k, X_{m,n}^k) G(\vec{c}, \vec{\mu}_{m,n}^k, C_{m,n}^k)}{\sum_{m=1}^{M} b_{m,n}^k G(\vec{c}, \vec{\mu}_{m,n}^k, C_{m,n}^k)}$$

Object detection and recognition requires the evaluation of this PDF at different locations in the parameter space. The mixture of gaussians is used to learn spatial distributions of objects from the spatial distribution of frequencies in an image.

Figure 13 presents results for selection of the attentional focus for objects from the low-level cues given by the distribution of frequencies computed by wavelet decomposition.

Some furniture objects were not moved (such as the sofas), while others were moved in different degrees: the chair appeared in several positions during the experiment, while the table and door suffered mild displacements. Still, errors on the head gazing control added considerable location variability whenever a non-movable object was segmented and annotated. It demonstrates that, given an holistic characterization of a scene (by PCA on the image wavelet decomposition coefficients), one can estimate the appropriate places whether objects often appear, such as a chair in front of a table, even if no chair is visible at the time – which also informs that regions in front of tables are good candidates to place a chair. Object occlusions by people are not relevant, since local features are neglected, favoring contextual ones.

3.7 Back-propagation Neural Networks

Activity Identification

A feature vector for activity recognition was proposed by (Polana & Nelson, 1994) which accounts for 3-dimensional information: 2-dimensional spatial information plus temporal information. The feature vector is thus a temporal collection of 2D images. Each of these images is the sum of the normal flow magnitude (computed using a differential method) – discarding information concerning flow direction – over local patches, so that the final resolution is of 4×4. The normal flow accounts only for periodically moving pixels. Classification is then performed by a nearest centroid algorithm.

Our strategy reduces the dimensionality of the feature vector to 2-dimensional. This is done by constructing a 2D image which contains a description of an activity. Normalized length trajectories over one period of the motion are mapped to an image, in which the horizontal axis is given by the temporal scale, and the vertical axis by 6 elements describing position and 6 elements for velocities. The idea is to map trajectories into images. This is fundamentally different to the trajectory primal-sketch approach suggested in (Gould & Shah, 1989), which argues for compact representations involving motion discontinuities. We opt instead for using redundant information.

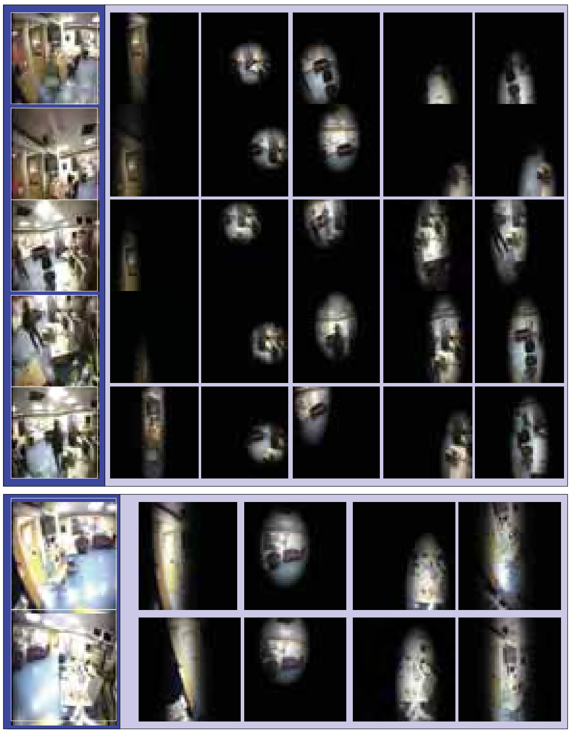


Fig. 13. Localizing and recognizing objects from contextual cues. (top) Samples of scene images are shown on the first column. The next five columns show probable locations based on context for finding a door, the smaller sofa, the bigger sofa, the table and the chair, respectively. Even if the object is not visible or present, the system estimates the places at which there is a high probability of finding such object. Two such examples are shown for the chair. Occlusion by humans do not change significantly the context. (bottom) Results in another day, with different lightning conditions.

Activities, identified as categories which include objects capable of similar motions, and the object's function in one activity, can then be learned by classifying 12 × 12 image patterns. One possibility would be the use of eigenobjects for classification (as described in this chapter for face and sound recognition). Eigenactivities would then be the corresponding eigenvectors. We opted instead for neural networks as the learning mechanism to recognize activities.

