IEEE Transactions on Systems, Man and Cybernetics, PART B, Cybernetics,
Special issue on learning in Computer Vision and Patter Recognition.
Accepted for publication, 2004

Visual learning by imitation with motor representations

Manuel Cabido Lopes[†,‡] José Santos-Victor[†]

[†]Instituto de Sistemas e Robótica
Instituto Superior Técnico
Lisboa, Portugal

[‡]Escola Superior de Tecnologia
Instituto Politécnico de Setúbal
Setúbal, Portugal

http://vislab.isr.ist.utl.pt
{macl,jasv}@isr.ist.utl.pt

Abstract—

We propose a general architecture for action (mimicking) and program (gesture) level visual imitation. Action-level imitation involves two modules. The *View-Point Transformation*(VPT) performs a "rotation" to align the demonstrator's body to that of the learner. The *Visuo-Motor Map*(VMM) maps this visual information to motor data.

For program-level (gesture) imitation, there is an additional module that allows the system to recognize and generate its own interpretation of observed gestures, so as to produce similar gestures/goals at a later stage.

Besides the holistic approach to the problem, our approach differs from traditional work in (i) the use of motor information for gesture recognition; (ii) usage of context (e.g. object affordances) to focus the attention of the recognition system and reduce ambiguities and (iii) use iconic image representations for the hand, as opposed to fitting kinematic models to the video sequence.

This approach is motivated by the finding of visuomotor neurons in the F5 area of the macaque brain that suggest that gesture recognition/imitation is performed in motor terms (mirror) and rely on the use of object affordances (canonical) to handle ambiguous actions.

Our results show that this approach can outperform more conventional (e.g pure visual) methods.

I. Introduction

The impressive advance of research and development in robotics and autonomous systems in the past years has led to the development of robotic systems of increasing motor, perceptual and cognitive capabilities.

These achievements are opening the way for new application opportunities that will require these systems to interact with other robots or non technical users during extended periods of time. Traditional programming methodologies and robot interfaces will no longer suffice, as the system needs to learn to execute complex tasks and improve its performance throughout its lifetime.

Similarly to the ability of human infants to learn through (extensive) imitation, an artificial system can retrieve a large amount of knowledge, simply by looking at other individuals, humans or robots working in the same area.

The long-term goal of our work is two-fold. On one hand, we want to develop methodologies whereby a system can learn how to perform complex tasks through imitation. On the other hand, our approach relies on recent findings in neuroscience and developmental psychology, aiming to contribute to a better understanding of the fundamental problem of how humans imitate each other and how they recognize and understand the observed behavior and actions.

A. Learning by imitation

Learning by imitation has been addressed before in the fields of e.g. humanoid robots [1], where the number of degrees of freedom is very large, tele-operation [2] or assembly tasks [3]. However, most published works only focus on specific components of an imitation system. Instead, we take an holistic approach to describe a complete architecture for arm-hand gesture imitation and recognition, following biologically plausible methodologies. In the work described in [4], the imitator can replicate both the demonstrator's gestures and dynamics. Nevertheless, it requires the usage of an exoskeleton to sense the demonstrator's behavior. Instead, our approach is exclusively based on vision.

We will distinguish two forms of imitation: action-level and program-level imitation. Action-level imitation (or mimicking) consists in replicating the gestures or movements of a demonstrator, without seeking to understand those gestures or the action's goal. Instead, program-level imitation (or gesture imitation) involves recognizing the performed gesture/goal so that the learner can produce its own interpretation of the gesture or action effect. Our overall approach to the problem of learning through (action-level or gesture) imitation is illustrated in Fig. 1 and considers a system composed by an anthropomorphic arm-hand and monocular vision.

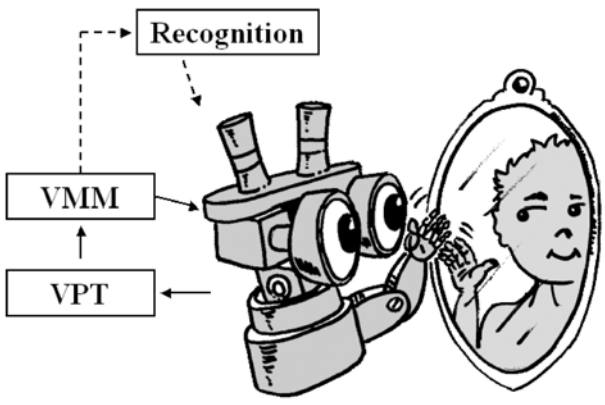

Fig. 1. The combination of the Sensory-Motor Map and the View-Point Transformation allows the robot to mimic the arm movements executed by another robot or human. The recognition module endows the robot to interpret and imitate gestures or goal-directed actions.

It contains three main modules. The *View-Point Transformation* (VPT) maps observed gestures to a canonical point-of-view, that corresponds to the visual appearance of a gesture, as if it were performed by the system itself. The *Visuo-Motor Map* (VMM) maps these visual features to motor data directly. The execution of these motor commands produces a gesture that mimics the one performed by the demonstrator, i.e. action-level imitation. The final recognition module endows the system with the ability to recognize goal-directed actions (gestures) executed by a demonstrator. Since the recognition is done in motor terms, the system can then reproduce its own interpretation of the observed gestures: program-level imitation.

B. View-Point Transformation

For action-level imitation, the learner has not only to visually detect the demonstrators arm (or hand) but also to conceive a "mental rotation" that will place the demonstrator's arm (*allo-image*) in correspondence with the learner's own body (ego-image). This spatial transformation is named the *View-Point Transformation* (*VPT*), illustrated in Figure 2. The VPT is needed because the image appearance of an object may change quite dramatically, as a function of the view-point. If we could find image descriptors invariant to view-point changes, then the VPT would no longer be needed.

The VPT maps the gestures of a demonstrator to the (*ego-*)image, that would be obtained if those same gestures were performed by the system itself. Surprisingly, in spite of the importance given to the VPT in psychology [5], it has received very little attention from other researchers in the field of visual imitation.

Fig. 2. Similar gestures can be seen from very distinct perspectives. The image shows one's own arm performing a gesture (ego-image) and that of the demonstrator performing a similar gesture (allo-image).

One of the few works that deals explicitly with the VPT is described in [6]. However, instead of considering the complete arm posture, only the mapping of the end-effector position is done. The VPT is performed using epipolar geometry, based on a stereo camera pair.

Other studies address this problem only in an implicit or more superficial way. A mobile robot, capable of learning the policy followed by another mobile vehicle, is described in [7]. Since the system kinematics is very simple, the VPT corresponds to a transformation between views of the two mobile robots. In practice, this is achieved by delaying the imitator's perception, until it reaches the same place as the demonstrator, without explicitly addressing the process of VPT. The work described in [8] has similar objectives to our own research, and allows a robot to mimic the "dance" of an Avatar. However, it does not address the VPT at all, and a special invasive hardware is used to perform this transformation.

If a system is able to handle the view-point correspondence, action-level imitation requires mapping the visual features expressed in the ego-image to the corresponding motor commands, through the *Visuo-Motor Map*.

C. Visuo-Motor Map (VMM)

The VMM can be computed explicitly if the parameters of the arm-hand-eye configuration are known a priori but– more interestingly–it can be learned from observations of arm/hand motions.

Again, biology can provide relevant analogies. The Asymmetric Tonic Neck reflex [9] forces newborns to look at their hands, allowing them to learn the relationship between motor actions and the corresponding visual stimuli. Similarly, in our work the robot learns the VMM during an initial period of self-observation, while performing hand/arm movements, as both visual and motor (proprioceptive) data are available.

