Learning and Nonlinear Models - Revista da Sociedade Brasileira de Redes Neurais, Vol. 6, No. 1, pp. 44-67, 2008.

Concept Learning by Human Tutelage for Social Robots

Renato Ramos da Silva [a] Claudio Adriano Policastro [a]
Giovana Zuliani [b] Ednaldo Pizzolato [b]
Roseli Aparecida Francelin Romero [a]

[a] *University of Sao Paulo*
Av. Trabalhador Sao-carlense, 400 - Centro
Caixa Postal: 668 - CEP: 13560-970 - Sao Carlos - Sao Paulo

[b] *Federal University of Sao Carlos*
Rod. Washington Luiz, km 235
Caixa Postal: 676 - CEP: 13565-905 - Sao Carlos - Sao Paulo

Abstract

Learning by human tutelage means that a human being guides the attention of a robot or agent in order to teach it a given concept. This kind of learning is very important to developing a robotic architecture for social robots. Social robots are embodied agents that are part of a heterogeneous society of robots and humans. The use of a robotic architecture may strongly reduce the time and effort required to construct a social robot. Such architecture must have structures and mechanisms to allow perception and attention, to enable a social robot to realize the richness of the human behavior and of the environment, and to learn from social interactions. In this paper, a robotic architecture inspired on Behavior Analysis is presented with two different learning algorithms, learning by Contingency and by Economic TG, for controlling a robotic head. For interaction between human and head robotic, it has also been developed a multimodal interface. The multimodal interface employs mechanisms of face detection, pose estimation, object saliency and speech recognition. The experimental results show that the robotic head, with the proposed tutelage mechanism, is able to learn concepts about objects of the environment successfully.

Key words: Learning, Multimodal Interface, Social Robots, Machine Learning

1 Introduction

Social robots should be able to interact, to communicate, to understand and to relate with human beings in a natural way. Additionally, these robots should be able to learn from the interactions with the human beings, acquiring new knowledge and adapting their behaviors in response to stimulus and the context of the environment (15; 4; 40).

In (15), Dautenhahn and Billard proposed the following definition for social robots:

> *Social robots are embodied agents that are part of a heterogeneous group: a society of robots or humans. They are able to recognize each other and engage in social interactions, they possess histories (perceive and interpret the world in terms of their own experience), and they explicitly communicate with and learn from each other.*

Many of the earliest motivations for developing humanoids centered on creating robots that can play a role in the daily lives of people. Today, humanoid robots are being developed to provide the elderly with assistance in their homes and to support medical care in hospitals. In other applications, humanoids are being developed to serve as members of human-robot teams. One such example is NASA JSCs Robonaut, a robot envisioned to serve as an astronauts assistant in space station maintenance operations. In the future, we expect to see more applications for robots that share our environment and tools and participate in joint activities with untrained humans. Robots with a human-like morphology would not require us to re-engineer our environment and tools to accommodate them. Beyond form factor, however, there are critical social issues that concern how robots should interact with us (8).

In (6), Breazeal defines four classes of social robots in terms of: how well the robot can support the social model that is ascribed to it and the complexity of the interaction scenario that can be supported as follows.

Socially evocative Robots that rely on the human tendency to anthropomorphize and capitalize on feelings evoked when humans nurture, care, or involved with their creation.

Social interface Robots that provide a natural interface by employing

Email addresses: ramos@icmc.usp.br (Renato Ramos da Silva), capoli@icmc.usp.br (Claudio Adriano Policastro), giovana@ufscar.br (Giovana Zuliani), ednaldo@dc.ufscar.br (Ednaldo Pizzolato), rafrance@icmc.usp.br (Roseli Aparecida Francelin Romero).

URLs: http://www.icmc.usp.br/ rafrance/, http://www2.dc.ufscar.br/ ednaldo/ (Roseli Aparecida Francelin Romero).

human-like social cues and communication modalities. Social behavior is only modeled at the interface, which usually results in shallow models of social cognition.

Socially receptive Robots that are socially passive but that can benefit from interaction (e.g. learning skills by imitation). Deeper models of human social competencies are required than with social interface robots.

Sociable Robots that pro-actively engage with humans in order to satisfy internal social aims (drives, emotions, etc.). These robots require deep models of social cognition.

Complementary to this list, Fong (21) added the following three classes:

Socially situated Robots that are surrounded by a social environment that they perceive and react to (16). Socially situated robots must be able to distinguish between other social agents and various objects in the environment.

Socially embedded Robots that are: situated in a social environment and interact with other agents and humans; structurally coupled with their social environment; and at least partially aware of human interactional structures (e.g., turn-taking) (16).

Socially intelligent Robots that show aspects of human style social intelligence, based on deep models of human cognition and social competence (13; 14)

Although social robots have already been used with success, much work remains to be done in order to increase their effectiveness. The use of a robotic architecture may strongly reduce the time and effort required to construct a social robot. Several robotic architectures were proposed in the literature (20) (5), (9). Indeed, a robotic architecture for social robots must have structures and mechanisms to allow appropriate interaction, to learn from the environment, and to *perceive* and to *understand* the richness of the human behavior and of the environment. Also, it should incorporate mechanisms for face recognition, object recognition, gesture recognition, saliency detection, among others (3) (1).

In a tentative to build a robotic architecture for social robots, we decide to focus on shared attention. In our previous works (38; 37), we proposed a robotic social architecture inspired on Behavior Analysis, which employs a relational representation to represent the domain and the learned knowledge. We have proposed some learning algorithms that were tested only in the level of simulation (37; 12). Our architecture incorporates mechanism for face detection, object recognition, salience detection and provides the ability of share attention to a robotic head.

Shared attention has been defined in the literature as the “capacity to use gestures and eye contact to coordinate attention with another agent to share experiences of interesting objects or events" (19; 26). It is associated to the situation in which two agents are looking each other, one agent turns his gaze to one object present in the environment, and the second one follow the gaze to the correct object. The ability of shared attention makes possible the learning of what is important in the environment (17).

