

# Robot imitation: Body schema and body percept

doi:10.1533/abbi.2004.0043

C. A. Acosta-Calderon and H. Hu

*Department of Computer Science, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, United Kingdom*

**Abstract:** There are two functional elements used by humans to understand and perform actions. These elements are: the body schema and the body percept. The first one is a representation of the body that contains information of the body's capabilities. The second one is a snapshot of the body and its relation with the environment at a given instant. These elements interact in order to generate, among other abilities, the ability to imitate. This paper presents an approach to robot imitation based on these two functional elements. Our approach is gradually expanded throughout three developmental stages used by humans to refine imitation. Experimental results are presented to support the feasibility of the proposed approach at the current stage for 2D movements and simple manipulation actions.

**Key words:** Robot imitation, body representation, action representation.

## INTRODUCTION

In order to perform an action, lifting a box for example, we require information from our body parts. Imagine that you are standing and wish to pick up a shoe box lying at your feet. You can easily bend your waist and knees until you can grasp the box with your hands. Once the box is grasped you can straighten back up to the standing position. The information that our bodies require to perform an action, like the one above, is frequently divided into two sources (Reed 2002): the "body schema", which contains the relations of the body parts and its physical constraints, and the "body percept", which refers to a particular body position perceived in an instant.

The body schema and the body percept give us an insight into how other people perform actions. Since the knowledge of feasible actions and physical constraints is implicit in the body schema, it is possible to do a mental rehearsal of our actions and gather the results of those actions at particular body percepts. At some extent, it is possible to simulate other peoples' actions. The previous idea is feasible but only after the individual has established a certain "level of similarity" to imitate (Alissandrakis et al 2004). Therefore, the understanding of similarities (e.g. body, actions, or effects) in other people facilitates imitation. The body schema and the body percept give us the insight into recognizing this level of similarity and thereby

performing movements and actions to achieve this similarity. It is believed that the body schema and the body percept contribute through different developmental stages to the imitation ability (Reed 2002, Rao and Meltzoff 2003).

Imitation, the ability to recognize, learn and copy the actions of others, rises as a very promising alternative solution to the programming of robots. It remains a challenge for roboticists to develop the abilities that a robot needs to perform a task while interacting intelligently with the environment (Bakker and Kuniyoshi 1996, Acosta-Calderon and Hu, 2003a). Traditional approaches to this issue, such as programming and learning strategies, have been demonstrated to be complex, slow, and restricted in knowledge.

Learning by Imitation will enhance the robots repertoire of abilities especially through human-robot interaction. Hence, robots might eventually help humans in their daily personal tasks (Acosta-Calderon and Hu, 2004a). Additionally, imitation presents some desirable characteristics for robotic systems such as effective learning, transfer of implicit knowledge, and reduced cost of programming (Acosta-Calderon and Hu, 2003b).

This paper presents an approach for robot imitation based on the two fundamental parts used by humans to acquire information, understand and execute actions: the body schema and the body percept. The interaction of these two parts are presented throughout different stages like those presented in the development of the imitative abilities in humans. The presented approach in this paper addresses imitation of 2D movements, handwriting, and simple manipulation actions.

The rest of the paper is organized as follows. The second section presents the background theory that has inspired our work on imitation. The next sections present the three stages of our imitation approach, and implementation

---

*Corresponding Author:*

C. A. Acosta-Calderon  
Department of Computer Science  
University of Essex, Wivenhoe Park,  
Colchester CO4 3SQ, United Kingdom  
Email: caacos@essex.ac.uk

---

issues are also offered. The third section discusses the body configuration and its importance for building the body schema. In the fourth section, it is described how the interaction between body schema and body percept produces imitation of body movements. The fifth section gives details of the use of environmental states to achieve imitation of body movements. Experimental results are presented in the sixth section. Finally, the last section concludes the paper and outlines the future work.

## FOUNDATION

The understanding of imitation on humans could give an insight of the processes involved and how this phenomenon could be commenced in robots. The intrinsic relation of human actions and human bodies is a good starting point. Humans can perform actions that are feasible with their bodies. In addition, humans are aware of which actions are feasible with their own bodies just by observing an action to be performed. Hence, the information used by the human body in order to carry out an action is derived from two sources (Reed 2002):

- The body schema is the long-term representation between the spatial relations among body parts and the knowledge about the actions that they can and cannot perform.
- The body percept refers to a particular body position perceived in an instant. It is built by instantly merging information from sensory sources, including visual input, and proprioception, with the body schema. It is the awareness of a body's position at any given moment.

The information from the body percept is instantly combined with that from the body schema to identify an action. For instance, by observing someone (with similar body) lifting a box, one can easily determine how one would accomplish the same task using one's own body. This means that it is possible to recognize the action that someone else is performing.

The body schema, a body representation with all its abilities and limitations, along with the representation of other objects can simulate any given action, for instance the box lifting example. Imagine that you see someone lifting a box. The body schema allows you to understand what someone else is doing and, for instance, how you would go about picking the box up yourself. It is possible then to indicate the main two functions of the body schema:

- Direct function is the process of executing an action that produces a new state.
- Inverse function is looking for an action that satisfies a goal state from a current state.

These two functions share the idea that has been used in motor control, but they are known as controllers and predictors (Wolpert and Flanagan 2001). The work of Demiris and Johnson in (Demiris and Johnson 2003) used functions with the same principle but they called them inverse and forward models.

The interaction of both functions allows simulating another person's actions (Goldman 2001). The inverse function generates the actions and movements that would achieve the goal, the actions and movements are sent to the direct action, which will predict the next state. This predicted state is compared with the target goal to take further decisions. The interaction between the body schema and the body percept permit us to understand the relationships between ourselves and other's bodies (Reed et al 2004; Viviani and Stucchi 1992). Thus, in order to achieve a perceived action a mental simulation is performed constraining the movements to those that are only physically probable.

The body schema provides the basis to understand similar bodies and perform the same actions (Meltzoff and Moore 1994). This idea is essential in imitation. In order to imitate, it is first necessary to identify the observed actions, and then be able to perform those actions. Psychologists believed these two elements interact between them through several developmental stages causing human imitative abilities. One attempt to explain the development of imitation is given by Rao and Meltzoff, who had introduced a four-stage progression of the imitative abilities (Rao and Meltzoff 2003), details of each stage are presented below:

- (1) *Body babbling*. This is the process of learning how specific muscle movements achieve various elementary body configurations. Thus, such movements are learned through an early experimental process, e.g. random trial-and-error learning. Because, both the dynamic patterns of movements and the resulting end-states achieved can be monitored proprioceptively, body babbling can build up a map of movements to end-states. Thus, Body babbling is related to the task of constructing the body schema (the system's physics and constrains).
- (2) *Imitation of body movements*. This demonstrates that a specific body part can be identified i.e. organ identification (Meltzoff and Moore 1992). This supports the idea of an innate observation-execution pathway in humans (Charminade et al 2002). The body schema interacts with the body percept to achieve the same movements, once these are identified.
- (3) *Imitation of actions on objects*. The ability to imitate the actions of others on external objects undoubtedly played a crucial role in human evolution. This is done by facilitating the transfer of knowledge of tool use and other important skills, from one generation to the next. This also represents flexibility to adapt actions to new contexts.
- (4) *Imitation based on inferring intentions of actions*. This requires the ability to read beyond the perceived behavior to infer the underlying goals and intentions. This involves visual behaviors and internal mental states (intentions, perceptions, and emotions) that underlie, predict, and generate these behaviors.

Imitation in robotics has been tackled for few years now. The main reason behind this is the promise of an alternative solution to the problem of programming robots (Bakker and Kuniyoshi, 1996), to fill the gap to human-robot

interaction (Dautenhahn and Nehaniv 2002; Becker et al 1999), and finally, to obtain new abilities by observation (Acosta-Calderon and Hu, 2003b).

Related work on imitation using robotics arms focus on reproducing the exact gesture while minimizing the discrepancy for each joint (Ilg et al 2003; Zollo et al 2003; Schaal et al 2003). The work described here uses a different approach; it focuses only on the target (i.e. demonstrator's hand) and to allow the imitator to obtain the rest of the body configuration (Acosta-Calderon and Hu 2005). The four developmental stages presented above serve as a guideline for the progress in this research. This paper reviews the experiences of a robotic system with the first three stages: body babbling, imitation of body movements, and imitation of actions on objects.

## BODY CONFIGURATION

Body babbling endows us, humans, with the elementary configuration to control our body movements by generating a map. This map contains the relation of all the body parts and their physical limitations. In other words, this map is essentially the body schema. At this stage, the body schema is learned generally through a process of random experimentation of body movements with their achieved end-states. When humans grow and their bodies change, the body schema is constantly updated by means of the body percept. The body percept, in turn, gathers its information from sensory sources. If there is an inconsistency between the body schema and the body percept, then the body schema is updated.

In robotics, since the bodies of robots are normally changeless in size and weight, body babbling should be simpler. Therefore, the proposed approach here is to endow a robot with a control mechanism that will permit the robot to know its physical abilities and limitations. Some other approaches in contrast try to follow nature's design in this matter to permit a robot to build its body schema by random learning, which would take much time to converge into a good control mechanism, without mentioning the handling of redundancy (Lopes et al 2005; Schaal et al 2003). In the following section the robotic platform used in our experiments is introduced and its control mechanism is described.

### The robot hardware and control mechanisms

The robotic platform used to validate our approach is a mobile robot Pioneer 2-DX with a Pioneer Arm (PArm) and a camera, namely United4 (see Figure 1). The robot is a small, differential-drive mobile robot intended for indoors. The robot is equipped with the basic components for sensing and navigation in a real-world environment. These components are managed via microcontroller board and the onboard server software. The access to the onboard server is through an RS232 serial communication port from a client workstation. United4 is also equipped

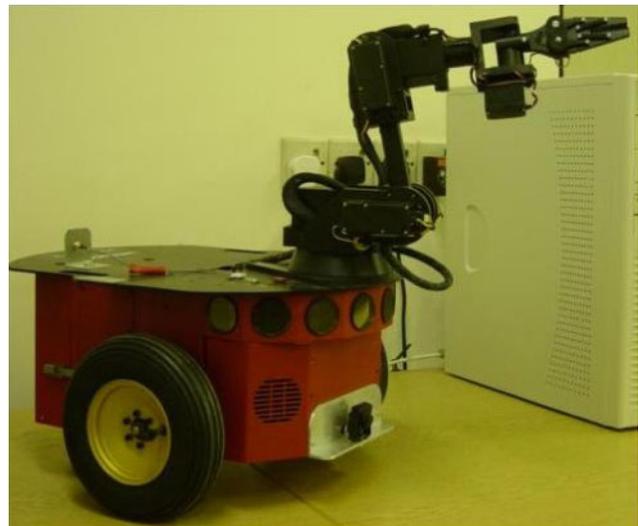


Figure 1 The robot United4 is the robotic platform we used. It is a Pioneer 2DX with a Pioneer Arm and a camera.

with a vision system and a color tracking system. The vision system consists of one camera placed at the “nose” of the robot, which captures the images to be processed. The color tracking system provides information about the position and size of the colored objects. Color simplifies the detection of the relevant features in the application.