Once the VMM has been estimated, the robot can observe a demonstrator, use the VPT to transform the image features to a canonical reference frame and map these features to motor commands through the VMM. The final result will be a posture similar to that observed.

The VMM can be learnt sequentially, thus decreasing the complexity at each step. The system can start to map the shoulder/elbow joints of the arm. Once this is done, it simplifies the learning of the VMM for the wrist configuration. Finally, once the arm VMM is available, it can provide information for learning the hand VMM. In this paper, all these VMMs are estimated using neural networks.

This sequential strategy for learning the different components of the VMM resembles human development stages [10], [11]. During the first months of life, infants have limited visual and motor capabilities. Both systems evolve side by side, with the visual system feeding information to "calibrate" hand/arm movements and arm movements providing stimuli to train and improve visual acuity [12]. Similarly, in our work, the sensory-motor map is learned in a sequential (developmental) process.

The VPT and VMM allow the system to perform action-level imitation. For program-level imitation, we need to provide the means for recognizing the gestures performed by someone, or their produced effect (goal). Then, an equivalent (but not necessarily equal) gesture can be elicited by the learner. One example could be producing the same effect on a certain object, even if the gesture is (from a kinematic point of view) different.

D. Program-level (gesture) imitation

Our work on program-level (gesture) imitation is strongly motivated by the recent discovery of visuomotor (*mirror* and *canonical*) neurons [13], [14] in the F5 area of the macaque's brain. These neurons discharge during the execution of hand/mouth movements. In this paper we will focus on arm-hand gestures, often referred to as grasp actions or grasps.

In spite of their localization in a pre-motor area of the brain, *mirror* neurons fire both when the animal performs a specific goal-oriented grasping task, and when it sees that same action being performed by another individual. This observation suggests that the motor system responsible for triggering an action is also involved when recognizing that same action. In other words, recognition would be performed in motor terms, rather than in a purely visual space. By establishing a direct connection between gestures performed by a subject and similar gestures performed by others, mirror neurons may be connected to the ability to imitate found in some species [14], establishing an implicit level of communication between individuals.

Canonical neurons [15] have the intriguing characteristic of responding when objects, that afford a *specific* type of grasp, are present in the scene, even if the grasp action is not performed or observed. Thus, canonical neurons may encode object affordances and help distinguishing ambiguous gestures during the process of recognition.

Many objects are grasped in very precise ways, since they allow the object to be used for some specific purpose. A pen is usually grasped in a way that affords writing and a glass is hold in such a way that we can use it to drink. Hence, if we recognize an object that is being manipulated, it immediately tells us some information about the most likely grasping possibilities (expectations) and hand motor programs, simplifying the task of gesture recognition.

Our work has three main distinctive aspects when compared to traditional approaches: (i) the use of motor information for gesture recognition; (ii) the use of context (e.g. object affordances) to focus the attention of the recognition system and reduce ambiguities and (iii) the use of iconic image representations for the hand, as opposed to fitting kinematic models to the video sequence.

In contrast with the approaches that perform gesture recognition in pure visual terms, we rely on motor information for (program-level) gesture imitation or recognition. We show that this approach leads to considerable simplification of the problem since the motor representations offer a (much) larger degree of invariance to view-point modifications. The work described in [16] is closely related to ours and proposes a model for *mirror neurons*. However, the visual features they use are very difficult to extract from a video sequence, which makes the approach unreliable.

Another important aspect in our work is the use of context information or object affordances [17] for recognition. If an object is more likely to be grasped in some specific way, then the observation of this object in the scene introduces prior knowledge that will bias the gesture classification process. Similarly, certain arm-gestures can be more likely in certain contexts than others. This methodology is in accordance with the observation of *canonical neurons* discharges, when a graspable object is present in the scene. We blend prior information and observations in a Bayesian framework that achieves high classification rates.

For program-level (gesture) imitation, we will concentrate on grasping actions. Grasp actions are usually partitioned into the *transport* and *grasp* phases [18]. Experiments in neurophysiology indicate that only the grasp phase is relevant for the process of gesture recognition. Figure 3 illustrates the hand appearance during the approach phase, together with the final phase of two broad classes of grasps that we used: precision grip and power grasp.

Fig. 3. Hand appearance during the approach phase (left), power grasp (center) and precision grip (right).

Gesture recognition has been addressed in the computer vision community in many different ways [19]- [24]. The difficulty of hand tracking/recognition arises from the fact that the hand is a deformable, articulated object, that may display many different appearances depending on its configuration, view-point or illumination. In addition, there are frequent occlusions between hand parts (e.g. fingers). Due to the extreme difficulty in extracting/tracking finger-tips or other notable points in the image, under varying viewpoints, we exploit more iconic, appearance based, representations for the hand shape, that are commonly believed to be used by humans when recognizing (known) gestures.

E. The role of observation in learning

A final aspect worth mentioning is the role of observation for the overall system we propose here, both self-observation or looking at other individuals. Observation is involved in different types of learning objectives:
(i) By manipulating objects, one can learn which grasp types are successful for a certain class of objects. Also, if we observe *other* people manipulating objects, we can learn the most likely grasps or functions, for a given class of objects, the *affordances* [17] associated to a certain object.
(ii) When observing one's own gestures, the hand appearance can be estimated and directly related with the corresponding motor commands. Hence, proprioceptive (motor) and visual information can be used to determine the *Visuo-Motor Map* in a natural way.

F. Structure of the paper

The structure of the paper is as follows. In Section II, we present the models used throughout this work, namely the arm/hand kinematics, the hand appearance model and the camera/eye geometry. Section III is devoted to the definition and estimation of the *Visuo-Motor Map* and how to learn this map from observations. In Section IV, we

describe how the system performs the *View-Point Transformation*. Section V describes our Bayesian framework for program-level (gesture) imitation/recognition. Recognition is done in the motor space (mirror neurons) and relies on prior knowledge provided by context or object affordances (canonical neurons). In Section VI we present results obtained using our approach, both for action and program-level imitation. In Section VII, we draw some conclusions and establish directions for future work.

II. Modeling

Our robotic system consisting of an anthropomorphic arm/hand, equipped with a single camera. While the robot arm/hand is simulated, we use a real camera and performed extensive experiments with real data. This section presents the models used for the camera and robot body.

A. Body/arm kinematics

The anthropomorphic arm is modeled as an articulated link system. Fig. 4 shows the four arm links: L_1 - forearm, L_2 - upper arm, L_3 - shoulder width and L_4 - body height.

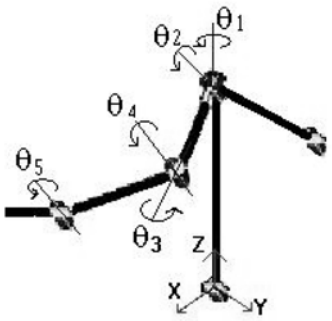

Fig. 4. Kinematic model of the human arm.

It is further assumed that the relative sizes of these links are known, e.g. from biometric measurements: $L_1 = L_2 = 1$, $L_3 = 1.25$ and $L_4 = 2.5$.

B. Camera/eye geometry

An image is a $2D$ projection of the $3D$ world whereby depth information is lost. In our case, we will retrieve depth information from a single image using knowledge about the body links and a simplified, orthographic camera model.