In this paper, our robotic architecture, employing the knowledge engaged on sessions of shared attention, has been extended. Firstly, a multimodal interface oriented to human behavior has been proposed. The multimodal interface is composed by a vision system and a voice system that allows perception and attention on social interactions. Second, a learning mechanism by tutelage has been proposed that allows the robot to learn concepts about objects of real world.

The tutelage mechanism employs an ART2 neural network (11) and associates visual and auditory stimulus in order to simulate the learning of concepts about objects. This mechanism enables the robot to learn by tutelage, that is, a human being guides the attention of the robot in order to teach it a given concept. The tutelage mechanism has been integrated to robotic architecture and may be activated by it, during a social interaction.

This article is organized as it follows. In section 2, we briefly discuss some concepts and mechanisms of perception and attention on social robot. In section 3, we present the proposed architecture, a multimodal interface for interaction between the user and the robotic head, the tutelage mechanism and the two learning algorithms used for the learning processing. In section 4, results obtained from several experiments carried out to evaluate the architecture in the concept learning are presented. Finally, in section 5, the conclusions and future works are presented.

2 Perception, Attention and Learning

Social robots need to possess a human oriented perception. They need to interpret social signs like gaze, facial expressions, body movements and speech (4). They should be able to track human characteristics (faces, hands, body) and interpret speech and natural language (23) (1).

An important perception mechanism that should be considered during a social robot's project is the face detection. Recently, several approaches of real time face detection systems were proposed in the literature.

In (45) was proposed an interface for human robot interaction through gesture recognition.

In (4), for example, a system of attention that integrates visual perceptions (movement detection, colors and human faces) with a habituation mechanism was presented. Each visual perception generates a map of characteristics that are combined through a weighted sum. The system influences and is influenced by the behavioral system and by the motivational system of the robot, providing an attention system dependent of the environment context and the robot's situation and motivation.

Additionally to the communication and interaction capabilities, a social robot should be able to learn and to adapt to new experiences. The ideal, people could teach robots on how to execute new tasks or new concepts about the real world. Consequently, a fundamental challenge is to project robots that can be taught like human beings are (or as similar as possible to them) (4) (7). Several approaches of learning mechanisms for social robots have been proposed in the literature (4) (30) (35).

In (30), for example, a biologically inspired imitation mechanism (based on a structure called mirror neurons) was proposed. The mechanism is composed by a perceptual system and by a scheme network that simulates the functions of the mirror neurons. The perceptual system is formed by a self-organizing neural network that is able to recognize and classify stimulus of the environment. The motor scheme contains sequences of motor scripts that perform a piece of a desired behavior. When a perceptual scheme receives a stimulus of the environment, it compares this stimulus with the structure stored to produce a confidence measure. If this measured is high enough(above a threshold), the correspondent motor scheme is activated.

In (35), Nagai proposed a constructive model that enables a social robot to acquire the ability of shared attention based on a mechanism of the visual attention and learning with solemnity-evaluation. The employed method acquires sense-motive information when the visual system successfully finds a salient object (which is the focus of the attention of a human being) in the environment. In this way, the robot acquires the ability of the shared attention finding the correlation between the sense-motive information and the responses of the visual system.

In this work, we are concerned with learning by tutelage. Learning by human tutelage is a collaboration between the teacher and the learner. In this process, teachers direct a learners attention and learners contribute by revealing their internal state to guide the teaching process (4) (28).

Lockerd and Breazeal (28), for example, presented a learning mechanism, implemented on a humanoid robot, to demonstrate that a collaborative dialog framework allows a robot to learn a task from a human tutelage. The robot has both speech and visual inputs. The cognitive system receives data continuously from the vision and the speech understanding systems and integrates them into several beliefs about objects in the world, gestures and speech.

In next section, the interface and all modules of the proposed robotic architecture will be described.

3 Proposed robotic architecture

In this section, we briefly present the proposed robotic architecture, composed by mechanisms and structures evidenced from Behavior Analysis, and then we detail the multimodal interface and the learning mechanism proposed in this work. All these mechanisms are embedded in the robotic head showed on Figure 1.

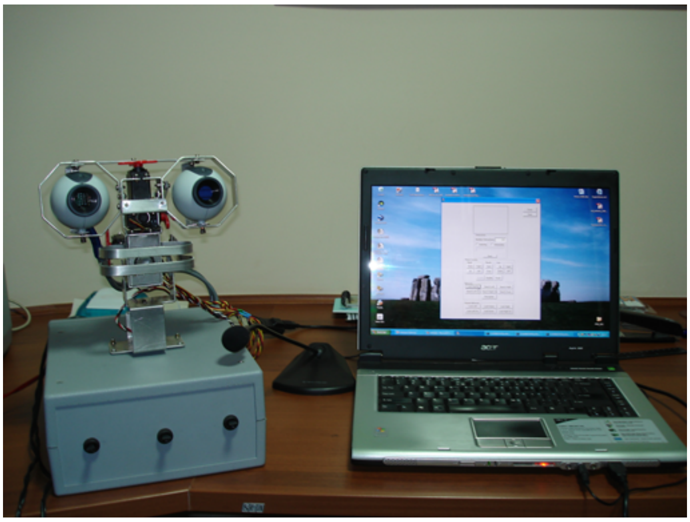

Fig. 1. Robotic Head - WHA8030 of Dr. Robot (41).

This interactive robotic head is composed of 5 servo-motors and two color web cameras (one camera used for face tracking and one camera used for object detection). This mechanism enables the robot head to move in 6 different directions (left, left down, down, right down, right and center). At moment when the architecture has been executed into computer, the communication with robotic head is done by a serial cable.

The robotic architecture simulates an individual's operant conditioning through histories of reinforcement. It is composed by three main modules: Stimulus Perception Module, Response Emission Module and Consequence Control Module. In Figure 2 illustrated the general organization of the proposed architecture and the interaction among the three main modules. In Figure 2, arrows indicate the flow of information in the three modules of the architecture, whereas the circles indicate the methods and component structures of the modules.