Target desired values, which are provided by the multiple object tracking algorithm, are used for the annotation of the training samples - all the training data is automatically generated and annotated, instead of the standard manual, offline annotation. An input feature vector is recognized into a category if the corresponding category output is higher than 0.5 (corresponding to a probability $p > 0:5$). Whenever this criterion fails for all categories, no match is assigned to the activity feature vector – since the activity is estimated as not yet in the database, it is labeled as a new activity.

We will consider the role of several objects in experiments taken for six different activities. Five of these activities involve periodic motion: cleaning the ground with a swiping brush; hammering a nail-like object with a hammer; sawing a piece of metal; moving a van toy; and playing with a swinging fish. Since more information is generated from periodic activities, they are used to generate both training and testing data. The remaining activity, poking a lego, is detected from the lego's discontinuous motion after poking. Figure 14 shows trajectories extracted for the positions of four objects from their sequences of images.

A three layer neural network is first randomly initialized. The input layer has 144 perceptron units (one for each input), the hidden layer has six units and the output layer has one perception unit per category to be trained (hence, five output units). Experiments are run with a set of (15, 12, 15, 6, 3, 1) feature vectors (the elements of the normalized activity images) for the swiping brush, hammer, saw, van toy, swinging fish and lego, respectively. A first group of experiments consists of selecting randomly 30% of these vectors as validation data, and the remaining as training data. The procedure is repeated six times, so that different sets of validation data are considered. The other two groups of experiments repeat this process for the random selection of 20% and 5% of feature vectors as validation data. The correspondent quantitative results are presented in figure 15.

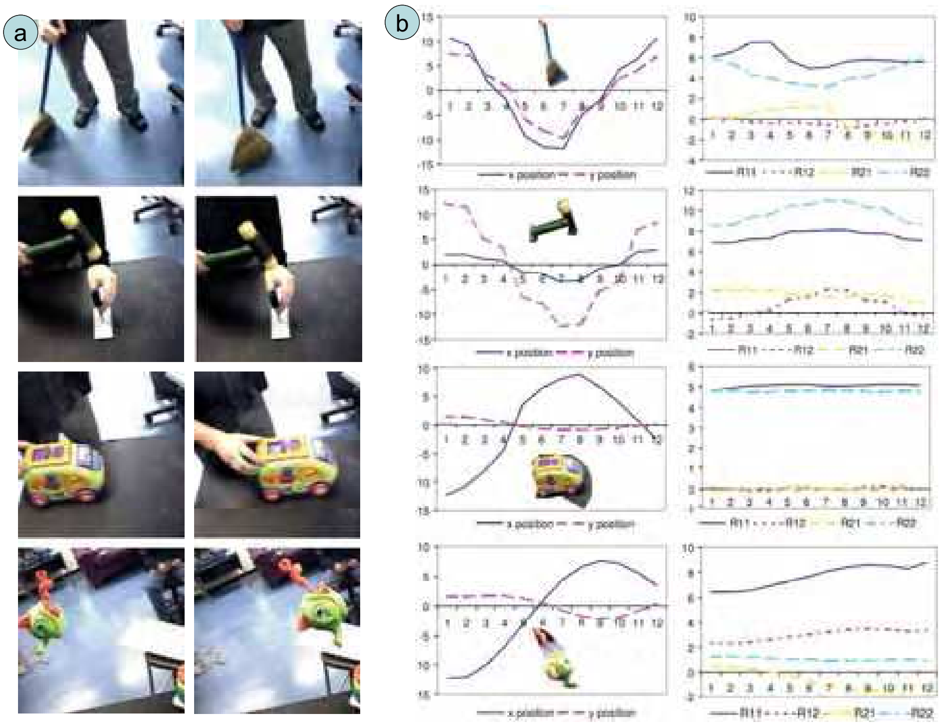
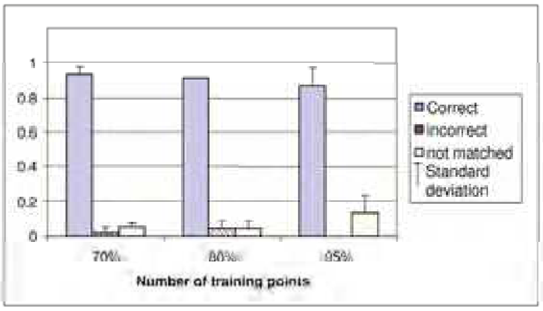
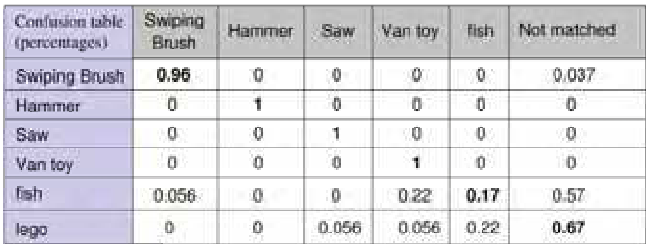