We use the scaled orthographic projection model that assumes that the image is obtained by projecting all points along parallel lines plus a scale factor. Interestingly, such approximation may have some biological grounding taking into account the scale-compensation effect in human vision [25] whereby we normalize the sizes of known objects irrespective to their distances to the eye.

Let $\mathbf{M} = [\mathbf{X}\ \mathbf{Y}\ \mathbf{Z}]^{\mathbf{T}}$ denote a 3D point expressed in the camera coordinate frame. Then, with an orthographic camera, $\mathbf{M}$ is projected onto $\mathbf{m} = [\mathbf{u}\ \mathbf{v}]^{\mathbf{T}}$, according to:

$$\mathbf{m} = \mathcal{P}\mathbf{M}$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = s \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (1)$$

where s is a scale factor that can be estimated placing a segment with size L fronto-parallel to the camera and measuring the image size $l (s = l/L)$.

For simplification, we assume that the camera axis is positioned in the imitator's right shoulder with the optical axis pointing forward horizontally. With this specification of the camera pose, there is no need for an additional arm-eye coordinate transformation in Equation (1).

C. Hand

Figure 5 shows the large variance of an human hand's appearance, observed under a variety of view-points. It consists of a complex, multi-link system, prone to generate numerous self-occlusions, which hardens feature extraction or model-based tracking. To avoid this problem and to enhance robustness, we rely on global features, obtained by projecting the hand images onto a lower dimension subspace, using Principal Component Analysis (PCA).

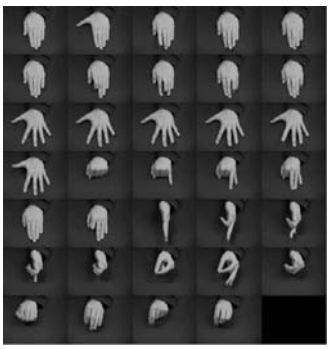

Fig. 5. The hand appearance changes dramatically as a function of its configuration and view-point.

While the system performs arm/hand movements, self-observation is a powerful means to gather a vast set of visual stimuli, corresponding to many distinct hand postures, view-points and appearances. The hand is segmented using colour information and size/orientation are normalized. These images form the database for the PCA. The number of components used for recognition purposes is 5 and 15 for the visuo-motor transformations.

It is not possible to use the same idea to create a set of images showing the arm in a sufficiently large variety of configurations and view-points. Therefore, we have to model the arm geometrically and explicitly handle view-point changes.

In the motor space, the hand configuration can be expressed by its several degrees of freedom. Figure 6 shows the kinematic model used. This simplified structure has 15 degrees of freedom, 3 for each finger. Finger abduction could be added to increase the model quality.

To perform experiments with motor and visual information, a data-glove system [26], capable of recording 22 values of the hand configuration, was used. It consists of a glove with strain gauges to measure joint angles. In a real

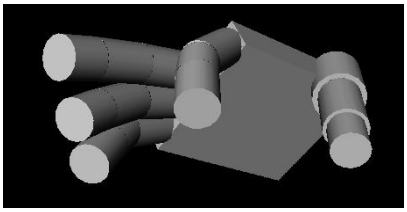

Fig. 6. Kinematic model of an human hand

(robotic or living) system, motor features would correspond to the hand/arm pose/motion proprioceptive information.

III. Visuo-Motor Map

The *Visuo-Motor Map* (VMM) defines a correspondence between perception and action. In our approach, the VMM is structured in two different ways, depending on whether the arm or the hand are being considered. Since the arm has a relatively simple kinematic model, we assume that the arm joints can be tracked and the VMM will thus relate the (ego-)image coordinates of those joints to the actual joint angles. Instead, the hand has a rather complex kinematic structure and tracking fingertips can be quite difficult. To overcome this problem, we rely on appearance based methods to visually represent the hand, in its various possible configurations. The VMM can be interpreted in terms of forward/inverse kinematics for the case of robot-eye system:

$$VMM : \mathbf{F}^V \longleftrightarrow \mathbf{F}^M$$

where F^V and F^M denote some visual (motor) features.

The VMM can be used to predict the image resulting from moving one's arm (or hand) to a certain posture ($\mathbf{F}^M$ to $\mathbf{F}^V$), or to infer the motor command used to achieve the observed posture ($\mathbf{F}^V$ to $\mathbf{F}^M$). This last capability will be used to make recognition in motor space and to imitate.

A. The arm Visuo-Motor Map

In the context of imitation, the VMM can be used with different levels of ambiguity/completeness. In some cases, one wants to replicate exactly someone else's arm gestures, considering all the joint angles. In some other cases, however, we may want to imitate arm poses only, while the position of the elbow or the rest of the arm configuration is irrelevant. For the arm case, and to encompass these possibilities, we have considered two cases: the *full arm VMM* and the *free-elbow VMM*.

A.1 Full-Arm VMM

We denote the elbow and wrist image coordinates by $\mathbf{m_e}$ and $\mathbf{m_w}$, the forearm and upper arm image lengths by l_1 and l_2 and the various joint angles by $\theta_{i=1..4}$. We have:

$$[\theta_1, \cdots, \theta_4] = \mathcal{F}_1(\mathbf{m_e}, \mathbf{m_w}, \mathbf{l_1}, \mathbf{l_2}, \mathbf{L_1}, \mathbf{L_2}, \mathbf{s})$$

where $\mathcal{F}_1(.)$ is the VMM, L_2/L_1 represents the (known) length of the upper/forearm and s is the camera scale factor.

The computation of this function can be done in successive steps, where the angles of the shoulder joint are determined first and used in a later stage to simplify the calculation of the elbow joint's angles.

The inputs to the VMM consist of features extracted from the image points of the shoulder, elbow and wrist; the outputs are the angular positions of every joint. The shoulder pan and elevation angles, θ_1 and θ_2 can be readily obtained from image data as:

$$\begin{aligned} \theta_1 &= f_1(\mathbf{m_e}) = \arctan(\mathbf{v_e}/\mathbf{u_e}) \\ \theta_2 &= f_2(l_2, L_2, s) = \arccos(l_2/sL_2) \end{aligned}$$

After extracting the shoulder angles, the process is repeated for the elbow. Before computing this second set of joint angles, the image features undergo a set of transformations so as to compensate the rotation of the shoulder:

$$\begin{bmatrix} u'_w \\ v'_w \\ \xi \end{bmatrix} = \mathcal{R}_{zy}(\theta_1, \theta_2) \left(\begin{bmatrix} u_w \\ v_w \\ \sqrt{s^2 L_1^2 - l_1^2} \end{bmatrix} - \begin{bmatrix} u_e \\ v_e \\ 0 \end{bmatrix} \right) \quad (2)$$

where ξ is not used in the remaining computations and $\mathcal{R}_{zy}(\theta_1, \theta_2)$ denotes a rotation of θ_1 around the z axis followed by a rotation of θ_2 around the y axis.

With the transformed coordinates of the wrist we can finally extract the remaining joint angles, θ_3 and θ_4:

$$\begin{aligned} \theta_3 &= f_3(\mathbf{m'_w}) = \arctan(\mathbf{v'_w}/\mathbf{u'_w}) \\ \theta_4 &= f_4(\mathbf{m'_w}, \mathbf{L1}, \mathbf{s}) = \arccos(\mathbf{l'_1}/\mathbf{sL_1}) \end{aligned}$$

The approach just described allows the system to determine the joint angles corresponding to a certain image configuration of the arm. In the next section, we will address the case where the elbow joint is allowed to vary freely.