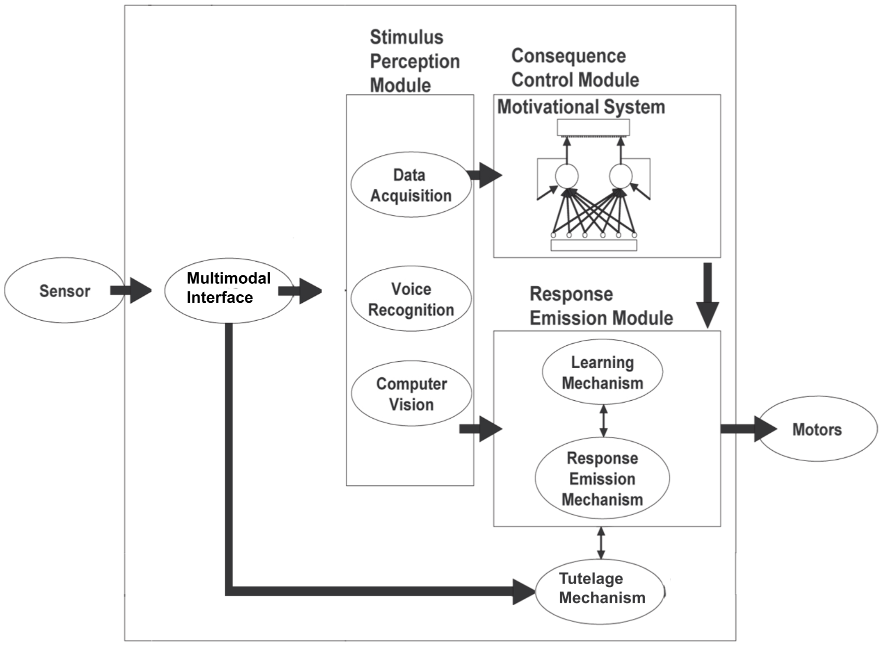

Fig. 2. General organization of the architecture.

The multimodal interface is composed by a vision system and a voice system that allows perception and attention on social interactions.

The vision system is composed by 2 mechanisms: the face recognition and head pose estimation (based on Adaptive Appearance Model (34)) and the visual attention mechanism (based on saliency (25)).

The voice system is able to recognize naturally spoken *Portuguese* utterances and is based on the Nuance System (36).

The Stimulus Perception Module may employ, depending on the application domain, algorithms of data acquisition, a vision system and a voice recognition system. This module detects the state from the environment and encodes this state using an appropriate representation.

The Consequence Control Module is composed by a motivational system that simulates internal necessities of the robot and detects reinforcements received from the environment. The motivational system is formed by necessity units that are implemented by a perceptron (24) with recurrent connections. These necessity units simulate the homeostases of an alive organism. A positive value of a necessity unit, greater than a predefined threshold, indicates the privation of the robot to certain reinforcement stimulus. In this way, the architecture supplies mechanisms to simulate privation states and satisfaction of necessities, and to determine reinforcements as consequences of an emitted response.

The tutelage mechanism employs an ART2 neural network (11). It is able to associate visual and auditory stimulus in order to simulate the learning of concepts about objects of the real world. This mechanism enables the robot to learn by tutelage, that is, a human being guides the attention of the robot in order to teach it a given concept. The tutelage mechanism is integrated with the robotic architecture and may be activated by the Response Emission Module, during social interactions, on appropriate contexts.

The architecture employs a working memory to exchange information among the three main modules. This memory is used to keep information about stimulus (antecedents and consequents), last emitted response and internal necessities. Each element inserted in the working memory has a counter that keeps the notion of time. When a new element is inserted in the working memory, its age counter is set to zero, and it is incremented by 1 whenever new subsequent predicates are inserted. So, elements persist for a number of time steps in the memory. This mechanism is employed to control the chronology of facts and events, and to determine the three terms of a contingency.

A contingency is constituted by three terms: antecedent stimulus, last emitted response and consequent stimulus, represented by a rule in our architecture.

Besides all that was said about the architecture, it still uses a prior knowledge. This prior knowledge contain the possible facts that can be happened. For example, the *looking_right(face)* is a fact defining one quality of the environment and some stimulus (such as, *face*). This is used in the architecture to construct a knowledge base.

3.1 Multimodal Interface

In this section, we detail the multimodal interface developed for interaction between the user and the robotic head.

This interface is composed by a vision system and by a voice recognition system (based on Nuance solution). The vision system is based on the work presented by Breazeal and Scassellati (10).

The implementation of the vision system is based on maps of characteristics processed for each perception (colors and faces). The vision system creates an activation map that can be used by the other modules of our architecture in order to control the robot's behavior.

The color map is based on the work of search and visual attention, presented by Itti and their colleagues in (25). This process employs a biologically inspired visual attention mechanism to create a map of characteristics that represents the *visual saliency* of the scene. This visual saliency is formed by the composition of several maps of characteristics extracted from the image. Each map of characteristics presents an elementary property of the image as color, intensity and orientation. These characteristics are known as primitive visual characteristics. The method for the construction of this saliency map can be divided into the 5 stages: extraction of characteristics, linear filtering, center-surround difference calculations, sum of the maps of characteristics (liner combination) and selection of salient areas. The selection of the most salient area is carried out using a saliency threshold and a minimum length of area threshold. Afterwards, a process based on color histograms is carried out to obtain the more frequent values of the channels r, g, b (of the RGB color space) and h (of the HSI color space) in the area of interest. This color map was developed using the functions of saliency of the Lti-Lib library (29).

The face map is based on works presented by Morency (34) (33), about face and pose detection. The face detection is carried out employing an approach based on active appearance model (34) (33). In this approach, the principal component analysis (PCA) is used for finding the vectors that best describe the distribution of images inside the whole space of training images. Once a face is detected, the vision system proceeds the detection of its pose (*pan* and *tilt* angle). This algorithm creates a reference model using an initial frame and calculates the changes of the pose employing the created model. This face map was developed using the functions of face detection of the Watson library (33).

The voice system, based on the Nuance solution (36), contains a speech recognizer and grammatical knowledge base. The speech synthesis is performed by joining pre-recorded prompts in order to build complete phrases. This strategy enables short conversations with the robot.