Fig. 14. a) Signals corresponding to one period segments of the object's trajectories normalized to temporal lengths of 12 points. From top to bottom: image sequence for a swiping brush, a hammer, a van toy and a swinging fish. b) Normalized centroid positions are shown in the left column, while the right column shows the (normalized and scaled) elements of the affine matrix R^i (where the indexes represents the position of the element on this matrix).

The lego activity, not represented in the training set, was correctly assigned as a new activity for 67% of the cases. The swinging fish was correctly recognized for just 17% of the cases, being the percentage of no matches equal to 57%. We believe that this poor result was due to the lack of a representative training set – this assumption is corroborated by the large number of times that the activity of swinging a fish was recognized as a new activity. The swiping brush was wrongly recognized for 3,7% of the total number of trials. The false recognitions occurred for experiments corresponding to 30% of the validation data. No recognition error was reported for smaller validation sets. All the other activities were correctly recognized for all the trials.





Confusion table (percentages)	Swiping Brush	Hammer	Saw	Van toy	fish	Not matched
Swiping Brush	**0.96**	0	0	0	0	0.037
Hammer	0	**1**	0	0	0	0
Saw	0	0	**1**	0	0	0
Van toy	0	0	0	**1**	0	0
fish	0.056	0	0	0.22	**0.17**	0.57
lego	0	0	0.056	0.056	0.22	**0.67**

Fig. 15. Experimental results for activity recognition (and the associated recognition of object function). Each experiment was ran six times for random initial conditions. Top graph) from left to right columns: 30%, 20% and 5% of the total set of 516 feature vectors are used as validation data. The total number of training and validation points, for each of the six trials (and for each of the 3 groups of experiments), is (15, 12, 15, 6, 3, 1) for the swiping brush, hammer, saw, van toy, swinging fish and lego, respectively. The three groups of columns show recognition, error and missed-match rates (as ratios over the total number of validation features). The bar on top of each column shows the standard deviation. Bottom table: Recognition results (as ratios over the total number of validation features). Row i and column j in the table show the rate at which object i was matched to object j (or to known, if j is the last column). Bold numbers indicate rates of correct recognitions.

Sound Recognition

An artificial neural network is applied off-line to the same data collected as before for sound recognition. The 32 × 32 sound images correspond to input vectors of dimension 1024. Hence, the neural network input layer contains 1024 perceptron units. The number of units in the hidden layer was set to six, while the output layer has four units corresponding to the four categories to be classified. The system is evaluated quantitatively by randomly selecting 40%, 30% and 5% of the segmented data for validation, and the remaining data for training. This process was randomly repeated six times. This approach achieves higher recognition rates when compared to eigensounds. The overall recognition rate is 96,5%, corresponding to a significant improvement in performance.

3.8 Other Learning Techniques

Other learning techniques exploited by Cog's cognitive system includes nearest-neighbor, locally linear receptive-field networks, and Markov models.

Locally Linear Receptive-field Networks

Controlling a robotic manipulator on the cartesian 3D space (eg. to reach out for objects) requires learning its kinematics – the mapping from joint space to cartesian space – as well as the inverse kinematics mapping. This is done through locally weighted regression and Receptive-field weighted regression, as proposed by (Schaal et al., 2000). This implementation on the humanoid robot Cog is described in detail by (Arsenio 2004c;d).

Markov Chains

Task descriptions can be modeled through a finite Markov Decision Process (MDP), defined by five sets <*S; A; P;R;O* >. Actions correspond to discrete, stochastic state-transitions a∈A={Periodicity, Contact, Release, Assembling, Invariant Set, Stationarity} from an environment's state s_i∈S to the next state s_i+1, with probability $P^a_{s_i s_{i+1}} \in P$, where P is a set of transition probabilities $P^a_{ss'} = P_r\{s_{i+1} = s' \mid s, a\}$.

Task learning consists therefore on determining the states that characterize a task and mapping such states with probabilities of taking each possible action (Arsenio, 2003; Arsenio, 2004d).