A.2 Free-Elbow VMM

The *free-elbow* VMM is used to generate a given hand position, while the elbow is left free to reach different configurations. The input features consist of the hand image coordinates and the shoulder-hand distance.

$$[\theta_1, \theta_2, \theta_4] = \mathcal{F}_2(\mathbf{m_w}, {}^{\mathbf{r}}\mathbf{dZ_w}, \mathbf{L_1}, \mathbf{L_2}, \mathbf{s})$$

The elbow joint, θ_3, is set to a comfortable position. This is done in an iterative process aiming at maintaining the joint positions as far as possible from their limit values. The optimal elbow angle position, $\hat{\theta}_3$ is chosen to maximize:

$$\hat{\theta}_3 = \arg\max_{\theta_3} \sum_i (\theta_i - \theta_i^{limits})^2$$

while the other angles can be calculated from the arm features. Again, the estimation process can be done sequentially, each joint being used to estimate the next one:

$$\begin{aligned} \theta_4 &= \arcsin\left(\frac{{}^r x_h^2 + {}^r y_h^2 + {}^r z_h^2}{2} - 1 \right) \\ \theta_2 &= 2\arctan\left(\frac{b_1 - \sqrt{b_1^2 + a_1^2 - c_1^2}}{a_1 + c_1} \right) + \pi \\ \theta_1 &= 2\arctan\left(\frac{b_2 - \sqrt{b_2^2 + a_2^2 - c_2^2}}{a_2 + c_2} \right) \end{aligned}$$

where the following constants have been used:

$$\begin{aligned} a_1 &= \sin\theta_4 + 1 \\ b_1 &= \cos\theta_3 \cos\theta_4 \\ c_1 &= -{}^r y_h \\ a_2 &= \cos\theta_4 \cos\theta_2 \cos\theta_3 - \sin\theta_2(1 + \sin\theta_4) \\ b_2 &= -\cos\theta_4 \sin\theta_3 \\ c_2 &= {}^r x_h \end{aligned}$$

A.3 Learning the Arm VMM

In the previous sections we have derived the expressions of the full-arm and free-elbow VMMs. We could thus use these expressions directly to predict the visual consequences of some arm motion. Instead, we adopted a learning approach whereby the system learns the VMM by performing arm movements and observing the effect on the image plane. In this way, the system will not depend explicitly on the knowledge of some design parameters and can adapt automatically to any changes or deviations from such theoretical model.

From the derivation of the analytical expressions, we see that the VMM can be computed sequentially: estimating the first angle, which is then used in the computation of the following angle and so forth. This fact allows the system to learn the VMM as a sequence of smaller learning problems.

This approach strongly resembles the development of sensory-motor coordination in newborns and young infants, which starts by simple motions that get more and more elaborate, as infants acquire a better control over motor coordination.

In all cases, we use a *Multi-Layer Perceptron* (MLP) to learn the VMM, i.e. to approximate functions $f_{i=1..4}$. Table I presents the learning error and illustrates the good performance of our approach for estimating the VMM. The value 3.6 corresponds to the threshold for the training algorithm. The order of magnitude is $100\times$ bigger in the last 2 degrees of freedom because they depend on the previous ones in a non-linear way.

θ_1	θ_2	θ_3	θ_4
$3.6e^{-2}$	$3.6e^{-2}$	3.6	3.6

TABLE I
MEAN SQUARED ERROR (IN DEG.^{2}) FOR THE EACH JOINT IN THE *full-arm VMM*

Ideas about development can be further exploited in this construction. Starting from simpler cases, de-coupling several degrees of freedom, interleaving perception with action learning cycles are developmental "techniques" found in biological systems.

B. Hand Visuo-Motor Map

During self-observation, the system can generate a large variety of hand visual stimuli, for the construction of the VMM. The learning consists in estimating a subspace, spanning hand images taken from a variety of view-points. The hand VMM relates the hand image (normalized for orientation and scale) directly to the finger joint angles.

As the transformation from the visual space to the motor space is quite complex, it was learned with a Multi-Layer Perceptron, for each joint angle. For each network, i, the input consists of a 15-dimensional vector $\mathbf{F}^V$, which are the PCA components of the imaged hand appearance. The output consists of a single unit, coding the corresponding joint angle, $\mathbf{F}_i^M$. There are 5 neurons in the hidden layer.

We assume that $\mathbf{F}^V$ is captured across many different view-points. This is possible to generate during self-observation since a huge variety of hand configurations can be easily displayed. Otherwise, a view-point transformation is needed to pre-align the visual data [27].

The VMM can lead to impossible (temporal) trajectories, as errors in input frames can cause discontinuities in the motor space. To overcome this problem, continuity is imposed in the motor data through a first-order dynamic filter.

Each neural network was trained with momentum and adaptive *back-propagation* with the data pre-processed to have zero mean and unitary variance. It converges to an error of 0.01 in less than 1000 epochs.

Figure 7 shows trajectories (solid-line) for a joint angle of the little finger when performing several precision grips. It is noticeable that, even inside each grasp class, the variability is very large. This is due to the differences between the grasped objects, and illustrates how the observed features depend not only on the "grasp" type but also on the manipulated object (see Section V-B for discussion). The dashed-line in the figure shows that the trajectory reconstructed through the neural-VMM is remarkably close to the "true" values. The accuracy of the VMM may degrade when more complex gestures are included, but then the type objects in the scene or overall context will play a more important role.

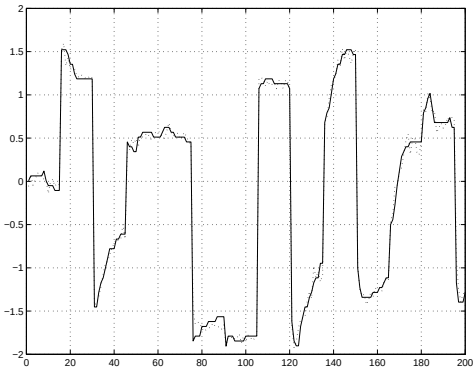

Fig. 7. Several trials of precision grip experiment. Solid line: original motor information. Dotted Line: reconstructed motor information using the Visual-Motor Map (VMM)

A final aspect worth mentioning is that the hand-VMM can also be learned during an initial phase, when the system (natural or artificial) performs hand gestures and observes the (visual) consequences of such gestures. Both proprioceptive (motor) and visual data are present and the

association can be established. An additional comment is that self-observation may allow the system to search and tune the most interesting visuo-motor features, such that a more compact representation could be used.

IV. View-Point Transformation

A certain arm gesture can be seen from very different perspectives, depending on whether the gesture is performed by the robot (self-observation) or by the demonstrator.

One can thus consider two distinct images: the *ego-centric* image, I_e, during self-observation and the *allo-centric* image, I_a, when looking at other robots/people. The *View-Point Transformation* (VPT) has to align the allo-centric image of the demonstrator's arm, with the ego-centric image, as if the system were observing its own arm.

In our work, we model the arm as a kinematic chain, whose image projection greatly depends on the observation view-point. For that reason, we explicitly develop a procedure to determine the VPT [27] for the arm.

Instead, the hand is treated in a different manner. To avoid the difficulty of fitting a kinematic model of the hand to the images, we chose to use the hands image appearance, directly as representations. Since different view-points and the resultant appearance changes are already taken into account, there is no need to explicitly define a VPT for the hand. Hence, from this point on, we only consider the VPT for the arm configuration.