3.2 Tutelage Mechanism

In this work, we are proposing a tutelage mechanism that is able to associate visual and auditory stimulus in order to simulate the learning of concepts about real world objects. This preliminary version of the learning mechanism supports only colors. Future work includes the extension of this mechanism by adding new techniques of computer vision for the processing of shapes and textures.

The tutelage mechanism uses the vision system and the voice system to integrate visual and auditory features about a given object. These features are learned and organized employing an ART2 neural network (11) and a flat memory that stores the characteristics of the learned objects to form a new concept. The ART2 neural network was chosen as learning mechanism because it has been used successfully in several works (37; 42). The learning mechanism contains three levels of memory organization (see Figure 3):

- The first level (LEVEL 1 in Figure 3) is composed by the ART2 neural network input layer. This input layer contains four input nodes, one for each color channel (r, g, b, and h).
- The second level (LEVEL 2 in Figure 3) is composed by the output layer of the ART2 neural network, which creates and also indicates clusters of objects with similar characteristics, enabling recognition and concept learning.
- The third level (LEVEL 3 in Figure 3) consists of a simple flat memory that stores the visual and auditory characteristics of the objects.

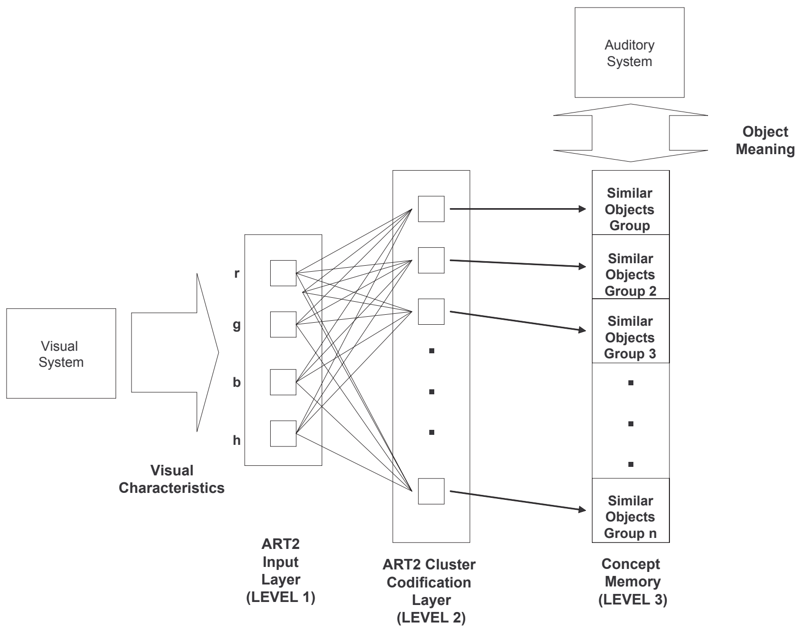

Fig. 3. General architecture of the learning mechanism. The ART2 *input layer* receives the color characteristics of an object (r, g, b, and h) from the vision system and indicates the *cluster codification* of the object. Then, for unknown objects, the learning mechanism obtains the *object meaning*, from the voice system, and join together the visual and auditory information in order to form a new concept in the *concept memory*.

The learning mechanism works as it follows. Initially, the *concept memory* is empty. When an object is presented to the robot, the vision system encodes this object by its more frequent value of the r, g, b, and h channels. The ART2 *input layer* receives the color characteristics of an object (r, g, b, and h) from the vision system and indicates that there is no active clusters in its output layer. Then, the learning mechanism enters into a state of *unknown mode* and activates the voice system to inform the user (through vocalization) that the object is unknown. Afterwards, the system waits for the correct object meaning, from the auditory system, and stores the new learned object in the *concept memory*.

When new objects are presented to the robot, the ART2 *input layer* receives the color characteristics of an object from the vision system and indicates the cluster of codification, if there is one, of the new object. Then, the search algorithm searches for objects in the indicated cluster, employing a metric given by: $m = \|h_n - h_r\|$, where h_n is the h value of the new object, and h_r is the h value of the object stored in the concept memory.

If the search algorithm finds some object bellow a *confidence threshold* (ϕ_c), the learning mechanism enters into a state of *known mode* and activates the voice system to vocalize the name of the object.

If the search algorithm finds only objects above a *knowledge threshold* (ϕ_k), the learning mechanism enters into a state of *unknown mode* and activates the voice system to vocalize that the object is unknown. Afterwards, the system waits for the correct object meaning, from the auditory system, and stores the new learned object in the *concept memory*.

If the search algorithm finds only objects between the *confidence threshold* and the *knowledge threshold*, the learning mechanism enters into a state of *uncertainty mode* about the object and activates the voice system to vocalize that the object is suppose to be the nearest object found in the concept memory. Afterwards, the system waits for the correct object meaning or for the confirmation about its guess, from the auditory system, and stores the new learned object in the *concept memory*.

3.3 Response Emission Module

The Response Emission Module is composed by a response emission mechanism and by a learning algorithm.

The response emission mechanism receives the action selected by the learning algorithm to execute motor script related.

The learning mechanism is concerned to construct a nondeterministic policy for response emission and this mechanism is answerable for connection of response emission module with others. This mechanism is made to provide the ability of share attention, one important ability to interaction of human and robot (12). We show two different learning algorithms which will be appraised on robotic architecture, but first, we will show the similarity between them.

During an interaction, the stimulus perception module acquires and codifies the environment state and deploys this coded state for the response emission and the consequence control modules. After, the consequence control module checks the internal state of the robot and sets the active necessities, if there is one. Then, the architecture control enters a loop that may be finished either at the end of an interaction or when the robot reaches its goal.

Afterwards, the selected response is emitted by executing a motor script. Then, the stimulus perception module acquires and encodes the new current environment state and sends it to the response emission. The consequence control module propagates the encoded new state through the motivational system and checks the internal state of the robot and any reinforcement got as consequence of the last emitted response.