The PArm is an accessory for the Pioneer robot, which is used in research and teaching. This robotic arm is five degrees of freedom (DOF); its end-effector is a gripper with fingers allowing for the grasping and manipulation of objects. The PArm can reach up to 50 cm from the center of its base to the tip of its closed fingers. All the joints of the arm are revolute with a maximum motion range of 180°. To control the arm, we use kinematic methods. These methods are dependent on the type of links that compose the robotic arm or manipulator, and more importantly they are determined by the way they are connected.

A situated machine, like United4, interacts with the environment by changing the values of its effectors, which produces a movement in space to reach a new position. Therefore, position control mechanism is required, which would allow the robot to know its position in time, and to place the robot in a desired position. The control mechanism used here is based on the kinematic analysis, which studies the motion of a body with respect to a reference coordinate system, without considering speed, force, or other parameters influencing the motion (Mckerrow 1991). A type of this kinematics is the “forward kinematics”. The forward kinematic analysis permits to calculate the position and orientation of the end-effector of the robotic arm, when its joint variables have changed.

$$n_x = -C_5(C_1S_2C_3C_4 - S_1S_4) - C_1C_2C_3S_5$$

$$n_y = -C_5(S_1S_2C_3C_4 + C_1S_4) - S_1C_2C_3S_5$$

$$n_z = C_2C_3C_4C_5 - S_2S_5$$

$$\begin{aligned}
o_x &= S_1 C_4 + C_1 S_2 S_4 \\
o_y &= -C_1 C_4 + S_1 S_2 S_4 \\
o_z &= -C_2 S_4 \\
a_x &= C_1 C_{23} C_5 - S_5 (C_1 S_2 S_4 - S_1 S_4) \\
a_y &= S_1 C_{23} C_5 - S_5 (S_1 S_2 S_4 + C_1 S_4) \\
a_z &= S_2 S_4 + C_2 S_4 \\
p_x &= a_1 C_1 + a_2 C_1 C_2 + d_4 C_1 C_{23} + d_6 (C_1 C_{23} C_5 \\
&\quad - S_5 (C_1 S_2 S_4 - S_1 S_4)) \\
p_y &= a_1 S_1 + a_2 S_1 C_2 + d_4 S_1 C_{23} + d_6 (S_1 C_{23} C_5 \\
&\quad - S_5 (S_1 S_2 S_4 + C_1 S_4)) \\
p_z &= a_2 S_2 + d_4 S_2 + d_6 (S_2 C_5 + C_2 S_4)
\end{aligned} \tag{1}$$

The forward kinematics as we mentioned before, calculates the position and orientation of the end-effector from a set of joint values. Equation (1) presents the forward kinematic equations to calculate the position and orientation of the PArm. Despite calculating the position from a given joint values, it is necessary for the robot to determine its joints values to reach a desired position and orientation. There are different methods to achieve this. The method used by United4 is called the resolve motion rate control (RMRC), which performs smooth movements of the arm from the current position of the end-effector to a desired position (Klafter et al 1989). This method is for continuous paths, which means that the end-effector of the robotic arm reaches one target and it does not have to change its entire joint configuration to find a new position. Instead, RMRC tries to minimize the kinetic energy, which causes less movements of the joints configuration.

In order to know the joint's velocities that are necessary to accomplish a change of position in the end-effector, we use the inverse Jacobian. We can also introduce a second criteria  $\phi$  to be minimized subject to the primary position task by using the *global solution*.

$$Dq = \mathcal{J}^+ D + (I - \mathcal{J}^+ \mathcal{J}) \phi \tag{2}$$

where  $\mathcal{J}^+$  is the pseudoinverse matrix of  $\mathcal{J}$ . The reason for using the pseudoinverse matrix is because the PArm's Jacobian matrix is not a square. In equation (2),  $\mathcal{J}^+$  is used along with a function that tries to maintain the joint values within the center of its limits, while it calculates the incremental changes in the joint variables  $\theta_i$ , which could produce the desired incremental change in the end-effector's location. Details of the kinematics equations and the RMRC for the PArm can be found at (Acosta-Calderon and Hu, 2004a; Acosta-Calderon and Hu, 2004b).

## IMITATION OF BODY MOVEMENTS

The imitation described in this paper will focus exclusively to the imitation of the arm movements. The level of

imitation for this stage is defined as the reproduction of the path followed by the target (Billard et al 2004; Dautenhahn and Nehaniv 2002; Nehaniv and Dautenhahn 2002). Thus, the imitator will only follow the path described by the end-effector of the demonstrator i.e. the hand of the human demonstrator. This level of imitation finds supported in way that humans observe a movement with the intention to imitate. When humans observe a body movement, e.g. observing someone waving the hand; they do not focus their attention on every body part. Instead, humans choose as the point of fixation the end-effector (i.e. the human hand), while the peripheral vision and the body schema help them to keep track of the position of the other body parts (shoulder, elbow, etc.) (Mataric and Pomplun 1998; Mataric 2002). When humans fix their gaze in the end-effector they rely on the body schema, which along with the body percept find the necessary body configuration for the rest of human body parts, automatically satisfying the target position for the end-effector. Hence, the level of the reproduction of the exact gesture was not central because our approach allows the body schema to find the body configuration satisfying the target position. The discrepancy among the bodies of the imitator and the demonstrator is another reason that the reproduction of the exact gesture is not intended. This issue is called the correspondence problem.

### Correspondence problem

If a child would like to imitate a body movement, e.g. a waving hand, the child would focus on the position of the hand, while the rest of all the other body parts are not attended directly. The body schema allows the child to do this because it finds the correct configuration for all the body parts satisfying the desired position. In this way, the child is able to avoid the hassle of concentrating on every single part of the body that the child is observing.

The above example works, when the demonstrator and the imitator have a common body representation. This implies that the body schema of the imitator is, by itself, capable enough to understand the demonstrator's body. In a situation, where the demonstrator's body differs from the imitator's body, the imitator's body schema would not be able to understand the demonstrator body representation; this situation is called the correspondence problem (Nehaniv and Dautenhahn 2001; Nehaniv and Dautenhahn 2002).

The imitator can overcome the correspondence problem by finding a certain level of similarity between the bodies. This level of similarity is intrinsically related to metrics, which capture the aspects of the demonstration (Alissandrakis et al 2004; Nehaniv and Dautenhahn 2001). Hence, the imitator can look for a level of similarity that involves body positions, actions, and states according with its own objectives.

For our implementation, this correspondence problem is presented due to the discrepancy of the bodies of the

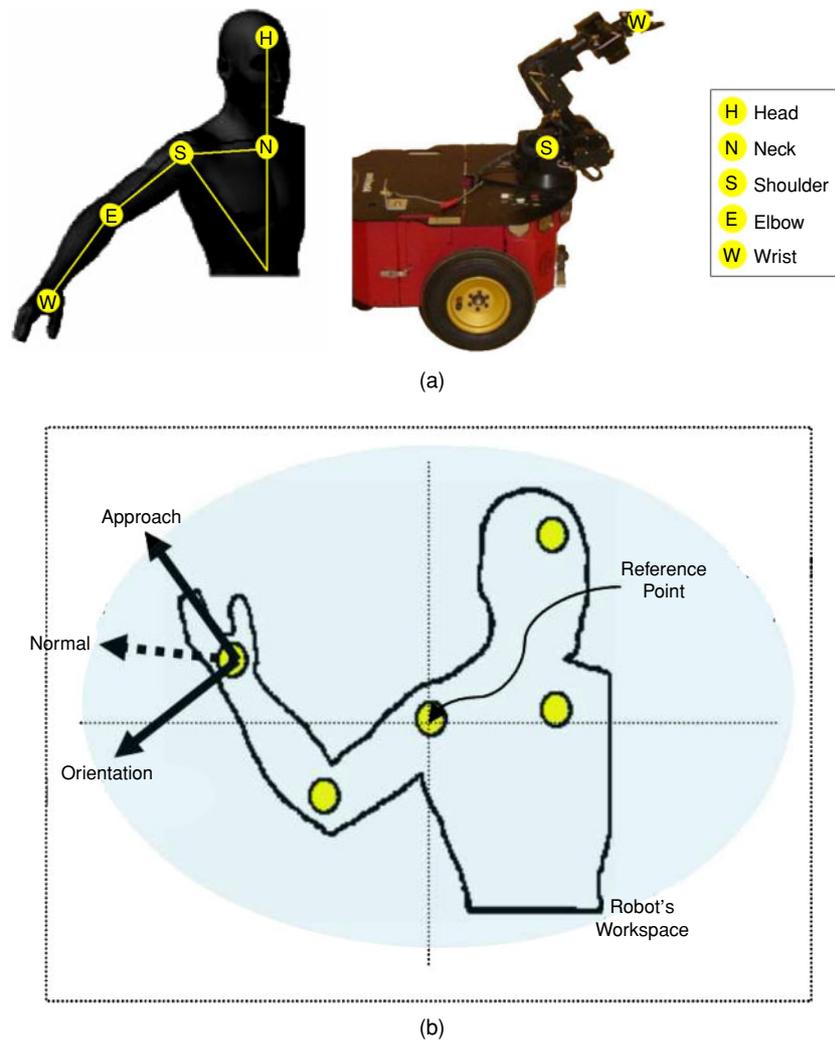


Figure 2 (a) The correspondence between the bodies of the demonstrator (left) and the robot (right). Two joints, the shoulder and the wrist, have correspondence in both bodies. (b) The relation between the image coordinate frame (XY) and the robot coordinate frame (YZ).

robot as imitator and the human as demonstrator. To work out this situation, the robot is provided with the representation of the body of the demonstrator and a way to relate this representation to its own (Acosta-Calderon and Hu, 2004a).