4. Cognitive development of a Humanoid Robot

The work here described is part of a complex cognitive architecture developed for the humanoid robot Cog (Arsenio, 2004d), as shown in Figure 16. This chapter focused on a very important piece of this larger framework implemented on the robot. The overall framework places a special emphasis on incremental learning. A human tutor performs actions over objects while the robot learns from demonstration the underlying object structure as well as the actions' goals. This leads us to the object/scene recognition problem. Knowledge concerning an object is organized according to multiple sensorial percepts. After object shapes are learned, such knowledge enables learning of hand gestures. Objects are also categorized according to their functional role (if any) and their situatedness in the world. Learning *per si* is of diminished value without mechanisms to apply the learned knowledge. Hence, robot tasking deals with mapping learned knowledge to perceived information, for the robot to act on objects, using control frameworks such as neural oscillators and sliding-motion control (Arsenio, 2004).

Teaching a humanoid robot information concerning its surrounding world is a difficult task, which takes several years for a child, equipped with evolutionary mechanisms stored in its genes, to accomplish.

Learning aids such as books or educational, playful activities that stimulate a child's brain are important tools that caregivers extensively apply to communicate with children and to boost their cognitive development. And they also are important for human-robot interactions.

If in the future humanoid robots are to behave like humans, a promising venue to achieve this goal is by treating then as such, and initially as children – towards the goal of creating a 2-year-old-infant-like artificial creature.

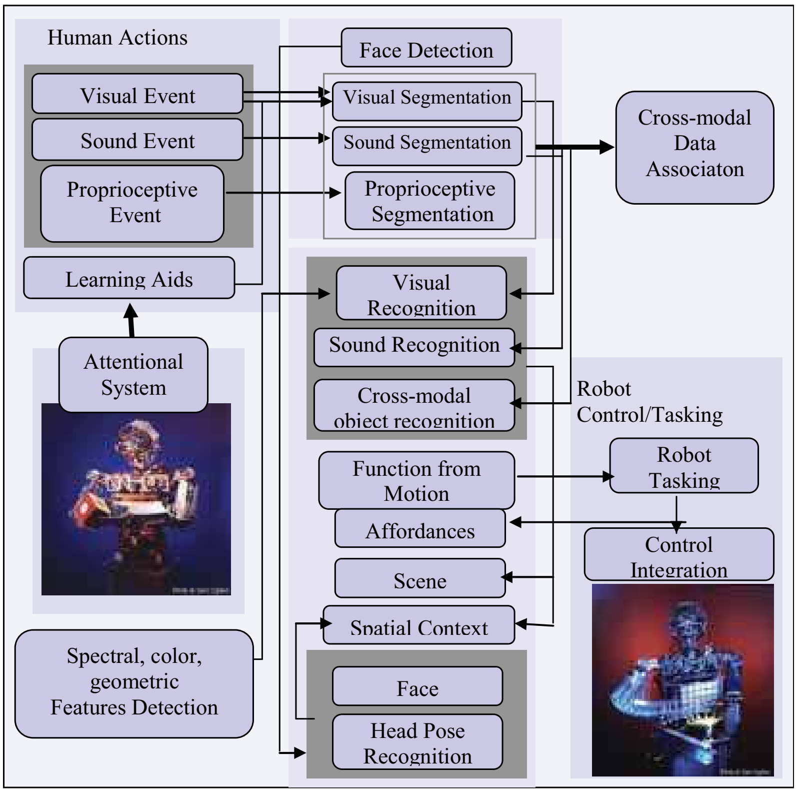


Fig. 16. Overview of the cognitive architecture developed for the humanoid robot Cog.

5. Conclusions

We proposed in this chapter the application of a collection of learning algorithms to solve a broad scope of problems. Several learning tools, such as Weighted-cluster modeling, Artificial Neural Networks, Nearest Neighbor, Hybrid Markov Chains, Geometric Hashing, Receptive Field Linear Networks and Principal Component Analysis, were extensively applied to acquire categorical information about actions, scenes, objects and people.

This is a new complex approach to object recognition. Objects might have various meanings in different contexts – a rod is labeled as a pendulum if oscillating with a fixed endpoint. From a visual image, a large piece of fabric on the floor is most often labeled as a tapestry, while it is most likely a bed sheet if it is found on a bed. But if a person is able to feel the fabric's material or texture, or the sound that it makes (or not) when grasped with other materials, then (s)he might determine easily the fabric's true function. Object recognition draws on many sensory modalities and the object's behavior, which inspired our approach.

6. References

Arsenio, A. (2003). Embodied vision - perceiving objects from actions. *Proceedings of IEEE International Workshop on Human-Robot Interactive Communication*, San-Francisco, 2003.