In Figure 4 is illustrated the process of learning. Initially, the robot is looking at some place in the environment. Then the robot searches for a human and find one. Then, the human gives attention to the robot (a reinforcer stimuli defined by the knowledge). Afterwards, the architecture detects the reinforcement received as consequence of the response emission (stated as "0 $get(attention)$"). Then, the architecture retrieves the last emitted response and all antecedent stimuli, and creates a new behavior rule from these terms. Then, the architecture sets the execution and reinforcement counter, C_n and C_r, to 1 and calculates the fitness value of the created rule, storing the necessity satisfied by the rule execution.

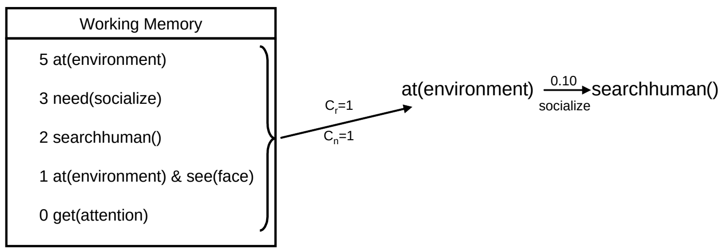

Fig. 4. Example of learning process. The bracket indicates all predicates used to create the new behavior rule. C_r and C_n, are the counter associated to the new rule, initially set to 1. The arrow represents the new rule creation process.

Before presenting the learning algorithm is important address its interaction with consequence module.

In fact, the necessity associates the state with actions look upon shared attention problem. We have used three necessities: none, attention and play. Each one is associated with a pair (action,state) in training phase of the learning algorithm. When the architecture searches for an action, it verifies if the necessity value is equal to that one produced by the consequence control module.

Only actions with the same necessity value are candidate to be chosen. Then, the reinforcement value is used to choose the best action. This mechanism is important to reinforcement learning operation because the problem of share attention can be addressed in a fully observable state.

We have implemented two learning algorithms, both described to follow. As it will be showed, each one of them can be adopted in the learning mechanism module (Figure 2). In the way how each one selects the action is the main difference between them.

3.3.1 Contingency Learning

The architecture is able to simulate learning of contingencies and stimulus discrimination from histories of reinforcement. Learning is carried out by a nondeterministic reinforcement learning algorithm (44) (32) by storing new behavior rules and updating the execution probability of existing ones. In Algorithm 1 is presented the contingency learning algorithm.

Algorithm 1 Contingency Algorithm

```
Get the environment state s_0
Check reinforcement r_0
Check active necessity
repeat
  emit a response non deterministically selected with equation (1)
  Get the environment state s_i
  Check reinforcement r_i
  Check active necessity
  if Last response is not a rule then
    Create a new rule setting a fitness value with equation (2)
  else
    Updated the fitness value of existing rule with equation (3)
  end if
until forever
```

In the loop, the response emission module uses the state and necessity information to select a response to be emitted by the robot. Response selection is done in a probabilistic way, based on Roulette Wheel selection method (22). This method is also called stochastic sampling with replacement. The roulette-wheel selection algorithm provides a zero bias and the probability to be chosen is proportional to the fitness value. The probability distribution has been defined by us:

$$P(s|a) = \frac{f_i \pm I}{\sum\limits_{j=1}^{n} f_j} \tag{1}$$

where f_i and f_j are fitness values of each response or behavior rule, n is the number of response and behavior rule and I is the influence rate. All responses in the robot's repertory keeps a default fitness value (f_d) that is predefined as a parameter in the architecture. This default fitness value, as well as fitness values from the behavior rules, are employed for building the selection roulette. While the appropriate behavior rule is selected, the response selection method can increase or decrease its fitness value by the influence rate, either if a rule satisfies an active necessity, or if a rule satisfies an inactive necessity, respectively. The influence rate (I) is given by the motivational system. It reflects the internal state of the robot and it is given by the difference between the activation value of a necessity unit and the activation threshold. Therefore, the influence is positive when the necessity unit is active and negative when the necessity unit is inactive.

If the last emitted response is not yet a rule, the learning algorithm then links the three-term contingency (antecedent stimulus, last emitted response and consequence), storing this new knowledge as a new behavior rule. The fitness of new behavior rule (f_n) take the default fitness value (f_d), it can be represented as:

$$f_n = f_d \tag{2}$$

If the behavior rule already exists, the architecture updates its fitness using the perceived consequence of its execution. Fitness update is carried out employing the learning rule given by:

$$f_t = \alpha_n \times (P \times \frac{C_r}{C_n}) + (1 - \alpha_n) \times f_t \tag{3}$$

where f_t is the new fitness value at present time, P is the power of a reinforcement stimuli, C_r and C_n are the reinforcement and execution counters that represent respectively the number of execution of the rule and the number of reinforcements got, and α_n is a decreasing learning rate given by:

$$\alpha_n = \begin{cases} \lambda & if\ C_n \leq N_{Interactions} \\ \frac{\lambda}{(C_n - N_{Interactions})} & if\ C_n > N_{Interactions} \end{cases} \tag{4}$$

where $N_{Interactions}$ denotes the minimum execution number of a behavior rule before α decreases and λ is a learning constant, both set us as parameters of the architecture. This decreasing learning rate allows the convergence of the algorithm to the optimal policy. The learning constant λ can take values $0 \leq \lambda < 1$. If $\lambda = 1$, then we obtain a deterministic learning algorithm.

This function enables to increase a fitness value when a behavior rule receives a reinforcement, and to decrease a fitness value when a behavior rule does not receive a reinforcement (punishment). Fitness value f_t may vary in a range $[-\infty, +\infty]$. This mechanism allows the system to converge to an optimum policy (39).

3.3.2 Economic TG

Another learning algorithm has been implemented by us and it can be adopted is the Economic TG(ETG). It was proposed by us in (43) to be inserted in the robotic architecture in learning mechanism module (Figure 2). It is based on the works of Driessens (18), Mccallum (31) and Kearns and Mansour (27). ETG is an enhancement of TG algorithm proposed by Driessens. TG algorithm combines a standard RL algorithm (Q-learning), relational representation, a relational regression algorithm (G algorithm), as a storage mechanism, and some properties of TILDE system (2).