Figure 2a presents the correspondence between the body of the demonstrator and that of the imitator. Here, a transformation is used to relate both representations. This transformation is based on the knowledge that in the set of joints of the demonstrator there are three points that represent an arm (shoulder, elbow, and wrist). The remaining two points (the head and the neck) are used just as a reference. This information about the representation of the demonstrator is extracted by means of color segmentation.

The transformation relates the demonstrator's body to the robot's body. Thus, the final result of this transformation is a point in the robot's workspace represented as follows:

$$p_i = [x_i, y_i, z_i, \phi_i, \theta_i, \psi_i] \quad (3)$$

where the location of the robot's end-effector is specified in terms of both the position (defined in Cartesian coordinates by  $x$ ,  $y$ , and  $z$ ) and the orientation (defined by the *roll*, *pitch*, and *yaw* angles  $RPY(\phi, \theta, \psi)$ ).

The reference points are used to keep a relation among the distances in the demonstrator model. The shoulder is the origin of the workspace of the robot. Thus, the shoulder of the demonstrator is used to convert the position of the remaining two points of the demonstrator's arm. The new position of the demonstrator's end-effector (i.e. wrist) is then calculated, and finally it is fitted into the actual workspace of the robot (see Figure 2b).

We need to keep in mind that the coordinate systems for the image perceived and for the robot are different. The main difference is that the robotic system is 3D whereas the image is just 2D. Thus, we consider only movements in the plane YZ of the arm (i.e.  $X$  is set to a fixed value), and then we would be able to match the plane XY of the image with the plane YZ of the arm (see Figure 2b) by

using simple transformations. Those transformations have to be applied in every point of that the image that we would like to convert to the arm frame.

$$\begin{bmatrix} y_i \\ z_i \end{bmatrix} = \begin{bmatrix} -k(xIM_i - xIM_{ref}) \\ -k(yIM_i - yIM_{ref}) \end{bmatrix} \quad (4)$$

Since, this arm representation must fit into the actual arm workspace; the  $k$  parameter encloses the relation of one pixel in the image to the working unit for the arm. We use a negative value of  $k$  in equation (4) to invert the  $Y$ -axis of the  $XY$  image plane and to be coplanar with the  $Z$ -axis of the  $YZ$  robot plane. This also produces a “mirror effect” on the value of  $Y$  for each point. Hence, if the model is located in front of the robot and moves its left arm, then the imitator would move the arm in the right side, acting as a mirror.

The orientation of the demonstrator’s hand is obtained by inverse solution described in equation (5). To compute these angles the unitarian orthogonal vectors  $[\vec{n}, \vec{a}, \vec{o}]$  need to be assigned.

$$\begin{aligned} \phi &= \arctan 2(n_y, n_x) \\ \theta &= \arctan 2(-n_z, n_x \cos \phi + n_y \sin \phi) \\ \psi &= \arctan 2(o_z, a_z) \end{aligned} \quad (5)$$

Since the demonstrator’s hand is assumed to move on a plane (i.e.  $YZ$ ), then the orientation of the demonstrator’s hand could be described in Figure 2b. The approach vector  $\vec{a}$  is coming out of the demonstrator’s hand and it is collinear to the line described by the wrist and the elbow

$$va = \text{Wrist}_i - \text{Elbow}_i \quad (6)$$

and the unitarian representation is

$$\vec{a} = \frac{va}{\text{Norm}(va)} \quad (7)$$

The normal vector  $\vec{n}$  is perpendicular to the plane  $YZ$ , and it has a fixed value

$$\vec{n} = [-1, 0, 0] \quad (8)$$

Thus, the orientation vector  $\vec{o}$  can be obtained by the cross product of  $\vec{a}$  and  $\vec{n}$  due to the orthogonal property of the three vectors

$$vo = \vec{a} \times \vec{n} \quad (9)$$

and the unitarian representation is

$$\vec{o} = \frac{vo}{\text{Norm}(vo)} \quad (10)$$

Finally, with the three vectors the orientation angles can be calculated by equation (5).

### Mechanism of imitation

The body movements studied here only involved arm movements. The mechanism designed to achieve imitation of body movements relies on two central issues:

1. The imitator focuses only on the position of the end-effector, which will define the path to follow

2. The imitator’s body schema and body percept solve the body configuration to achieve each new position on the path.

The robotic scenario is set-up like this, the robot imitator observed a human demonstrator performing arm movements. The human demonstrator wore color markers to bring out the relevant features, which are extracted and tracked by the color tracking system. The obtained information is then used to solve the correspondence problem as described in the previous section. The reference point used to achieve this correspondence is the shoulder of the demonstrator, which corresponds to the base of the PArm. The robotic arm and the human arm share neither the same number of degrees of freedom nor the workspace. These physical differences between the robot and the demonstrator made an exact copy of their gestures impossible, apart from following the position of the end-effector. When a representation of the body of the demonstrator is provided, it is possible for the robot to obtain the position that corresponds to that of the demonstrator (Acosta-Calderon and Hu, 2004b).

Movement imitation is not as trivial as copying an observed movement onto one’s own workspace. The process can be more complex as explained next while describing how the mechanism works (see Figure 3). The color tracking system keeps feeding the mechanism with the position of the markers. The body percept uses this information along with the body schema of the demonstrator to calculate the position of the demonstrator’s end-effector. The body percept is in charge of merging these data to estimate the pose of the demonstrator, and keep tracking of it, when it changes. The body percept uses the demonstrator’s body schema to overcome the correspondence problem, and eventually, the coordinate transformation from an external workspace to the egocentric workspace.

Each new position of the end-effector transformed into the workspace of the robot is sent to the body schema of the robot. The body schema employs the data send by the body percept as a target to be satisfied by the inverse function (i.e. RMRC). This function obtains the new values for the body parts to satisfy the desired position. The inverse function resolves the redundancy of unconstrained degrees of freedom by applying the following criteria.

- The satisfaction of the desire position of the end-effector.
- The minimization of the kinetic energy between positions.
- Maintaining the joints inside their working limits.

The body configuration obtained for the robot might not be same as the one presented by the demonstrator, but a similar position of end-effector is achieved in the robot’s workspace. Here, the body schema plays a crucial role in minimizing the motion trajectory, while considering the physical constrains, and selecting the more efficient body configuration.

Once a body configuration has been calculated, it is sent to the direct function (i.e. forward kinematics). The direct function will use the new body configuration calculated by

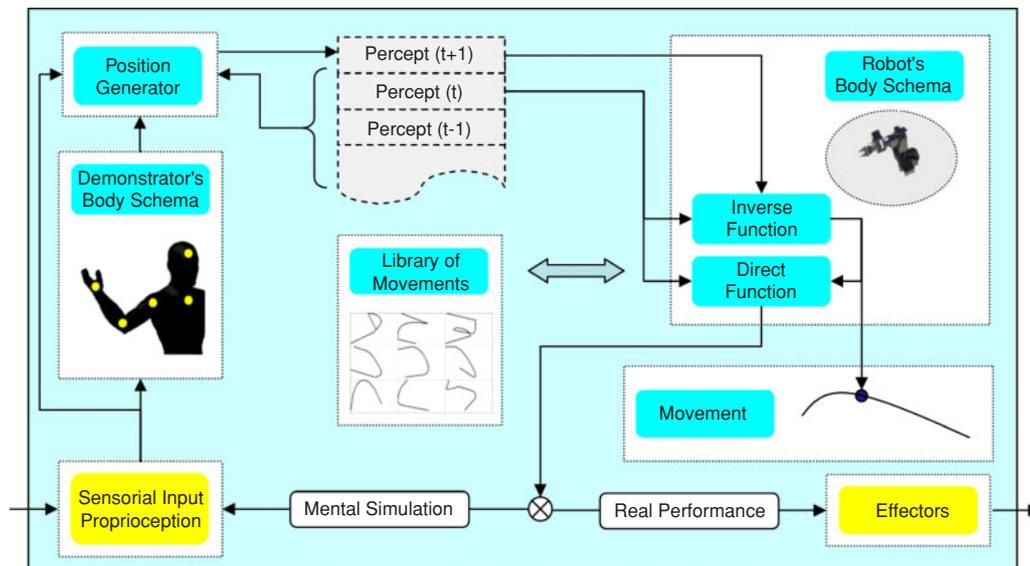


Figure 3 The mechanism of imitation for body movements.

the inverse function along with the current state from the body percept (body percept ( $t$ )) to predict the new position of the end-effector. The predicted position of the end-effector is then compared with that used as a desired target (body percept ( $t + 1$ )). This comparison decides whether the new calculated body configuration is appropriate to achieve the desired position. Otherwise, the inverse function will need to solve the desired position again or the position might not be reachable by the robotic arm.

The use of the direct function to follow the state or results of what the imitator is observed, is what Demiris and Hayes (Demiris and Hayes 2002) called active imitation. Active imitation contrasts from passive imitation, where the imitator observes, recognises the complete sequence of actions and then performs the whole sequence.

The body configuration can now be sent to the motor control system, which modularizes the signals for the actuators. The mechanism until now shows an immediate imitation, which means that the imitator can only reproduce the gesture when observing the demonstration. The introduction of a memory will produce deferred imitation, where the imitator is able to reproduce the copied gesture even if the demonstrator is not executing the gesture or even if the demonstrator is not present. Therefore, the desired position is added to the representation of the movement, which can be used later for a performance of the learnt movements, or for classification. This memory requires a suitable representation of the movements.

### Movements and library

When the demonstrator is performing a movement, the imitator extracts a discrete set of points, which can be seen as an abstraction of the movement. Here, each point represents both the position (defined in cartesian coordinates by  $x$ ,  $y$ , and  $z$ ) and the orientation (defined by the roll, pitch and yaw angles) of the demonstrator's end-effector. These

points are smoothed by using cubic spline curves, which have the feature that they can be interrupted at any point and fit smoothly to another different path. More points can be added to the curve without increasing the complexity of the calculation. While the position of the movement ( $x$ ,  $y$ , and  $z$ ) is smoothed by the spline, the orientation (roll, pitch and yaw angles) is interpolated.

Once a movement is smoothed and fitted by a spline curve, the next step is to identify whether the observed movement has been learned previously or it is a novel movement. However, the identification of a movement is a complex process. The demonstration of a movement will never be exactly same during its repetition. However, the demonstrator may produce movements that share similar features. These features provide the way to classify the movements and to identify to which class a new movement corresponds.