The precise structure of the VPT is related to the ultimate meaning of imitation. Experiments in psychology show that imitation tasks can be ambiguous. In some cases, humans imitate only partially the gestures of a demonstrator (e.g. replicating the hand pose but having a different arm configuration, as in sign language), use a different arm or execute gestures with distinct absolute orientations [28]. In some other cases, the goal consists in mimicking someone else's gestures as completely as possible, as when performing dancing or dismounting a complex mechanical part.

According to the structure of the chosen VPT, a class of imitation behaviors can be generated. We consider two different cases. In the first case - 3D VPT - a complete three-dimensional imitation is intended. In the second case - 2D VPT - the goal consists in achieving coherence only in the image, even if the arm pose might be different. Depending on the desired level of coherence (2D/3D) the corresponding (2D/3D) VPT allows the robot to transform the image of an observed gesture to an equivalent image as if the gesture were executed by the robot itself.

A. 3D View-Point Transformation

In this approach we explicitly reconstruct the posture of the observed arm in 3D and use fixed points (shoulders and hip) to determine the rigid transformation that aligns the allo-centric and ego-centric image features: We then have:

$$I_e = \mathcal{P}\, T\, Rec(I_a) = VPT(I_a)$$

where P is a orthographic projection matrix, T is a 3D rigid transformation and $Rec(I_a)$ stands for the 3D reconstruction of the arm posture from allo-centric image features. Posture reconstruction and the computation of T are presented in the following sections.

A.1 Posture reconstruction

To reconstruct the 3D posture of the observed arm, we will follow the approach in [29], based on the orthographic camera and articulated arm models presented in Section 2.

Let M_1 and M_2 be the 3D endpoints of an arm-link whose image projections are denoted by $\mathbf{m_1}$ and $\mathbf{m_2}$. Under orthography, the X, Y coordinates are readily computed from image coordinates (simple scale). The depth variation, $dZ = Z_1 - Z_2$, can be determined as:

$$dZ = \pm\sqrt{L^2 - \frac{l^2}{s^2}}$$

where $L = \|M_1 - M_2\|$ and $l = \|m_1 - m_2\|$. If the camera scale factor s is not known beforehand, one can use a different value provided that the following constraint, involving the relative sizes of the arm links, is met:

$$s \geq \max_i \frac{l_i}{L_i} \quad i = 1..4 \tag{3}$$

Fig. 8 illustrates results of the reconstruction procedure. It shows an image of an arm gesture and the corresponding 3D reconstruction, achieved with a single view and considering that s and the arm links proportions were known.

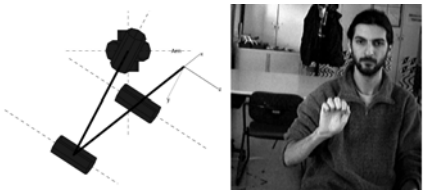

Fig. 8. Left: Reconstructed arm posture. Right: Original view.

With this method there is an ambiguity in the sign of dZ. We overcome this problem by restricting the working volume of the arm. In the future, we will further address this problem and several approaches may be used: (i) optimization techniques to fit the arm kinematic model to the image; (ii) explore occlusions to determine which link is in the foreground; or (iii) use kinematics constraints to prune possible arm configurations.

A.2 Rigid Transformation (T)

A 3D rigid transformation is defined by three angles for the rotation and a translation vector. Since the arm joints are moving, they cannot be used as reference points. Instead, we consider the three points in Fig. 4: left and right shoulders, (M_{ls}, M_{rs}) and hip, M_{hip}, with image projections denoted by $(m_{ls}, m_{rs}, m_{hip})$. The transformation T is determined to translate and rotate these points until they coincide with those of the system's own body.

The translational component must place the demonstrators right shoulder at the image origin (which coincide's

with the system's right shoulder) and can be defined directly in image coordinates:

$$t = -{}^{a}m_{rs}$$

After translating the image features directly, the remaining steps consist in determining the rotation angles to align the shoulder line and the shoulder-hip contour. The angles of rotation along the z, y and x axes, denoted by ϕ, θ and ψ are given by:

$$\begin{aligned} \phi &= \arctan(v_{ls}/u_{ls}) \\ \theta &= \arccos(u_{hip}/L_4) \\ \psi &= \arccos(v_{hip}/L_3) \end{aligned}$$

Hence, by performing the image translation first and the 3D rotation described in this section, we complete the process of aligning the image projections of the shoulders and hip to the ego-centric image coordinates.

B. 2D View-Point Transformation

The $2D$ VPT is used when one is not interested in imitating the depth variations of a certain movement, alleviating the need for a full 3D transformation. It can also be seen as a simplification of the 3D VPT if one assumes that the observed arm describes a fronto-parallel movement with respect to the camera.

The $2D$ VPT performs an image translation to align the shoulder of the demonstrator (${}^{a}m_s$) and that of the system (at the image origin, by definition). The VPT can be written as:

$$VPT({}^{a}m) = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} [{}^{a}m - {}^{a}m_s] \tag{4}$$

and is applied to the image projection of the demonstrator's hand or elbow, ${}^{a}m_h$ or ${}^{a}m_e$.

Notice that when the arm used to imitate is the same as the demonstrator, the imitated movement is a mirror image of the original. If we use a identity matrix in Equation (4) then the movement will be correct. At the image level both the 2D and 3D VPTs have the same result but the 3D posture of the arm is different in the two cases.

From the biological standpoint, the 2D VPT is more plausible than the 3D version. In [28] several imitation behaviors are presented which are not always faithful to the demonstrated gesture: sometimes, people do imitate with the contra-lateral hand, depth is irrelevant in some other cases, movements can be reflections of the original ones, etc. The 3D VPT might be more useful in industrial facilities where gestures should be exactly reproduced.

V. A Bayesian model for program-level (gesture) imitation

We model gesture recognition in a Bayesian framework, which allows to naturally combine *a priori* information and knowledge derived from observations (likelihood). The role played by canonical and mirror neurons will be interpreted within this setting.

Let us assume that we want to recognize (or imitate) a set of gestures, G_i, using a set of *observed* features, F. For the time being, these features can either be represented in the motor space (as mirror neurons seem to do) or in the visual space (directly extracted from images). Let us also define a set of contexts, C_k, related to the scene. Contexts represent the situations that influence the actions or gestures that may occur. Typical examples would be a tennis or golf match (where only some sets of movements are normally executed) or the presence of specific objects in the scene (which tend to be grasped in specific ways).

The prior information is modeled as a probability density function, $p(G_i|C_k)$, describing the probability of each gesture, given a certain context. The observation model is captured in the *likelihood function*, $p(F|G_i, C_k)$, describing the probability of observing a set of (motor or visual) features, conditioned to an instance of the pair gesture and context. The *posterior* density can be directly obtained through Bayesian inference:

$$\begin{aligned} p(G_i|F, C_k) &= \frac{p(F|G_i, C_k)p(G_i|C_k)}{p(F|C_k)} \\ \hat{G}_{MAP} &= arg\max_{G_i} p(G_i|F, C_k) \end{aligned} \tag{5}$$

where $p(F|C_k)$ is just a scaling factor that will not influence the classification.