We proposed the ETG algorithm in an attempt to solve shared attention problem. For this problem, the algorithm does not use any properties TILDE system, and it has some modifications on standard RL algorithm and relational regression mechanism used by TG.

The works proposed by Mccallum (31) and Kearns and Mansour (27) are related to ETG. They use RL algorithm and a tree based method to store examples. However, they do not use relational representation.

The ETG algorithm learns a control policy for an agent as it moves through the environment and receives rewards for its actions. An agent perceives a state s_t, decides to take some action a_t, makes a transition from s_t to s_{t+1} and receives the reward r_t. The task of the agent is to maximize the total reward it gets while doing actions. Agents have to learn a policy which maps states into actions.

The ETG is based in Q-learning algorithm. It uses a relational regression tree to store and to access the information (12).

In Algorithm 2 is showed the processing of ETG. The algorithm starts by initializing the Q-function and creates an empty regression tree (12).

The learning mechanism takes from the environment state, an necessity of the agent, then it chooses (using the current polity) and takes an action. This process changes the state and the agent receives its reward. The reward can be either positive (equals to 10) or negative (equals to -1). After this occurred, the *qvalue* is computed by:

$$\hat{q}_i \leftarrow Q(s_i, a_i) + \alpha[r_{i+1} + \gamma * \max(Q(s_{i+1}, a_{i+1})) - Q(s_i, a_i)] \tag{5}$$

Then, the set of (state, action, *qvalue*, necessity) is presented to relational regression engine. This process is repeated until there are not more interactions to be executed. All processing can be found in Algorithm 2.

Algorithm 2 The ETG Algorithm

initialize the *Q-function* hypothesis $\hat{Q}_0$ and create a tree with a single leaf
$i \leftarrow 0$
repeat
  take state s_i
  take necessity n_i
  choose a_i for s_i using a policy derived from the current hypothesis $\hat{Q}_i$
  take action a_i, observe r_i and s_{i+1}
  Update $\hat{Q}_i$ using the equation 5
  Update relational regression algorithm using $x = (s_i, a_i, \hat{q}_i, n_i)$ to produce $\hat{Q}_{i+1}$ {Use algorithm 3}
  $i \leftarrow i + 1$
until forever

The relational regression engine receives a set of (state, action, *qvalue*, necessity) and tests the internal nodes if the state already exists. Case this performance is false, the state is inserted in the tree and the leaf receives the action with *qvalue* and necessity, forming a new branch. Otherwise, it updates the *qvalue* for respective action in the leaf node.

In a leaf node, more than one action can be considered. For an easy access to the most adequate action, these actions can be ordered in decreasing order according to their *qvalue* always that an example is inserted or updated. Each leaf also has a necessity associated with action and it refers to a necessity of the robot to choose this action on this state. Here, we use only the attention necessity. The tree algorithm adopted as a relational regression engine is presented in Algorithm 3.

As we have implemented two learning algorithms: Contingency and ETG into the learning mechanism module, we have performed two experiments to test the proposed robotic architecture under tutelage learning. These experiments are described in next section.

Algorithm 3 ETG-regression engine

repeat
  sort the state down the tree using the tests of the internal nodes until to reach leaf node or null
  if the node is a leaf **then**
    if action exists **then**
      the *Q-value* is updated for the action in the leaf node according to the example {time indicates to update the Q value of rule}
    else
      the *Q-value* is inserted and the necessity for the action in the leaf node according to the example {time indicates the creation of a rule}
    end if
  else if the null is attained **then**
    generate a node
  end if
until the example in a branch
if necessary **then**
  order actions in decreasing order
end if

4 Main Results

In this section, the main results of the experiments carried out to evaluate the proposed multimodal interface and the tutelage mechanism are presented and discussed.

The robotic head has a software development kit (SDK). SDK is composed of an *Active X* control with a several functions of the programming interface API for robot controller, which can be added to developed programs in *Visual C++*. To develop a control system, it is necessary to add the *Active X* software on a project in the programming environment of the *Visual C++* routines and to access control provided by this component of control.

4.1 Experiment #1

In the experimental scenario, a human being directed the attention of the robot and presented different objects and its names. The objects employed in the experiments were 4 types of fruits: a *red apple*, a yellow *lemon*, an *orange*, and a red *pomegranate*. The purpose of the experiments was to evaluate the capability of the learning mechanism on exhibiting appropriate social behavior, of learning from interaction and of generalizing the learned concepts about the objects. Figure 5 shows the image processing carried out by the vision system when an apple was presented to the robot.

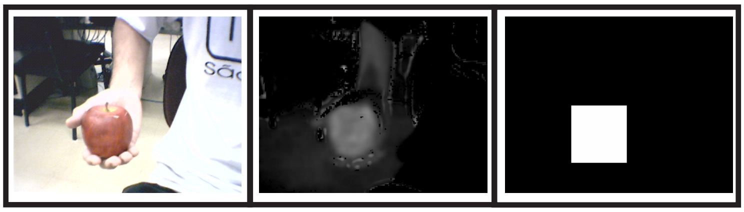

Fig. 5. Image processing carried out by the vision system when an apple was presented to the robot (a). First, the vision system process the saliency map (b). Then it selects an area of interest based on the saliency map and two thresholds: saliency and minimum length of area.

For the experiments, the *confidence threshold* (ϕ_c) of the tutelage mechanism was set to 2, and the *knowledge threshold* (ϕ_k) was set to 6. The ART2 neural network was set as it follows. The vigilance parameter (λ) was set to 0.999. Parameters a, b, c and d were set respectively to 10.0, 10.0, 0.1 and 0.9. The authors have found empirically that these parameters have produced better results for the experiments.

The experiments were composed by a presentation phase where the 4 fruits were presented under 5 different light conditions: *all lights on*, *natural lights*, *natural lights with a light source above the fruit*, *natural lights with a light source above and on the left of the fruit*, *natural lights with a light source above and on the right of the fruit*. Resulting in 20 presentations of the fruits for each presentation phase.