### Feature extraction

The selection of the relevant features of a movement is always a tough job. The features selected here do not require too much computational processing but they are still useful to differ the crucial points in the spline.

A movement  $\vec{S}p$  defined by equation (11) is a sequence of discrete points  $Sp_i$ , where each point is expressed as  $Sp_i = [x, y]$  (Despite the actual spline uses a 3D representation, here each point of the spline is expressed in a plane to simplify the description methodology implemented).

$$\vec{S}p = [Sp_1, \dots, Sp_N] \quad (11)$$

In our feature selection, the movement  $\vec{S}p$  is transformed into a sequence of feature vectors  $\vec{F}$  defined in equation (12). Each feature vector is defined as  $f_i = [x_{f_i}, y_{f_i}, \theta_{f_i}]$ , where the feature  $f_i$  is a triplet formed of the position  $x_{f_i}$ ,  $y_{f_i}$ , and the angle  $\theta_{f_i}$ . These values represent a discontinuity of motion in the spline.

$$\vec{F} = [f_1, \dots, f_M] \tag{12}$$

Since  $\theta_i$  is a directional quantity, a special treatment for the computation of probabilities, statistics, and distances, is necessary. Each feature  $f_i$  is selected as follows:

- (1) For each point, it is necessary to calculate  $\gamma$  and sign by using the following equations:

$$\gamma_j = \left\| \arctan \left( \frac{y_{j+1} - y_j}{x_{j+1} - x_j} \right) \right\| \tag{13}$$

$$\text{sign}_j = \text{sign}(\gamma_j) \tag{14}$$

- (2) The slope is then computed by using equation (15). The slope represents the slope of the  $S_p$  line, basically going up (value  $-1$ ) or going down (value  $1$ ). In contrast with  $\gamma$  and sign, slope is not calculated for each point but only at the beginning of the  $S_p$  line and when a feature is selected.

$$\text{slope} = \begin{cases} \gamma_j < \gamma_{j-1}, & 1 \\ \gamma_j > \gamma_{j-1}, & -1 \end{cases} \tag{15}$$

- (3) With all these values a feature is selected if any of the following conditions are met.

$$\begin{aligned} & \text{sign}_j \neq \text{sign}_{j-1} \\ & (\gamma_j < \gamma_{j+1}) \text{ AND } (\text{slope} > 0) \\ & (\gamma_j > \gamma_{j+1}) \text{ AND } (\text{slope} < 0) \end{aligned} \tag{16}$$

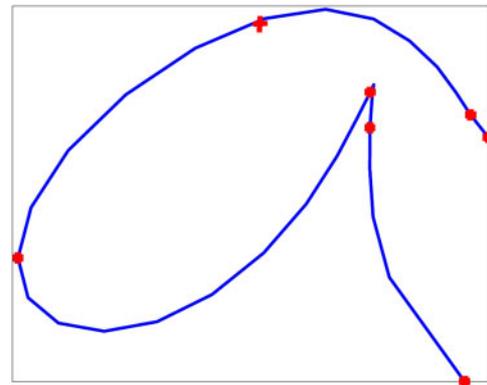
It should be noticed that, the number of features  $M$  is not equal to the number of points in the  $S_p$  line  $N$ . In addition, the number of features might vary from movement to movement. Figure 4 shows two movements and their respective features denoted by the dots on the  $S_p$  line. Despite both Figures 4a and 4b are similar; Figure 4a contains only five extracted features, whereas Figure 4b contains six extracted features. This situation makes the identification and classification a complicated job.

**Identification and classification**

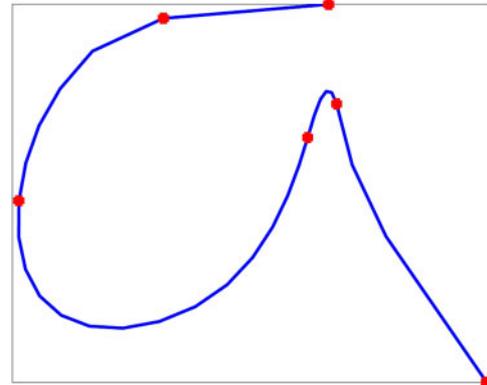
After feature extraction a movement is now represented by a sequence of feature vectors as depicted in equation (4). A matching process is used to find the movement, whose template gives the best fit to the observed movement. Let a library's template consist of another sequence of feature vectors:

$$\overrightarrow{\text{Class}}_k = [f_1, \dots, f_T] \tag{17}$$

where  $\overrightarrow{\text{Class}}_k$  is the template  $k$  in the library and each feature vector is still defined as  $f_i = [x_{fi}, y_{fi}, \theta_{fi}]$  as same as equation (12). However, it is important to remark that the template in the library is not a single movement but instead is a sequence of clusters, where each cluster represents a single feature of the template (see Figure 5). Therefore, here the feature  $f_i$  is a triplet formed of the mean values of position  $x_{fi}$ ,  $y_{fi}$ , and the angle  $\theta_{fi}$  of the elements in the cluster  $i$ .



(a)



(b)

Figure 4 Two similar movements of the handwriting of the letter “a” (solid line), and their extracted features (circle markers).

The matching steps to identify to which class belong the movement  $\vec{F}$  are described below:

1. For each template  $k$  in the library
- (4) The distance between all the points of  $\vec{F}$  and  $\overrightarrow{\text{Class}}_k$  is calculated by using:

$$d = d\text{DTW}(\overrightarrow{\text{Class}}_k, \vec{F}) \tag{18}$$

where  $d\text{DTW}$  is the dynamic time warping (DTW) distance, which is defined for two particular feature vectors used here as equation (19). DTW is a method that allows an elastic match of two sequences used in speech recognition (Rabiner and Juang 1993). This method produces an optimal alignment (or warping) path ( $\Phi$ ), which defines an alignment of corresponding regions in both sequences; by means of a local distance function.

$$\begin{aligned} d\text{DTW}(E, O) = & \frac{\sqrt{(x_e - x_o)^2 + (y_e - y_o)^2}}{\alpha} \\ & * \frac{||\theta_e - \theta_o||}{\beta} \end{aligned} \tag{19}$$

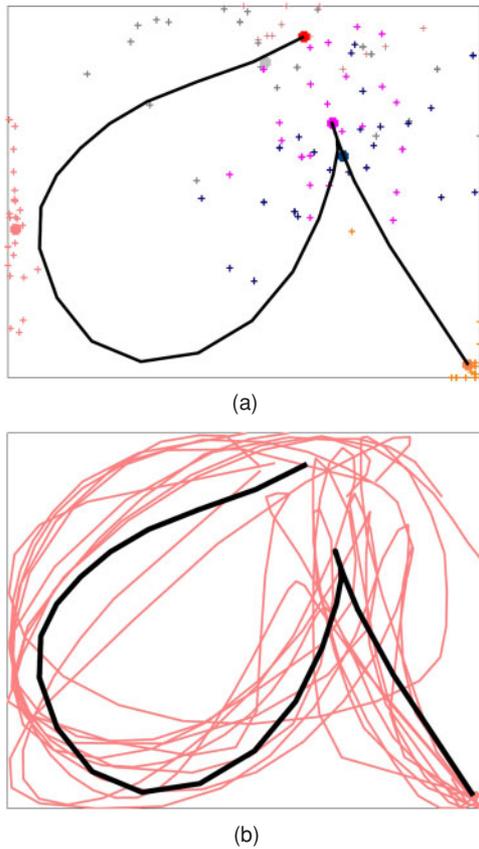


Figure 5 (a) A template in the library is represented as a sequence of clusters. The mean values of each cluster (circular markers) are used as a feature vector. The unique representation for this template is the solid line. (b) Shows the samples that form the template, and the unique representation.

- (5) The matrix of distances generated by equation (10) is then used to obtain the alignment path ( $\Phi$ ). This alignment path is generated by dynamic programming solution to Time Wrapping pattern matching problem (see equation (21)). This particular solution employs the Sakoe–Chiba algorithm, which only allows forward steps of size one in either one direction, or in both of them.

$$\Phi = \text{Sakoe} - \text{Chiba}(d) \quad (20)$$

$$f(1, 1) = 2d(1, 1)$$

$$f(e, 1) = f(e - 1, 1) + d(e, 1), \text{ (for } 2 \leq e \leq E)$$

$$f(1, o) = f(1, o - 1) + d(1, o), \text{ (for } 2 \leq o \leq O)$$

$$f(e, o) = \min \begin{cases} f(e - 1, o) + d(e, o), & \text{(right)(19)} \\ f(e - 1, o - 1) \\ \quad + 2d(e, o), & \text{(diagonal)(20)} \\ f(e, o - 1) + d(e, o), & \text{(up)} \end{cases} \quad (21)$$

The alignment path is extracted from the matrix generated by equation (19); the sequence of related points that equals  $f(E, O)$  in the matrix is the optimal

path. The optimal path is a sequence of  $\Phi = [(1, 1), \dots, (e, o), \dots (E, O)]$ , where the indexes  $e$  and  $o$  can be found at least once.

- (6) The relations depicted by the alignment path  $\Phi$  are used to calculate the next values. Equation (22) calculates the relation of the feature vector  $F$  of a new movement, and the template  $k$  of the library. The relation of the features is given by the optimal path obtained by the dynamic programming.

$$DC_k(\vec{F}) = \frac{1}{M} \sum_{j=1}^M \text{DistCluster}_j(f_i) \quad (22)$$

In equation (22), DistCluster calculated the distance of the feature  $f_i$  to the center of the cluster  $j$ . Equation (23) uses the mean and standard deviation of the cluster. All the distances are added and finally normalized by the number of clusters  $M$  in the template  $k$ .

$$\text{DistCluster}_j(f_i) = \exp\left(-\frac{|f_i - \mu_j|^2}{2\sigma_j^2}\right) \quad (23)$$

- (7) It is appropriate to calculate the percentage of activation of the clusters in the template as well as the activation of the features of the movement  $\vec{F}$ . The number of active clusters is computed by equation (24). A cluster is active when the value of DistCluster is greater than a selected threshold.

$$\text{ActPerCluster}_k = \frac{1}{M} \sum_{j=1}^M \text{ActCluster}_j(f_i) \quad (24)$$

$$\text{ActPerFeature}_k = \frac{1}{T} \sum_{j=1}^M \text{ActCluster}_j(f_i) \quad (25)$$

- (8) The values calculated above are used to decide whether the new movement is similar to those represented by the template  $k$  or not, i.e. subject to the following conditions:

$$\begin{aligned} DC_k &> \text{Threshold}_\delta \\ \text{ActPerCluster}_k &> \text{Threshold}_\epsilon \\ \text{ActPerFeature}_k &> \text{Threshold}_\rho \end{aligned} \quad (26)$$

If any relation of the movement with templates does not meet the constraints in equation (26), then it is valid to say that the movement  $\vec{F}$  is a novel movement. Once the movement  $\vec{F}$  has been classified as either one of the classes in the library or a novel movement, the next action is to update or create a new library's template with the values of the features of  $\vec{F}$ .