The MAP estimate, G_{MAP}, is the gesture that maximizes the posterior density in Equation (5). In order to introduce some temporal filtering, features of several images can be considered:

$$p(G_i|F, C_k) = p(G_i|F_t, F_{t-1}, ..., F_{t-N}, C_k),$$

where F_j are the features corresponding to the image at time instant j. The posterior probability distribution can be estimated using a naive approach, assuming independence between the observations at different time instants. The justification for this assumption is that, recognition does not necessarily require the accurate modeling of the density functions. Therefore, the the temporal relationship between the different frames can be ignored. We then have:

$$p(G_i|F_t, F_{t-1}, ..., F_{t-N}, C_k) = \prod_{j=0}^{N} \frac{p(F_{t-j}|G_i, C_k)p(G_i|C_k)}{p(F_{t-j}|C_k)}$$

In the future, we plan to use the information related to the temporal dependency of the different frames to further improve our results and – above all – to allows us to do time predictions.

A. Estimating the prior and the likelihood function

The *prior* density function, $p(G_i|C_k)$ is blended together with evidence from the observations, to shape the final decision. This density can be estimated by the relative frequency of gestures in the training set for each context.

Computing the likelihood function, $p(F|G_i, C_k)$, is more elaborated. As it may correspond to a complex distribution, it will be modeled by a Gaussian mixture, which is

fitted to data points. In what follows we will describe the process of fitting a mixture model to a density, $p(x)$:

$$p(x) = \sum_{j=1}^{K} \pi_j \, p(x|j),$$

where $p(x|j) \sim N(\mu_j, \sigma_j)$ is a Gaussian distribution. For a proper probability density function, we need to ensure that $\sum_{i=1}^{K} \pi_i = 1, \ \pi_i \geq 0$.

The Expectation-Maximization (EM) algorithm is used to estimate the parameters μ_i, σ_i, π_i that best fit the data. The main problem with this solution is the necessity of knowing in advance the number of kernels, K. In [30], [31] there is the option of modifying the number of Gaussian kernels used to best fit the data. The number of kernels can be increased during the learning process, based on a new measure designated as the total *kurtosis*, $\mathcal{K}$:

$$\mathcal{K} \triangleq \int_{-\infty}^{\infty} \left(\frac{x - \mu_j}{\sigma_j} \right)^4 \frac{p(j|x)}{\pi_j} p(x) dx - 3$$

The *kurtosis* measures how far a distribution is from a Gaussian and it is zero for a Gaussian function. If the *kurtosis* is not close to zero for a given kernel, it means that the data are not Gaussian and this kernel must be split. On the other hand, the number of kernels can sometimes be reduced (merged) in order to reduce the model complexity. A "closeness" metric between two kernels, can be defined as follows:

$$d(p_1, p_2) = \frac{\prod_{x_i \in X_1} p_2(x_i) \prod_{x_i \in X_2} p_1(x_i)}{\prod_{x_i \in X_1} p_1(x_i) \prod_{x_i \in X_2} p_2(x_i)}$$

where X_i are the data points used for the estimation of $p_i(x)$.

Two different kernels can be merged if the distance between them is sufficiently small. At the end of this process, we have an estimate of the likelihood function directly from the data, without imposing a particular structure for the underlying distribution. An important point worth mentioning is that this method can cope with clusters that with very irregular shapes and that it automatically adapts to the shape of such clusters..

B. The role of canonical and mirror neurons

The role of canonical neurons in the overall classification system lies essentially in providing the affordances or prior knowledge. In the specific case of grasp actions, affordances are related to graspable objects in the scene and, the various possible ways in which they can be grasped. Canonical neurons are also somewhat involved in the computation of the likelihood function, since it depends both on the *gesture* and *context/object*, thus implicitly defining another level of association between these.

Mirror neurons are also represented in our methodology by the fact that the recognition takes place in the motor space as opposed to visual terms. Also, in the same way as mirror neurons respond to specific grasp actions, each recognized *gesture* constitutes a symbolic motor representation to be used later on, when eliciting more complex composed gestures. Noteworthily, the ability of recognizing someone's gestures is facilitated by the fact that the system knows how to perform those same gestures.

VI. Experiments

We have implemented the modules discussed in the previous sections to build a system able to imitate and recognize gestures.

The following sections present results of hand segmentation, action-level and program-level(gesture) imitation.

A. Automatic body segmentation

For visual segmentation of hand/body we have three steps: background, person and hand segmentation.

The background is estimated, during an initial period of 100 frames, by considering the intensity of each pixel, as a gaussian random variable. After this process, we can estimate the probability of each pixel being part of the background. People are detected, after background subtraction, by template matching. The template consists of two rectangular areas shown in Figure 9. By scaling the template we can estimate the size of the person and the scale parameter, s, of the camera model. In addition, if we need to detect if the person is rotated with respect to the camera, we can scale the template independently in each direction, and estimate this rotation by the ratio between the head height and shoulder width. To detect the hand position we use skin color segmentation, based on the RGB color scheme normalized with the blue channel. The classification of skin pixels was implemented by a feed-forward neural network with three neurons in the hidden layer. The training data were obtained by selecting skin color and the background in sample images. After color classification a *majority* morphological operator is used. The hand is identified as the largest blob found and its position is estimated over time with a Kalman filter. Figure 9 shows the result of this process.

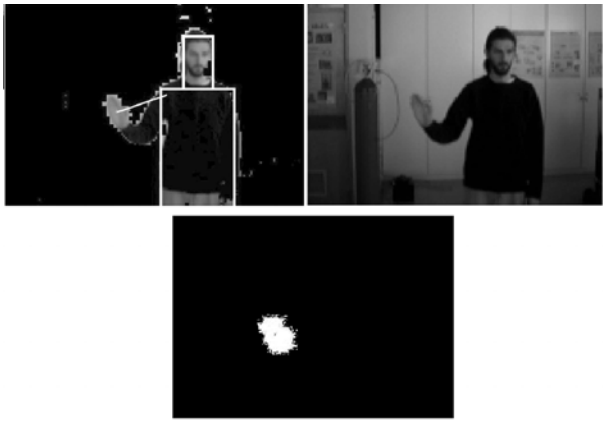

Fig. 9. Vision system. From left to right: original image, background segmentation with human (the frame corresponds to the template matching) and hand detection.

B. Action-Level Imitation

The first step for action-level imitation consists in training the system to learn the *Visuo-Motor Map*, as described in Section III. This is accomplished by a neural network that estimates the VMM while the system performs a large number of arm movements.

The imitation process consists of the following steps: (i) the system observes the demonstrator's arm movements; (ii) the VPT is used to transform these image coordinates to the *ego-image*, as proposed in Section IV and (iii) the VMM generates the adequate joint angle references to execute the same arm movements.

Figure 10 shows experimental results obtained with the $3D$-VPT with the learned VMM (full-arm). To assess the quality of the results, we overlaid the images of the executed arm gestures (wire frame) on those of the demonstrator. The quality of imitation is very good.

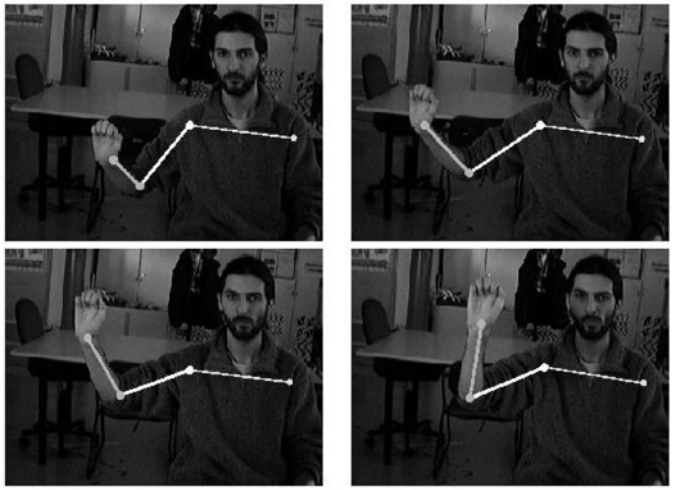

Fig. 10. The quality of the results can be assessed by the coincidence of the demonstrator gestures and the result of imitation.