In order to evaluate the proposed learning mechanism, 5 measures were calculated during the experiments: *unknown rate*, *correct guess rate*, *incorrect guess rate*, *error rate* and *success rate*. The *unknown rate* is the frequency that the learning mechanism entered the *unknown mode*. The *correct guess rate* is the frequency that the learning mechanism entered the *uncertainty mode* and have correctly guessed the name of the fruit. The *incorrect guess rate* is the frequency that the learning mechanism entered the *uncertainty mode* and have incorrectly guessed the name of the fruit. The *error rate* is the frequency that the learning mechanism entered the *known mode*, but incorrectly pointed the name of the fruit. The *success rate* is the frequency that the learning mechanism entered the *known mode*, and pointed out the correct name of the fruit.

To quantify the learning capabilities of the proposed mechanism, the presentation phase was repeated 20 times (20 executions), varying the sequence of the light conditions. After each execution, the 5 measures were calculated and stored. Then, after the 20 executions, the average value and standard deviation, for each measure in the 20 executions, were calculated.

Table 1 shows the average values and standard deviation of the 5 measures for the 20 executions carried out during the experiments.

Table 1
Results obtained after the 20 executions of socially guided learning sessions.

Measure	Average rate (%)
Unknown rate	7.50 ± 4.93
Correct guess rate	15.00 ± 7.34
Incorrect guess rate	1.25 ± 2.64
Error rate	1.88 ± 2.03
Success rate	74.38 ± 7.48

4.2 Experiment #2

The second experiment is similar to the first one. The difference is the number of fruits presented to the robot (here we have employed 3 fruits: apple, lemon and orange). The presentation phase was repeated 5 times (5 executions) and we did not change the light conditions during the experiments.

To quantify the learning capabilities of the proposed mechanism, the presentation phase was repeated 5 times (5 executions). After each execution, the same 5 measures, mentioned in the experiment #1, were calculated and stored. Then, after the 5 executions, the average value and standard deviation, for each measure in the 5 executions, were calculated.

In Table 2 is showed the average values and standard deviation of the 5 measures for the 5 executions carried out during the experiments.

Table 2
Results obtained after the 5 executions of socially guided learning sessions.

Measure	Average rate (%)
Unknown rate	7.23 ± 0.58
Correct guess rate	17.47 ± 1.28
Incorrect guess rate	1.8 ± 0.44
Error rate	0.6 ± 0.2
Success rate	72.89 ± 2.19

The results show that the learning mechanism is able to exhibit appropriate behavior and to learn from social interactions. The learning mechanism was able to exhibit *knowledge*, *uncertainty*, and *certainty* about the names of the fruits, during the presentation phase, enabling a socially guided learning process in a more natural way. The use of certainty and uncertainty about an object enables the human being to determine exactly the understanding of the robot about the learned concept. The following example illustrates how a human being interacts with the robot according to its responses:

[human presents a red apple]
h Robot, what is this?
r I dont know.
h It is an apple.
r Ok.
h Very well!
. . .
[human presents a red apple]
h Robot, what is this?
r I am not sure. I think it is an apple.
h Very well!
. . .
[experimenter changes the lights condition]
[human presents a red apple]
h Robot, what is this?
r It is an apple.
h Very well!
. . .

The experiments have demonstrated how it is possible to turn the machine learning problem into a problem of collaboration among robots and human beings, using the human beings' natural social abilities to teach a robot.

From the experiments performed, we can conclude that both learning algorithms proposed and incorporated into the extended architecture are promissory tools to control social robots during interactions in a social environment.

5 Conclusions

In this paper, we extended our robotic architecture inspired on Behavior Analysis by adding a multimodal interface. A tutelage mechanism that is able to associate visual and auditory stimulus was proposed. For the learning of concepts about real world objects, an ART2 neural network is employed. Further, two different learning algorithms, Contingency and Economic TG algorithms, for controlling a robotic head, were incorporated to the robotic architecture. The results obtained show that the proposed tutelage mechanism is able to produce appropriate behavior, to learn from social interactions by human tutelage, and to generalize the learned knowledge. Further, they showed also that both learning algorithms proposed and incorporated into the robotic architecture can be adopted with success as learning mechanism. Future work includes the addition of new techniques of computer vision for the processing of shapes and textures. We intend also to extend the architecture by implementing new mechanisms and skills like, *long term interaction control*, *learning by imitation* and *verbal behavior*.

Acknowledgment

The authors would like to thank FAPESP and CNPq for support received.

References

[1] J. Aggarwal and Q. Cai. Human motion analysis: A review. *Computer Vision and Image Understanding*, 73(3):428−−440, 1999.

[2] H. Blockeel, L. D. Raedt, and J. Ramon. Top-down induction of clustering trees. In *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, pages 55–63, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[3] C. Breazeal. *Sociable Machines: Expressive Social Interaction Between Human and Robots*. PhD thesis, MIT Department of Electrical Engineering and Computer Science, Cambridge, MA, 2000.

[4] C. Breazeal. *Designing Sociable Robots*. MIT Press, 2002.

[5] C. Breazeal. Emotion and sociable humanoid robots. *Human-Computer Studies*, 59(1):119155, 2003.

[6] C. Breazeal. Toward sociable robots. *Robotics and Autonomous Systems*, 42(3−4):167−−175, 2003.

[7] C. Breazeal. Social interactions in hri: the robot view. *IEEE Transactions on Man and Cybernetics , Part C: Applications and Reviews*, 34(2):181−−186, 2004.

[8] C. Breazeal, A. Brooks, J. Gray, G. Hoffman, C. Kidd, J. Lieberman, A. Lockerd, and D. Mulanda. Humanoid robots as cooperative partners for people. *International Journal of Humanoid Robotics*, 1(2):1–34, 2004.

[9] C. Breazeal, D. Buchsbaum, J. Gray, D. Gatenby, and B. Blumberg. Learning from and about the others: Towards using imitation to bootstrap the social understanding of others by robots. *Artificial Life*, 11(1-2):31−−62, 2005.

[10] C. Breazeal and B. Scassellati. A context-dependent attention system for a social robot. In *International Joint Conference on Artificial Intelligence*, pages 1146−−1153, 1999.