For each feature of  $\vec{F}$  a cluster is created and added. Because,  $f_i$  will be the only point in the cluster the mean values of the cluster will be equal to  $f_i$ . Nevertheless, the value of the standard deviation of the cluster is set to a fixed initial value  $\lambda$ . This value plays a crucial role for the cluster, because it determines the identification of similar features

and allows the update of the cluster and the features are related to the initial value.

One advantage of the use of the template representation for a movement is that each template has attached a unique representation of the class of movements that it represents. In Figure 5, the solid black line is the unique representation of a class of movements for the letter “a”. The grey lines are a set of movements classified for that class.

### IMITATION OF ACTIONS ON OBJECTS

In learning by imitation, the robot should be able to identify the goals of perceived actions, instead of only identifying the physical positions. Thus, once a goal has been identified, there are different physical positions that can lead to the same goal (Bekkering et al 2000). Hence, the level of imitation for this stage is reproducing the same goal, which means to focus on the essential effects of the observed action rather than just the particular physical motions (Billard et al 2004). An imitated action based only on physical movements might fail, when it is reproduced in altered environments or when, due to the different sizes and configuration of the bodies, the exact movement cannot be accomplished (Nehaniv and Dautenhahn 2002).

The mechanism of imitation described here, intends to obtain the state and the effects of the demonstrator in the environment. Hence, to achieve this, the mechanism uses information of two sources i.e. the observation of the performance of the demonstrator and a simulation of what the imitator believes the demonstrator is doing. The simulation of what the imitator believes the demonstrator is performing produces a change in the simulated agent state and also in the effects of the simulated environment. When the discrepancy, between the imitator’s simulation and what the imitator perceives, is marginal then the imitator is “confident” to know the state of the demonstrator and its effects on the environment. This mental simulation also provides a suggestion of the actions performed to reach that state and effects.

Although, the mechanism is able to extract the state and effects of the demonstrator’s performance, it is valid to say that the mechanism is able to identify the goals from the observed actions. The previous idea is true by assuming that a goal is a configuration of both the agent’s state and the agent’s effects on the environment (Nehaniv and Dautenhahn 2002). This configuration might be delineated by a sequence of actions.

#### Action representation

Since the reproduction of the goal is the main concern of the mechanism at this stage, it is crucial to define the goal and to specify the way that the robot could be able to identify such goals from observed actions. The purpose of this work is not to find out how to extract the goals from

actions. Therefore, we simplify the scenario with actions that the robot is able to identify its goals. In other words, each action has at least an embedded goal.

Despite the human body sharing many features with other types of objects, the body schema is treated differently from other object representations. Certain brain damages (i.e. Finger agnosia, autotopagnosia, hemispatial neglect, and ideomotor apraxia) suggest a separate representation of the body from other object representations (Reed 2002; Paillard 1999). In a study, the subjects tended to classify objects by their visual and their functionality, but only the body was classified first for its functionality (Reed et al 2004). The body schema is able to relate object representation through actions. Actions have a double presentation, an abstract one, used for planning and a motor one, used for the execution. The following details are only regarded to the abstract representation of an action used for planning.

An action can be defined as a *transition or affordance* from an *initial state* (namely *pre-conditions*) to produce a *final state* (namely *post-conditions*). According to the previous concept, an action consists of three elements, a transition, an initial state, and a final state. Both initial and final states are environment-agent state values or just *state values*, these values represent the condition of the environment, the agent or the relation environment-agent. These values can be enumerable, e.g. “State of the gripper” (Open or Close); or measurable, e.g. “distance end-effector object” (20 cm).

The pre-conditions are the state of the relation environment-agent before the transition can be executed. On the other hand, post-conditions are values that express the resulting environment-robot relation after the transition has taken place. Therefore, the post-conditions can be interpreted as intrinsic goals of an action (Nicolescu and Mataric 2001). An example of an action is, “to perform the action *grab* on the *object A*, the pre-conditions ask that the *object A* must be within a reachable distance, and that the gripper is not carrying another object. Once the action is executed, the post-conditions should be that the *object A* is within the gripper”.

#### Body schema and body percept

The body schema and the body percept have been expanded from the previous stage. Now, the body schema is able to use a library of actions that the robot can execute. Besides, the system is able to create new actions and store them in the library. The body percept stores abstract states, which record the relation environment-agent at a given instant, much like a snapshot. The robot’s percept sequence can be thought of as a history of what the robot has perceived during its lifespan (Russell and Norvig 2003).

The body percept can be interpreted in two different ways according to, whether the robot is observing to imitate or performing an action (see Table 1). When the robot is about to perform an action the body percept contains the relation of the environment-robot, and it is used as a pre-condition of the action about to perform. In contrast,

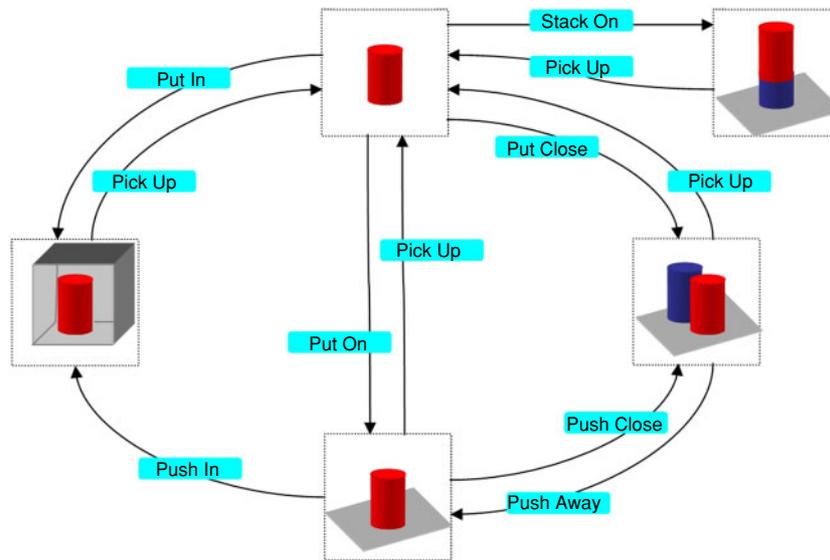


Figure 6 A FSM representation for the set of possible actions on the object “Cylinder”.

Table 1 The difference of the body schema and body percept for observing and performing an action

Feature	Observing	Performing
<i>Body percept</i>		
Type of relation	Environment–demonstrator	Environment–imitator
Percept (t)	Pre-condition (known)	Pre-condition (known)
Percept (t + 1)	Post-condition (known)	Post-condition (unknown)
<i>Body schema</i>		
Action	Unknown	Known
Function	Inverse function	Direct function

during a demonstration, the body percept is used as the goal to be achieved.

The robot–environment states values of the body percept are calculated in some cases by using the current reading from the sensorial, visual, and proprioception data. For example, the value for *Gripper state* is obtained directly from the sensory input, checking if the gripper is either closed or open. Nevertheless, in some cases we not only use the current sensorial readings at time  $t$ , but also those of previous states of the percept sequence  $[t-1, t-2, \dots, 0]$  as well. For example, in the case of the value of *Approaching*, the robot uses the previous states to check if the end-effector was far from an object, and now getting closer to that object.

The body schema, now, not only contains information about the physical parts of the body, but also keeps the knowledge of which actions can be performed. This knowledge about feasible actions is used to recognise those actions when they are executed. Therefore, the main two functions of the body schema regarding actions are stated as:

- Direct action, it is the action being taken when the robot has an action to perform and the pre-conditions in the current body percept ( $t$ ) are met. Then the goal of the action is achieved;

- Inverse action, this situation arises when the robot knows the goal that it has to achieve, which is contained in the body percept ( $t + 1$ ). Besides, the robot knows its current body percept ( $t$ ), which is a pool of possible preconditions. Thus, the robot can look an action up in its repertoire. That action would need to satisfy the goal and its preconditions must be met.

The robot’s body schema has been equipped with an initial set of actions known as “primitive actions”. The majority of these actions are atomic actions, which can be taken in one step. In contrast, a complex action is an action that contains at least two actions whether they are atomic or not. The robot is able to create new complex actions by employing the original set of primitive actions. The action in a complex action connected as a net by their pre-conditions and post-conditions. This net can be seen as a plan to achieve a particular task.

The primitive actions have been implemented as a finite state machine (FSM) (see Figure 6), in which the states are the pre-conditions and post-conditions, whereas the links between the states are the transitions. Therefore, possible actions over a particular object are gathered in a single FSM. One advantage of the FSM representation is that they tell us about the necessary sequence of actions from an initial state to a goal state.

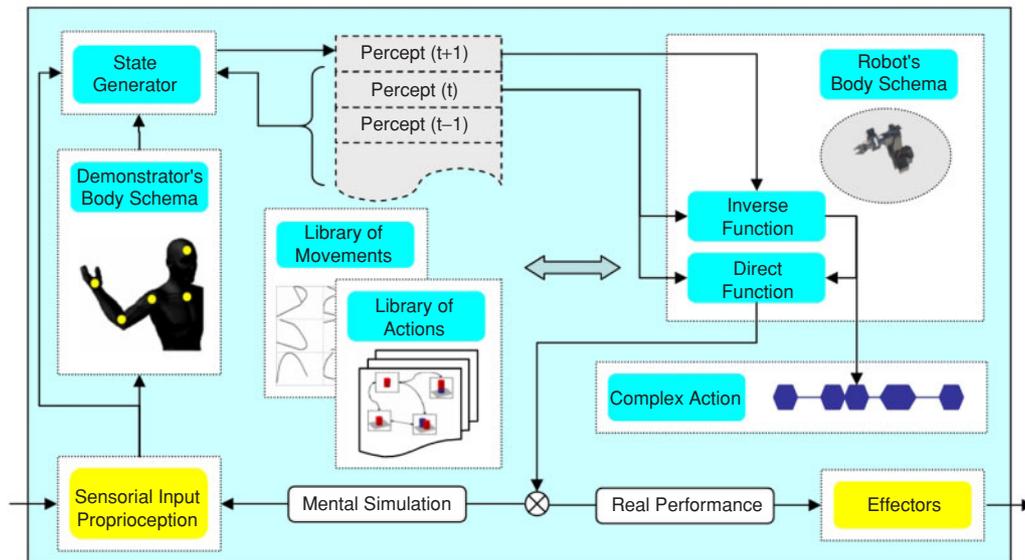


Figure 7 The interaction of the body percept and the body schema, when the robot system is learning. The output from an action can be inhibited to the effectors, which will produce a mental rehearsal of the action.