Figure 11 shows results obtained in real-time (about 5 Hz) when using the $2D$ VPT and the *free-elbow* VMM. The goal of imitating the hand gesture is well achieved but, as expected, there are differences in the configuration of the elbow, particularly at more extreme positions.

Figure 12 shows a result of hand imitation using the *hand-VMM*. This imitation was done after detecting the hand, projecting the image in the PCA base and then using as motor commands the result of the VMM. Visually, is possible to see the quality of reconstruction. For quantitative quality evaluation we see the results in Section III.

These tests show that encouraging results can be obtained with our framework, in realistic conditions.

C. Program-Level (gesture) Imitation

To collect experimental data, we asked several subjects to perform three grasps on different objects [32]. The experiment begins with the subject sitting in a chair with his hand on the table. Finally, the subject is told to grasp the object that is in front of him.

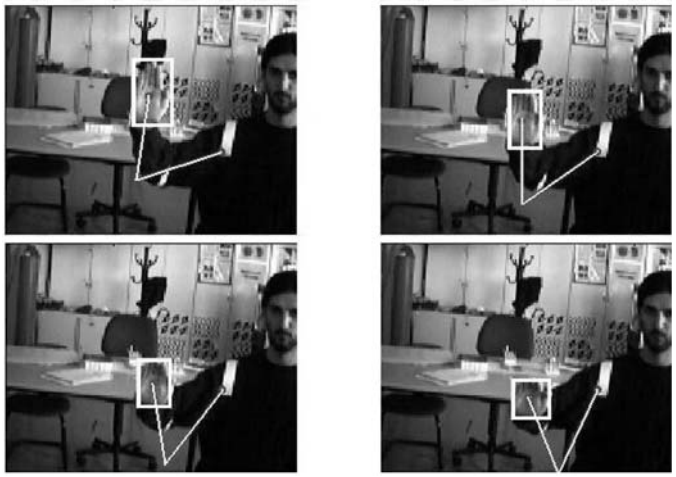

Fig. 11. Family of solutions with different elbow angles, while the hand position is faithfully imitated.

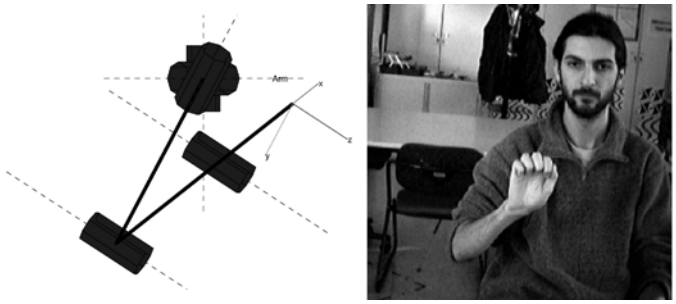

Fig. 12. Action-Level Hand Imitation

The experiments include two types of grasp: power grasp and precision grip. Power grasp is defined when all the hand fingers and palm are in contact with the object. Instead, in precision grip, only the fingertips touch the object.

We considered three different objects: a small sphere, a large sphere and a box. The small sphere is sufficiently small so that only precision grip is allowed. The big sphere allows only power grasps. The box is ambiguous because it allows all possible grasps with different orientations.

Every experiment was repeated several times under varying conditions. The subject and the camera go around the table to cover a large variation of view-points. To record the sequences we use a stereo-pair. In total, we record the experiments from 6 different azimuths (12 if we consider the stereo-pair). In order to record the motor information we used a data-glove [26]. Altogether the data-set contains sixty grasp sequences with three objects, two grasps with six different azimuths.

Figure 13 shows sample images of the data set acquired according to process just described. Notice the multiplicity of grasps, hand appearance and view-points.

Table II shows the obtained classification rates. It allows us to compare the benefits of using motor representations for recognition as opposed to visual information only. The results shown correspond to the use of the ambiguous objects only, when the recognition is more challenging. We varied the number of view-points included in both the training and test sets, so as to assess the degree of view

Fig. 13. Data set illustrating some of the used grasp types: power (left) and precision (right). Altogether the tests were conducted using 60 sequences, from which a total of about 900 images were processed.

	Exp. I (visual)	Exp. II (visual)	Exp. III (visual)	Exp. IV (motor)
	Training			
# Sequences	16	24	64	24
View-points	1	1	4	1
Classif. Rate	100%	100%	97%	98%
# Features	5	5	5	15
# Modes	5-7	5-7	5-7	1-2
	Test			
# Sequences	8	96	32	96
View-points	1	4	4	4
Classif. Rate	100%	30%	80%	97%

TABLE II

GESTURE RECOGNITION RESULTS. THE USE OF MOTOR REPRESENTATIONS GREATLY IMPROVES THE RECOGNITION RATE AND VIEW-POINT INVARIANCE (SEE TEXT FOR DETAILS).

invariance attained by the different methods.

In the first experiment, both the training and test sets correspond to one single view-point. Training was based on 16 grasp sequences, while test was done in 8 (different) sequences. The achieved classification rate was 100%. The number of visual features (number of PCA components) was also tuned and the value of 5 provided good results. The number of modes (gaussians in the mixture) were typically from 5 to 7.

The second experiment shows that this classifier is not able to generalizes to other view-points / camera positions. We used the same training-set as in $Exp.I$, but the test-set is formed with image sequences acquired with 4 different camera positions. In this case, the classification rate is worse than random (30%).

In the third experiment, we added view-point variability in the training set. When sequences from all camera positions are included in the training-set, the classification rate in the test-set drops to 80%. While this is a more acceptable value, it is nevertheless a significant drop from the desired 100%. This result shows that the view-point variation introduces such challenging modifications in the hand appearance that classification errors occur.

The final experiment corresponds to the main approach proposed in this paper. The system learns a visuo-motor map during an initial period of self-observation. Then, the VMM is used to transform the (segmented) hand images to motor information, where classification is conducted. A very high degree of classification was achieved (97 %). Interestingly, the number of modes need for the learning is between 1-2 in this case as opposed to 5-7, when recognition takes place in the visual domain. This also shows that mapping visual data to motor representations, helps clustering the data, as it is now view-point invariant.

Notice that view-point invariance is achieved when the training set only contains sequences from one single viewpoint. These experiments show that motor representations describe the hand better. As only visual information is available during recognition, the process greatly depends on the *VMM*. The results also validate that our approach to estimate the VMM allows recognition to be performed. For the case of only one camera position the quality obtained was very good, if the number of visual features used were 15. As the grasp recognition is done in motor space, our system has the capability of doing program-level imitation.

VII. Conclusions and future work

We presented a general approach for action and program level learning by imitation. Action-level (mimic) imitation involved the *View-Point Transformation* and the *Visuo-Motor Map*, which led to encouraging results. These modules were developed with several properties in mind. Properties of 3D or 2D imitation for the case of the arm-VPT. View-point invariant properties for the case of the hand-VMM, or rigid vs free elbow for the case of the arm-VMM.