[11] G. Carpenter and S. Grossberg. ART 2 : Self-organization of stable

category recognition codes for analog input paterns. *Applied Optics*, 26:23:4919−−4930, 1987.

[12] R. R. da Silva, C. A. Policastro, and R. A. F. Romero. Relational reinforcement learning applied to shared attention. In *Proceedings of 2009 International Joint Conference on Neural - To be published*, Atlanta, Georgia, USA, 2008. IEEE.

[13] K. Dautenhahn. Getting to know each other - artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems*, 16:333−−356, 1995.

[14] K. Dautenhahn. The art of designing socially intelligent agents - science, fiction and the human in the loop. *Applied Artificial Intelligence Journal*, 12(7−8):573−−617, 1998.

[15] K. Dautenhahn and A. Billard. Bringing up robots or−the psychology of socially intelligent robots: From theory to implementation. In *Autonomous Agents*, 1999.

[16] K. Dautenhahn, B. Ogden, and T. Quick. From embodied to socially embedded agents-implications for interactionaware robots. *Cognitive Systems Research*, 3(3), 2002.

[17] G. Dek and J. Triesch. Origins of shared attention in human infants. In K. Fuzita and S. Itakura, editors, *Diversity of Cognition (In Press)*, pages 67−−74. Kyoto University Press, 2005.

[18] K. Driessens. *Relational reinforcement learning.* PhD thesis, Department of Computer Science, K.U.Leuven, Leuven, Belgium., 2004.

[19] W. Dube, R. McDonald, R. Mansfield, W. Holcomb, and W. Ahearn. Toward a behavioral analisys of joint attention. *The Behavior Analyst*, 27(2):197−−207, 2004.

[20] B. Duffy, M. Dragone, and G. O'Hare. Social robot architecture: A framework for explicit social interaction. In *Android Science: Towards Social Mechanisms, CogSci 2005 Workshop*, Stresa, Italy, 2005.

[21] T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 60:143−−166, 2003.

[22] D. Goldberg. *Genetic Algorithm in Search, Optimization and Machine Learning.* Addison-Wesley Publishing Company, Inc., Massachusetts, 1989.

[23] M. Goodrich and A. Schultz. Human-robot interaction: A survey. *Foundations and Trends in Human-Computer Interaction*, 1(3):203−−275, 2007.

[24] S. Haykin. *Neural Networks - A Comprehensive Foundation.* Prentice Hall, 1999.

[25] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254−−1259, 1998.

[26] F. Kaplan and V. Hafner. The challenges of joint attention. In L. Berthouze, H. Kozima, C. Prince, G. Sandini, G. Stojanov, G. Metta, and C. Balkenius, editors, *4th International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic System*, pages

67−−74, 2004.

[27] M. Kearns and Y. Mansour. On the boosting ability of top-down decision tree learning algorithms. In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 459–468, New York, NY, USA, 1996. ACM.

[28] A. Lockerd and C. Breazeal. Tutelage and socially guided robot learning. In *International Conference on Intelligent Robots and Systems (IROS'2004)*, pages 3475−−3480, 2004.

[29] Lti-Lib. Lti image processing library - developers guide. Technical report, Lehrstuhl fuer Technische Informatik - Aachen University of Technology, 2003.

[30] Y. Marom and G. Hayes. Attention and social situatedness for skill acquisition. Technical Report EDI-INF-RR-0069, University of Edinburgh, 2001.

[31] A. K. Mccallum. Learning visual routines with reinforcement learning. In *AAAI Fall Symposium*, pages 82–86, 1996.

[32] T. Mitchell. *Machine Learning.* McGraw-Hill, 1997.

[33] L.-P. Morency. User guide - head tracking and gesture recognition library. Technical report, MIT Department of Electrical Engineering and Computer Science, 2007.

[34] L.-P. Morency, P. Sundberg, and T. Darrel. Pose estimation using 3d view-based eigenspaces. In *Analysis and Modeling of Faces and Gestures (AMFG'03)*, pages 45−−52. IEEE, 2003.

[35] Y. Nagai, A. Hosoda, and M. Asada. A constructive model for the development of joint attention. *Connection Science*, 15(4):211−−229, 2003.

[36] Nuance. Introduction to the nuance system. Technical report, Nuance Communications, Inc., 2001.

[37] C. Policastro, G. Zuliani, R. da Silva, V. Munhoz, and R. Romero. Hybrid knowledge representation applied to the learning of the shared attention. In *IEEE World Congress on Computational Intelligence (WCCI'08), Hong Kong. To be published.* IEEE, 2008.

[38] C. Policastro, G. Zuliani, and R. Romero. Robotic architecture inspired on behavior analysis. In *IEEE International Joint Conference on Neural Network (IJCNN07), Orlando, Florida, USA*, pages 1482−−1487. IEEE, 2007.

[39] C. A. Policastro. *Arquitetura robtica inspirada na anlise do comportamento.* PhD thesis, Universidade de So Paulo, Outubro 2008.

[40] B. Robins, P. Dickerson, P. Stribling, and K. Dautenhahn. Robot-mediated joint attention in children with autism: A case study in robot-human interaction. *Interaction Studies*, 5(2):161−−198, 2004.

[41] D. Robot. Animated head system with camera, servo & multimedia controller board wha8030.

[42] R. A. F. Romero, J. F. Vicentini, P. R. Oliveira, and A. M. J. Traina. Investigating the potential of art neural network models for indexing and information retrieval: Research articles. *Int. J. Intell. Syst.*, 22(4):319–

336, 2007.

[43] R. R. SILVA, C. A. Policastro, and R. A. F. Romero. An enhancement of relational reinforcement learning. In *Proceedings of 2008 International Joint Conference on Neural*, pages 2056–2061, Hong Kong, 2008. IEEE.

[44] R. Sutton and A. Barto. *Reinforcement Learning: An Introduction.* MIT Press, 1998.

[45] S. Waldherr, R. Romero, and S. Thrun. A gesture based interface for human-robot interaction. *Autonomous Robots*, 9:151–173, 2000.