The FSM representation defines the *a priori* knowledge about a specific object. It only tells the imitator, which actions can be performed and which actions can not be performed on the object. Furthermore, the direct and inverse functions of the body schema exploit this representation.

#### Mechanism of imitation

The FSM representation of actions and the functions of the body schema only establish what the imitator can do with the object, but they do not tell the imitator how to achieve a task. In contrast, the mechanism of imitation allows the imitator to learn the necessary actions to accomplish a task. The aim of the mechanism is to achieve the same sequence of goals as the ones observed from the demonstrator. This sequence defines a way to solve a task.

This mechanism works in two different ways depending on whether the robot is observing to imitate or perform an action. The relation between the body schema and the body percept also changes. Table 1 shows the differences.

When the imitator is observing to imitate, the body percept will contain the environment–demonstrator state, which is to simulate what the demonstrator experiences. A change in the state of the demonstrator by an action is registered as a new body percept ( $t + 1$ ). Therefore, the robot will interpret this change as a new goal, i.e. the difference between its current body percept ( $t$ ) and the new body percept ( $t + 1$ ).

The body schema is now able to employ the inverse function to search for an action that covers the criteria of the new goal. The inverse function uses the body percept ( $t$ ) as pre-conditions and the body percept ( $t + 1$ ) as post-conditions. From the body percept ( $t$ ) and ( $t + 1$ ) is extracted the relation of possible relevant objects for the action. The relation of possible relevant objects leads the inverse function to search into specific FSM, where

the pre-conditions and post-conditions narrow the search to an action.

This selected action is passed to the direct function to predict the state after the execution of the action. These values of the predicted state are compared with the ones in the body percept ( $t + 1$ ); thus, one of the following situations may occur:

- If the predicted state is contained in the goal, then the action is added into the complex action;
- If the predicted state contains both the goal and extra values that are not part of the body percept ( $t + 1$ ); then these values are added to the body percept ( $t + 1$ ) and the action is added into the complex action;
- If the predicted state does not satisfy the goal, then the action is discarded and other relation of objects is used to find another action.

This learning process is depicted in Figure 7. The imitator can either, perform an action and send its signals to the effectors, or it can inhibit the physical motion thereby producing a mental simulation of the action.

Executing a learnt action involves a different relation of the body schema and the body percept. The body percept now contains environment–imitator state. Thus, the body percept ( $t$ ) is the current relation of environment and the imitator. The complex action to be executed is decomposed into its subaction. Each action  $A_i$  is executed following these steps:

- (1) It is checked whether the pre-conditions of action  $A_i$  are part of the current body percept ( $t$ ). If they are part of the body percept ( $t$ ) then step 2 is taken. But if the preconditions are not met then the inverse function is called to search for an action that once executed satisfies the pre-conditions of  $A_i$ . Therefore, the body percept ( $t$ ) becomes the pre-conditions for the inverse function, and

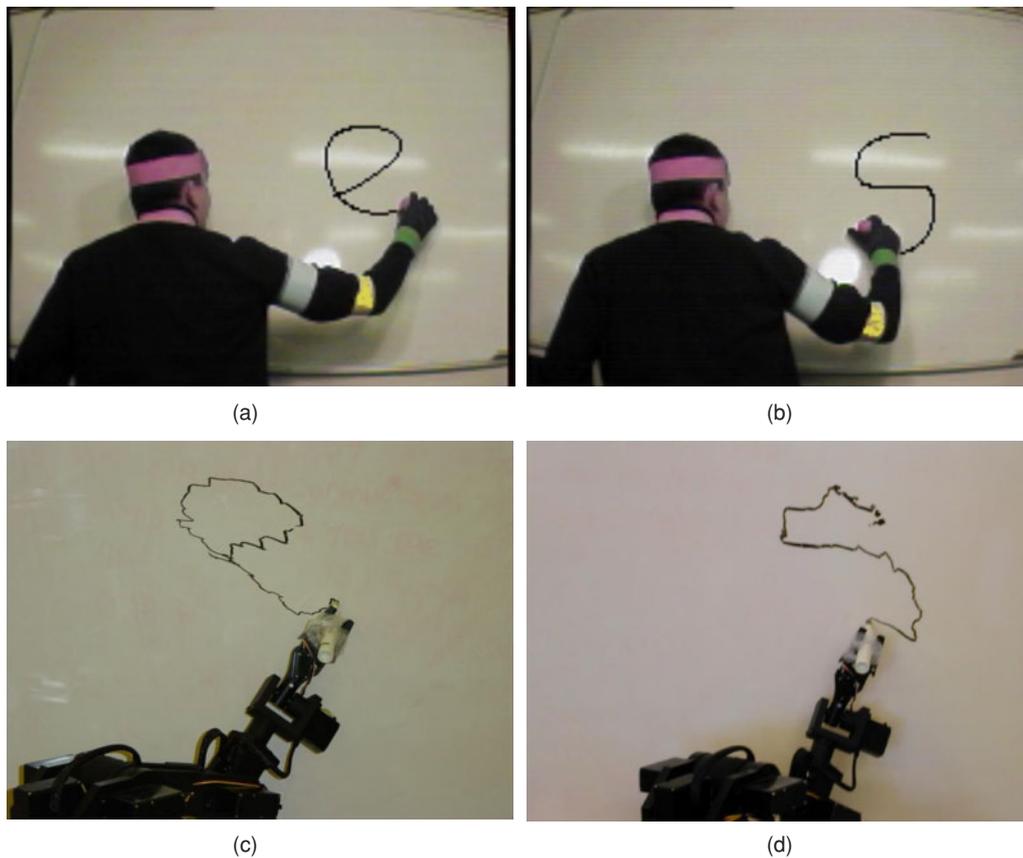


Figure 8 Learning phase, in (a) and (b) the demonstrator is writing the letters “e” and “s”. Execution phase, in (c) and (d) the robot is writing those letters.

the pre-conditions of  $A_i$  become the post-conditions of the inverse function. If an action is found by the inverse function, which satisfies the pre-conditions of the action to be executed, then this new action must be executed first, and it is sent to step 2. However, if no action is found then the system reports the error and stops the execution of the complex action;

- (2) The action  $A_i$  is executed by the direct function of the body schema, which produces the expected environment–robot state. Hence, the motor control receives both the motor commands for actuators and the expected environment–robot state. The sensors keep registering the values until either these values fulfill the expected environment–robot state or the watchdog-time is up. If the post-conditions of  $A_i$  are not reached, the system could try again the same motor commands or other that produce the same effect, when these exist. If these motor commands fail, the current environment–robot state is reflected into a new body percept ( $t + 1$ ) and the inverse function searches for an alternative action. This time the inverse function uses the new body percept ( $t + 1$ ) as pre-conditions and the post-conditions of  $A_i$  as post-conditions. If an action is found this must be executed and it is sent to step 2. When no action is found the system reports the error and stops;
- (3) When the action  $A_i$  is performed the environment–robot state values perceived in sensors are reflected into a new

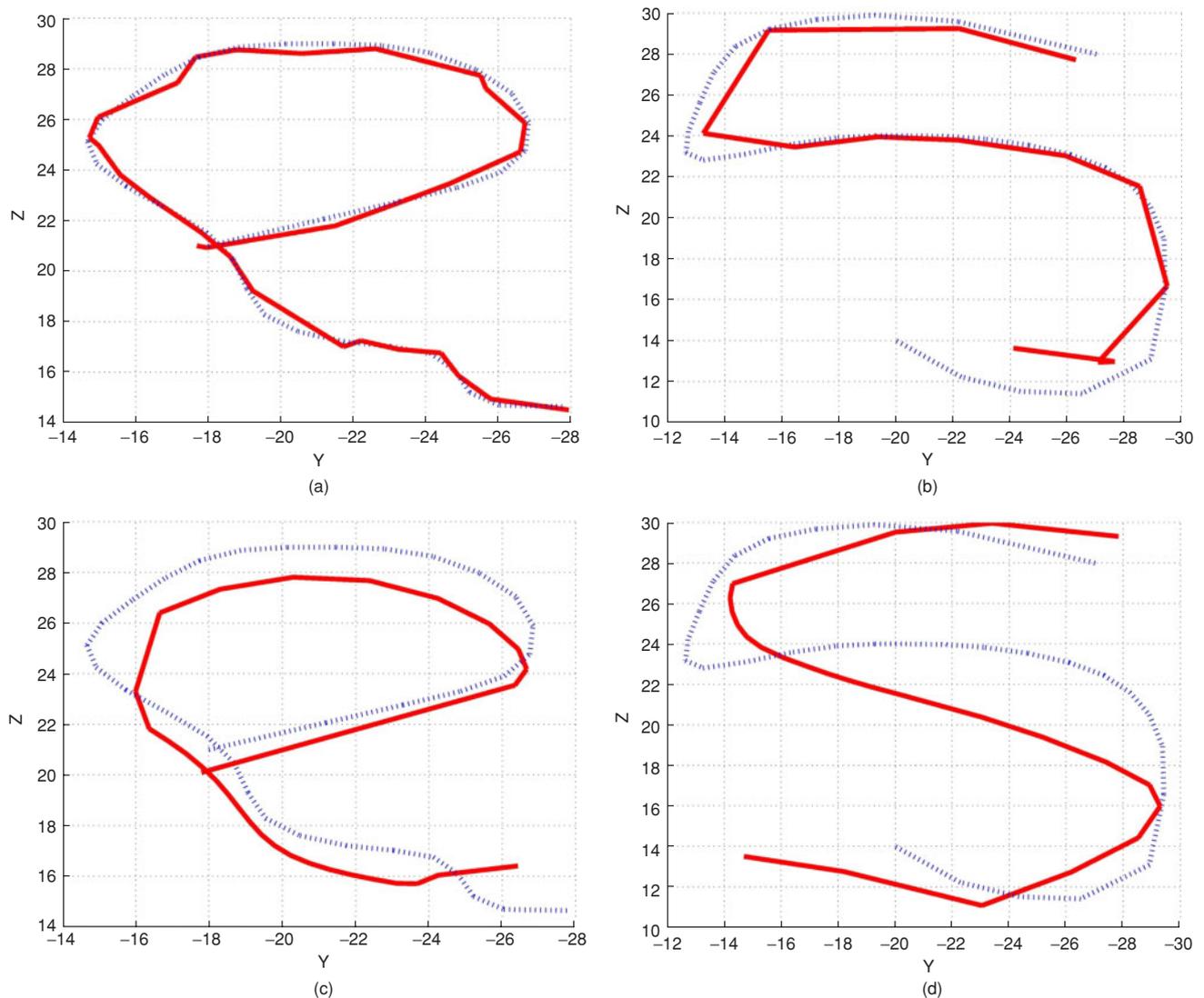
body percept ( $t + 1$ ). If there are more actions then the next action  $A_{i+1}$  is sent to step 1; until the complex action is finished.