Arsenio, A. (2004a). Teaching a humanoid robot from books. *Proceedings of International Symposium on Robotics*, March 2004.

Arsenio, A. (2004b). Map building from human-computer interactions. *Proceedings of IEEE CVPR International Conference - Workshop on Real-time Vision for Human Computer Interaction*, 2004.

Arsenio, A. (2004c). Developmental Learning on a Humanoid Robot. *Proceedings of IEEE International Joint Conference on Neural Networks,* Budapest, 2004.

Arsenio, A. (2004d). Cognitive-developmental learning for a humanoid robot: A caregiver's gift, MIT PhD thesis, September 2004.

Arsenio, A. (2004e). Figure/ground segregation from human cues. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-04),* 2004.

Arsenio, A. (2004f). Object recognition from multiple percepts. *Proceedings of IEEE/RAS International Conference on Humanoid Robots*, 2004.

Cutler, R. & Turk, M. (1998). View-based interpretation of real-time optical flow for gesture recognition, *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 1998.

Darrel, T. & Pentland, A. (1993). Space-time gestures, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 335-340, New York, NY, 1993.

Fitzpatrick, P. & Arsenio, A. (2004). Feel the beat: using cross-modal rhythm to integrate robot perception. *Proceedings of Fourth International Workshop on Epigenetic Robotics*, Genova, 2004.

Gershenfeld, N. (1999). The nature of mathematical modeling. *Cambridge university press*, 1999.

Gould, K. & Shah, M. (1989). The trajectory primal sketch: A multi-scale scheme for representing motion characteristics. *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pages 79–85, 1989.

Metta, G. & Fitzpatrick, P. (2003). Early integration of vision and manipulation. *Adaptive Behavior*, 11:2, pp. 109-128, June 2003.

Oliva, A. & Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, pages 145–175, 2001.

Perrett, D.; Mistlin, A.; Harries, M. & Chitty, A. (1990). Understanding the visual appearance and consequence of hand action, *Vision and action: the control of grasping,* 163-180, Ablex, Norwood, NJ, 1990.

Polana, R. & Nelson, R.. (1994). Recognizing activities. *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, October 1994.

Rao, K.; Medioni, G.& Liu, H. (1989). Shape description and grasping for robot hand-eye coordination, *IEEE Control Systems Magazine*, 9 (2) 22{29, 1989.

Rissanen, J. (1983). A universal prior for integers and estimation by minimum description length. *Annals of Statistics*, 11:417–431, 1983.

Schaal, S.; Atkeson, C. & Vijayakumar, S. (2000). Real-time robot learning with locally weighted statistical learning. *Proceedings of the International Conference on Robotics and Automation*, San Francisco, 2000.

Strang, G. & Nguyen, T. (1996). Wavelets and Filter Banks. *Wellesley-Cambridge Press*, 1996.

Torralba, A. (2003). Contextual priming for object detection. *International Journal of Computer Vision*, pages 153–167, 2003.

Turk, M. & Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1), 1991.

Vigotsky, L. (1962). Thought and language. *MIT Press*, Cambridge, MA, 1962.

Viola, P. & Jones, M. (2001). Robust real-time object detection. *Technical report, COMPAQ Cambridge Research Laboratory*, Cambridge, MA, 2001.

Wolfson, H. & Rigoutsos, I. (1997). Geometric hashing: an overview. *IEEE Computational Science and Engineering*, 4:10–21, 1997.

Humanoid Robots: New Developments
Edited by Armando Carlos de Pina Filho

ISBN 978-3-902613-00-4
Hard cover, 582 pages
Publisher I-Tech Education and Publishing
Published online 01, June, 2007
Published in print edition June, 2007

For many years, the human being has been trying, in all ways, to recreate the complex mechanisms that form the human body. Such task is extremely complicated and the results are not totally satisfactory. However, with increasing technological advances based on theoretical and experimental researches, man gets, in a way, to copy or to imitate some systems of the human body. These researches not only intended to create humanoid robots, great part of them constituting autonomous systems, but also, in some way, to offer a higher knowledge of the systems that form the human body, objectifying possible applications in the technology of rehabilitation of human beings, gathering in a whole studies related not only to Robotics, but also to Biomechanics, Biomimmetics, Cybernetics, among other areas. This book presents a series of researches inspired by this ideal, carried through by various researchers worldwide, looking for to analyze and to discuss diverse subjects related to humanoid robots. The presented contributions explore aspects about robotic hands, learning, language, vision and locomotion.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821