The execution of a primitive action requires looking for the action into the library. On the other hand, to execute a complex action the direct function has to make a recursive call for each single action that compounds its net, until it finds the primitive actions. The direct function also monitors that the effects of the executed action are obtained, by checking them in the body percept ( $t + 1$ ).

## RESULTS AND DISCUSSION

In this section experimental results are presented to validate the abilities of the approach presented. Different sets of experiments were conducted for the following stages of imitation: imitation of body movements and imitation of actions on objects. The set-up for the experiment consists of the robot United4, as the imitator, and a human demonstrator. The experiments were conducted in two phases for all the cases:

- Learning phase, here the robot faced the demonstrator, while observing the movements/actions performed by the demonstrator; the aim of this phase was that the robot identified and learned those movements/actions;



*Figure 9* The performance of the robot in the solid line. In (a) and (c) the dotted line is the path of the demonstration from Figure 8a. While, (b) and (d) the dotted line is the path of the demonstration from Figure 8b. In (a) and (b) are the robot's performance by a mimicry process of what the robot observed. In contrast, (c) and (d) are the robot's performance by using the classifying method.

- Execution phase, here the robot was placed in the observed environment from where it learned; the aim of this phase was that the robot performed the movements/actions learned from the learning phase;

The experiments have been conducted in our Brooker laboratory. Relevant objects in the environment, including the joints of the demonstrator, were marked with different colors to simplify the feature extraction. In addition, the less clustered background permitted the robot to focus only on the color marked information.

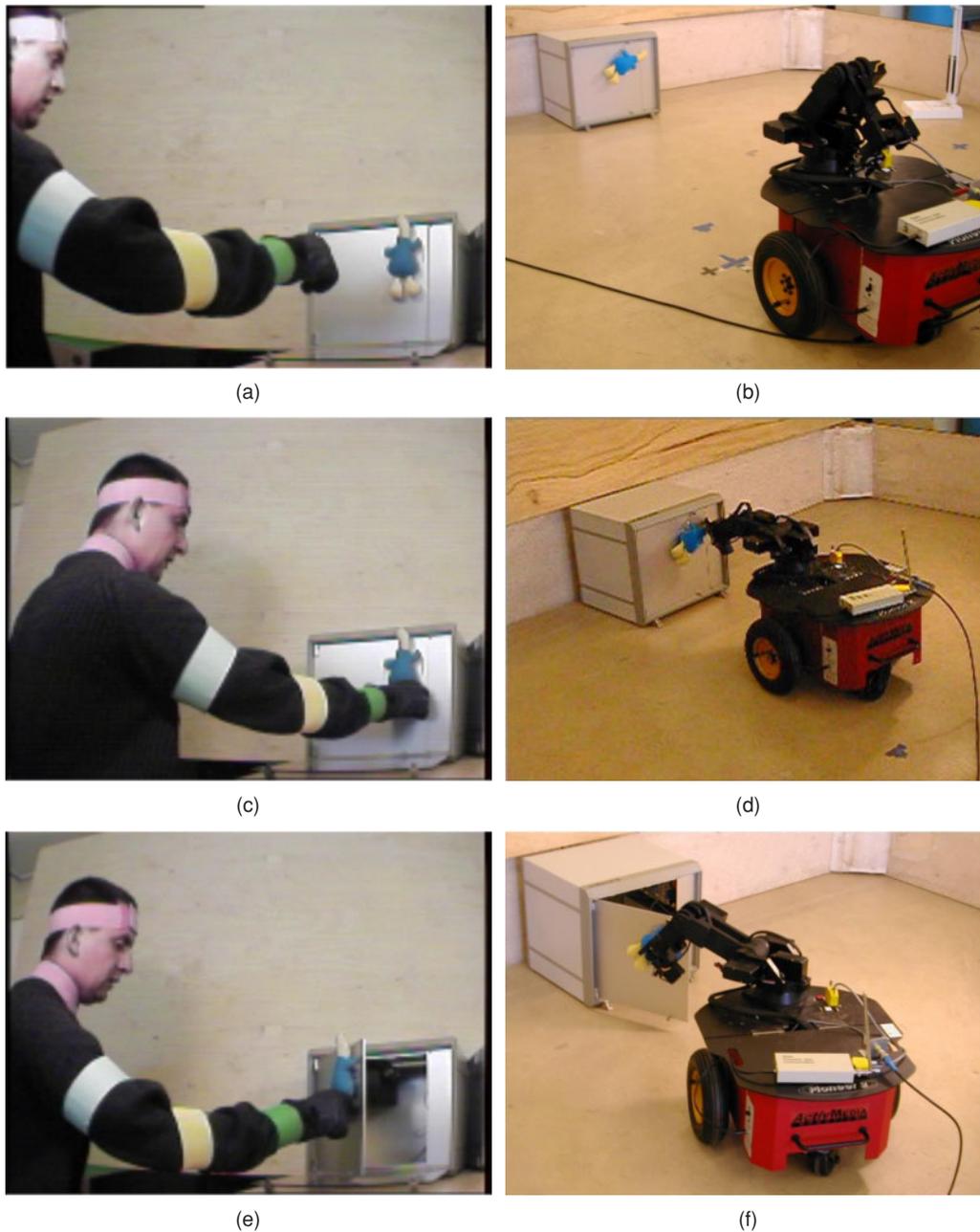
#### Experiments on imitation of body movements

The experiments for this stage of imitation consisted in the learning phase of the imitator observing the movements of the demonstrator's end-effector i.e. handwriting. The

execution phase consisted of the robot writing on a board, the learned letters.

The robot observed the demonstrator performing the handwriting whereas, by means of the colored markers that the demonstrator wears, the body representation of the demonstrator was extracted. This representation was related with the robot's representation by the body schema. Therefore, the robot could understand the new position of the demonstrator's end-effector within its workspace. The configuration needed to reach this desired position was eventually calculated by means of the kinematics methods. Finally, the path described by the end-effector was recorded and was ready to be executed.

Figure 8 presents the letters "e" and "s". The learning phase is presented in Figures 8a and 8b, where the demonstrator has written these letters. When the demonstrator



*Figure 10* Learning phase, (a) the demonstrator is approaching to the fridge, (c) the demonstrator is grabbing the handle of the fridge, and (e) opening the fridge's door. Execution phase, (b) the robot is looking for the fridge handle; in (d) the robot has approached to the object and grabs it, and (f) robot moving the fridge handle therefore, opening the door.

was describing the path of these letters, the robot was observing and relating those movements to its own. In the execution phase, Figures 8c and 8d, the robot is performing the paths described by the letters. Figures 9a and 9b show the learned path in a dotted line and the robot path in a solid line. The robot is writing the same letters by a mimicry process of the observed movement.

Figures 9c and 9d present the same observed movements from the demonstrator, but in this case the robot uses the method of classification and identification movements. Thus, the robot is able to identify the movements, which

represent the letters “e” and “s”. Besides, instead of using the same movement that the one observed, the robot uses the unique representation for each movement, which is defined by the cluster of each template.

The performance of the classification method is shown in Table 2, including five classes of letters, i.e. “A, B, C, D, and S”, and 155 sample data for each class 31 samples were provided. In the table below, the performance of classification for each class can be found. The column *Missed* presents the number of samples that were classified into a different class. The column *No Classified* shows the

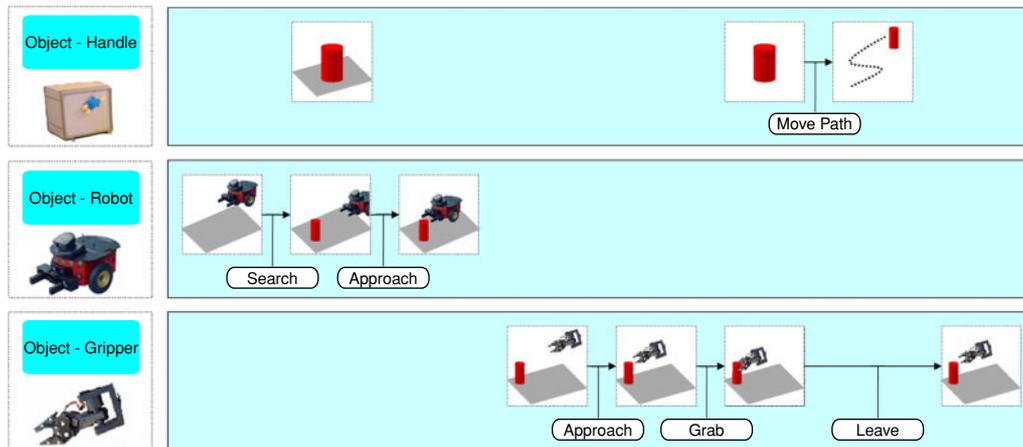


Figure 11 The complex action for the open door task. The initial state is the pre-conditions of the Search(Handle), and the final state is the post-conditions of Leave(Handle).

Table 2 The performance of the classification method for 5 different letters. The average performance of the system is 77.42%

Class	Values	Classified	Missed	No classified	Percentage
A	31	29	0	2	0.935484
B	31	25	2	4	0.806452
C	31	27	0	4	0.870968
D	31	14	9	8	0.451613
S	31	25	2	4	0.806452

number of samples for each class that were not classified into a particular class. Therefore, those samples could be treated as samples of a class that is not in the set of the five classes used for classification.

**Experiments on imitation of action on objects**

Regarding the experiments of imitation of action on objects, the robot was observing the demonstrator open a small fridge. Here, the robot was collecting not only data of the demonstrator but also information about the relevant objects in the environment.

For these experiments, the robot was endowed with a set of primitive actions. This set of primitive actions is mainly defined by the environmental states that the robot is able to recognize. The body percept is structured by both the position of the demonstrator’s end-effector and the states that represent the relation environment–robot. The state generator contains rules defining the behavior of the relation robot–objects in the environment. For example, for the action (*Carrying*) on an object, the object must be in the gripper, and moving at a constant motion with the gripper.

In Figure 10, the demonstrator shows the actions in (a), (c), and (e). The robot is performing those actions in (b), (d), and (f). First, it looks for the fridge’s handle, once this has been found (*Search*), the robot approaches to it (*Approach*). When the robot is close enough to reach the

object with the arm it stops, this behavior has been programmed in this way, since the robot is unable to estimate the distance from the camera image. The object is then centered in the image, so the arm can stretch (*Reach*) and grab it (*Grab*).

In all the previous actions there is only one parameter the object involved, the fridge’s handle. However, for the last action (*Carry*) we required two parameters: the object and the path to be followed by the arm. Thus, the arm is moving describing the path shown by the demonstrator as it is carrying the object, or to be more specific, pulling the object in this case.

Figure 11 presents the description of the actions, which compound the learnt complex action and their associated pre-conditions and post-conditions. The actions are taken from the finite state machine representation and add to the complex action, where they are related by their conditions.

**CONCLUSIONS AND FUTURE WORK**

Roboticists have begun to focus their attention on imitation, since the capability to obtain new abilities by observation represents many important advantages to the way of programming robots. Furthermore, learning by imitation presents considerable advantages in contrast with traditional learning approaches. It is believed that imitation, for its intrinsic social nature, might equip robots with new abilities through human–robot interaction. Therefore, the robots would eventually be ready to help humans in personal tasks.

We described our approach based on two main components that humans use when performing an action. These elements are believed to play a crucial role to achieve imitation: the body schema and the body percept. Developmental stages of imitation in humans were used to prove the key-role of these two components. The scope of this paper describes the progress on the following three stages:

- Body babbling.
- Imitation of body movements.
- Imitation of actions on objects.

For body babbling, i.e. at the first stage, we argue that a control method is a good way to endow the robot with a model to describe its body configuration (body schema). In this way we avoid a random trial-and-error learning process, as presented in infants. In the second stage, imitation of body movements, we defined the level of imitation selected was the reproduction of the path followed by the target. Here, the imitator focuses on the end-effector of the demonstrator, while the body schema finds the rest of body configuration satisfying the target position. In the last part of our work, imitation of actions on objects, the level of imitation pursued is to reproduce the same goal, which involves the identification of an action by using the body schema and using the body percept as the target goal.

The paper also presented experimental results supporting the feasibility of the proposed approach. Our future work comprises further experiments at the stage of imitation of action on objects, increasing the complexity of the tasks of the robot, as well as increasing the level of human-robot interaction.

## REFERENCES

- Acosta-Calderon C, Hu H. 2003. 'Robotic Societies: Elements of Learning by Imitation'. In *Proceeding of 21st IASTED International Conference on Applied Informatics, Innsbruck, Austria*, 315–320.
- Acosta-Calderon C, Hu H. 2003. 'Goals and Actions: Learning by Imitation', In *Proceeding of AISB03 Second International Symposium on Imitation in Animals and Artifacts*, Aberystwyth, Wales, 179–182.
- Acosta-Calderon C, Hu H. 2004. 'Imitation Towards Service Robotics'. In *Proceeding of International Conference on Intelligent Robots and Systems IROS 2004, Sendai, Japan*, 3726–3731.
- Acosta-Calderon C, Hu H. 2004. 'Robot imitation: A matter of body representation'. In *Proceeding of International Symposium on Robotics and Automation ISRA 2004, Queretaro, Mexico*, 137–144.
- Acosta-Calderon C, Hu H. 2005. 'Robot imitation from human body movements'. In *Proceeding AISB05 Third International Symposium on Imitation in Animals and Artifacts, Hatfield, UK*, 1–9.
- Alissandrakis A, Nehaniv CL, Dautenhahn K. 2004. 'Towards Robot Cultures? - Learning to Imitate in a Robotic Arm Test-bed with Dissimilar Embodied Agents'. *Int Studies: Soc Behav Commun Biol Artif Syst*, 5 (1):3–44.
- Bakker P, Kuniyoshi Y. 1996 'Robot see robot do: An overview of robot imitation'. In *Proceeding of AISB96 Workshop on Learning in Robots and Animals*, Brighton, England, 3–11.
- Becker M, Kefalea E, Mael E et al. 1999. 'GripSee: A Gesture-controlled Robot for Object Perception and Manipulation'. *Autom Robots*, 6:203–221.
- Bekkering H, Wohlschlagel A, Gattis M. 2000. 'Imitation of Gestures in Children is Goal Directed'. *Q J Exp Psychol*, 53:153–164.
- Billard A, Epars Y, Calinon S, et al. 2004. 'Discovering optimal imitation strategies'. *Robotics Autom Syst*, 47:69–77.
- Charminade T, Meltzoff A, Decety J. 2002. 'Does the End Justify the Means? A PET Exploration of the Mechanisms Involved in Human Imitation'. *Neuro Image*, 15:318–328.
- Dautenhahn K, Nehaniv C. 2002. 'The Agent-Based Perspective on Imitation' In Nehaniv CL, Dautenhahn K, *Imitation in Animals and Artefacts*. Cambridge, MA: The MIT Press, p.1–40.
- Demiris J, Hayes G. 2002. 'Imitation as a dual-route process featuring predictive and learning components: A biologically plausible computational model' In Nehaniv CL and Dautenhahn K, *Imitation in Animals and Artefacts*, Cambridge, MA: The MIT Press, p. 321–361.
- Demiris Y, Johnson M. 2003. 'Distributed, predictive perception of actions: a biologically inspired robotics architecture for imitation and learning'. *Connection Sci*, 15(4):231–243.
- Goldman A. 2001. 'Desire, Intention, and the Simulation Theory'. In Malle F, Moses L, Baldwin D, *Intention and Intentionality*, Cambridge, MA: MIT Press, p. 207–224.
- Ilg W, Bakir G, Franz M, et al. 2003. 'Hierarchical spatio-temporal morphable models for representation of complex movements for imitation learning'. In Nunes U., de Almeida A., Bejczy A., Kosuge K., and Machado J., In *Proceeding of The 11th International Conference on Advanced Robotics*, (2), 453–458.
- Klaffer R, Chmielewski T, Negin M. 1989. *Robotic Engineering and Integral Approach*, London, Prentice Hall.
- Lopes M, Bernardino J, Santos-Victor J. 2005. 'A Developmental Roadmap for Task Learning by Imitation in Humanoid Robots'. In *Proceeding of AISB05 Third International Symposium on Imitation in Animals and Artifacts, Hatfield, UK*.
- Mataric M, Pomplun M. 1998. 'Fixation Behavior in Observation and Imitation of Human Movement'. *Cogn Brain Res*, 7(2):191–202.
- Mataric M. 2002. 'Sensory-Motor Primitives as a Basis for Imitation: Linking Perception to Action and Biology to Robotics'. In Nehaniv CL, Dautenhahn K., *Imitation in Animals and Artefacts*, Cambridge, MA: The MIT Press, p. 391–422.
- Mckerrrow P. 1991. *Introduction to Robotics*, Reading, MA: Addison-Wesley.
- Meltzoff A, Moore M. 1992. 'Early Imitation Within a Functional Framework: The Important of Person Identity, Movement, and Development'. *Inf Behav Develop*, 15:479–505.
- Meltzoff A, Moore M. 1994. 'Imitation, Memory, and Representation of Persons'. *Inf Behav Develop*, 17:83–99.
- Nicolescu M, Mataric M. 2001. 'Experience-based Representation Construction: Learning from Human and Robot Teachers'. In *Proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hawaii, USA, 740–745.
- Nehaniv C, Dautenhahn K. 2001. 'Like Me? - Measures of Correspondence and Imitation'. *Cybern Syst*, 32 (1/2):11–51.
- Nehaniv C, Dautenhahn K. 2002. 'The Correspondence Problem'. In Nehaniv CL, Dautenhahn K., *Imitation in Animals and Artefacts*, Cambridge, MA: The MIT Press, p. 41–62.
- Paillard J. 1999. 'Body schema and body image. A double dissociation in defferented patient', in Gantchev GN, Mori S, Manssion J, *Motor Control, Today and Tomorrow*. Sofia, Bulgaria: Professor Marin DRINOV Academic Publishing House, p. 197–214.

- Rabiner L, Juang B. 1993. Fundamentals of Speech Recognition, Chapter 'Time Alignment and Normalization'. London, UK: Prentice Hall, p. 200–238.
- Rao R, Meltzoff A. 2003. 'Imitation Learning in Infants and Robots: Towards probabilistic Computational Models'. In Proceeding of AISB03 Second International Symposium on Imitation in Animals and Artifacts, Aberystwyth, Wales, 4–14.
- Reed C. 2002. 'What is the body schema?'. In Meltzoff AN, Prinz W, The Imitative mind, Cambridge: Cambridge University Press, p. 233–243.
- Reed C, McGoldrick J, Shackelford R, Fidopiastis C. 2004. 'Are human bodies represented differently from other animate and inanimate objects?'. *Vis Cogn*, 11:523–550.
- Russell S, Norvig P. 2003. Artificial Intelligence a Modern Approach, Second Edition, London, Prentice Hall.
- Schaal S, Peters J, Nakanishi J, Ijspeert A. 2003. 'Learning movement primitives'. In *Proceeding of International Symposium on Robotics Research (ISRR2003)*, Ciena, Italy.
- Viviani P, Stucchi N. 1992. 'Motor-perceptual interactions', *Tutorials in Motor Behavior*, Elsevier Science, New York, p. 229–248.
- Wolpert DM, Flanagan RJ. 2001. 'Motor Prediction'. *Current Biol*, 11 (18):R729–R732.
- Zollo L, Siciliano B, Laschi C, et al. 2003. 'An Experimental Study on Compliance Control for a Redundant Personal Robot Arm'. *Robot Autom Syst*, 44:101–129.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