For program-level (gesture) imitation an additional module was necessary. The interpretation of observed gestures allows to produce similar gestures/goals, at a later stage. This is similar to the *Mirror System*, where a classification of the observed action's goal is done. Our approach for action level and gesture imitation, draws inspiration from the role that *canonical and mirror neurons* seem to play for grasp recognition or imitation in primates. We adopt a Bayesian formulation, where all these observations are taken into account. We describe how to estimate the prior density and likelihood functions directly from the data.

Our approach dealt explicitly with the sensing problems involved in imitation. Although we relied exclusively in a single camera good results were possible due to:
1. the use of motor information for gesture recognition, inspired by studies on mirror neurons;
2. the use of context (e.g. object affordances) to focus the attention of the recognition system and reduce ambiguities, suggested by canonical neurons;
3. the use of iconic image representations for the hand, as opposed to fitting kinematic models to the video sequence;
4. temporal integration of information;
5. use of self-observation information in order to understand others;

In our opinion, the results obtained are an encouraging step in the endeavor of understanding the biological grounding of imitation and, at the same time, develop the

principles to build more performing and robust machines, able to cope with complex tasks and to interact with humans. The results obtained illustrate the benefits of designing intelligent machines inspired on biological findings and hypotheses, while at the same time, offering robotics technologies as a testbed for such hypotheses.

In the future work, we will test this methodology on an anthropomorphic robot, composed of an arm, articulated hand and binocular head (Figure 14). The robot has been built and the first tests are currently ongoing. In addition, we plan to address more complex tasks where the temporal chaining of elementary gestures must be taken into account. In addition, the goal of the action is expected to become more and more important as the actions themselves become richer.

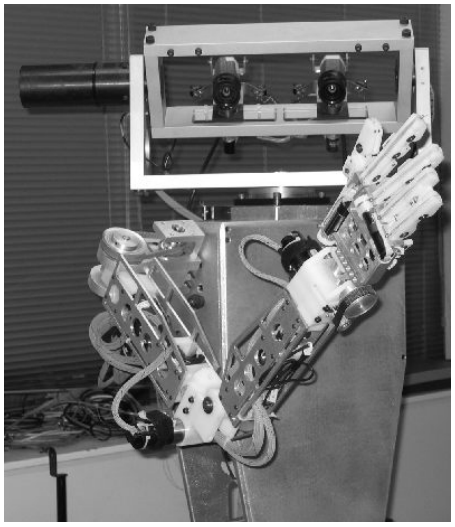

Fig. 14. Baltazar Robot Platform that will be used in future experiments

ACKNOWLEDGMENTS

We thank Matteo Schenatti et al [32], Lira-Lab, University of Genova for recording the data-set. This work was (partially) supported by the FCT, the FCT Programa Operacional Sociedade de Informação (POSI) in the frame of QCA III and the EU-Project IST-2000-28159, Mirror

References

[1] S. Schaal. Is imitation learning the route to humanoid robots. *Trends in Cognitive Sciences*, 3(6), 1999.

[2] J. Yang, Y. Xu, and C.S. Chen. Hidden markov model approach to skill learning and its application to telerobotics. *IEEE Transations on Robotics and Automation*, 10(5):621–631, October 1994.

[3] T. G. Williams, J. J. Rowland, and M. H. Lee. Teaching from examples in assembly and manipulation of snack food ingredients by robot. In *2001 IEEE/RSJ, International Conference on Intelligent Robots and Systems*, pages 2300–2305, Oct.29-Nov.03 2001.

[4] Aaron D'Souza, Sethu Vijayakumar, and Stephan Schaal. Learning inverse kinematics. In *International Conference on Intelligent Robots and Systems*, Maui, Hawaii, USA, 2001.

[5] J.S. Bruner. Nature and use of immaturity. *American Psychologist*, 27:687–708, 1972.

[6] Minoru Asada, Yuichiro Yoshikawa, and Koh Hosoda. Learning by observation without three-dimensional reconstruction. In *Intelligent Autonomous Systems (IAS-6)*, pages 555–560, 2000.

[7] A. Billard and G. Hayes. Drama, a connectionist architecture for control and learning in autonomous robots. *Adaptive Behaviour*, 7(1):35–63, 1999.

[8] Maja J. Matarić. Sensory-motor primitives as a basis for imitation: Linking perception to action and biology to robotics. In C. Nehaniv and K. Dautenhahn, editors, *Imitation in Animals and Artifacts*. MIT Press, 2000.

[9] G. Metta, G. Sandini, L. Natale, and F.Panerai. Sensorimotor interaction in a developing robot. In *First International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*, pages 18–19, Lund, Sweden, September 2001.

[10] Giorgio Metta, Riccardo Manzotti, Francesco Panerai, and Giului Sandini. Development: Is it the right way towards humanoid robotics? In *IAS*, Venice, Italy, July 2000.

[11] M. Asada, K.F. MacDorman, H. Ishiguro, and Y. Kuniyoshi. Cognitive developmental robotics as a new paradigm for the design of humanoid robots. *Robotics and Automation*, 37:185–193, 2001.

[12] V.G. Payne and L.D. Isaacs. *Human Motor Development: A Lifespan Approach*. Mayfield Publishing Company, 2002.

[13] L. Fadiga, L. Fogassi, V. Gallese, and G. Rizzolatti. Visuomotor neurons: ambiguity of the discharge or 'motor' perception? *International Journal of Psychophysiology*, 35, 2000.

[14] V.S. Ramachandran. Mirror neurons and imitation learning as the driving force behind the great leap forward in human evolution. *Edge*, 69, June 2000.

[15] A. Murata, L. Fadiga, L. Fogassi, V. Gallese, V. Raos, and G. Rizzolatti. Object representation in the ventral premotor cortex (area f5) of the monkey. *Journal of Neurophysiology*, 78(4):2226–2230, October 1997.

[16] Erhan Oztop. *Modeling the Mirror: Grasp Learning and Action Recognition*. PhD thesis, University of Southern California, August 2002.

[17] J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, 1979.

[18] L. Fogassi, V. Gallese, G. Buccino, L. Craighero, L. Fadiga, and G. Rizzolatti. Cortical mechanism for the visual guidance of hand grasping movements in the monkey: A reversible inactivation study. *Brain*, 124(3):571–586, March 2001.

[19] James M. Rehg and Takeo Kanade. Visual tracking of high DOF articulated structures: an application to human hand tracking. In *ECCV (2)*, pages 35–46, 1994.

[20] Ying Wu and Thomas S. Huang. Capturing articulated human hand motion: A divide-and-conquer approach. In *ICCV (1)*, pages 606–611, 1999.

[21] Michael J. Black and Allan D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. In *ECCV (1)*, pages 329–342, 1996.

[22] D. M. Gavrila. The visual analysis of human movement: A survey. *CVIU*, 73(1):82–98, 1999.

[23] James M. Rehg and Takeo Kanade. Model-based tracking of self-occluding articulated objects. In *ICCV*, pages 612–617, 1995.

[24] Ying Wu and Thomas S. Huang. View-independent recognition of hand postures. In *CVPR*, pages 88–94, June 2000.

[25] Richard L. Gregory. *Eye and Brain, The Psychology of Seeing*. Princeton University Press, Princeton, New Jersey, 1990.

[26] CyberGlove. http://www.immersion.com.

[27] Manuel Cabido-Lopes and José Santos-Victor. Visual transformations in gesture imitation: What you see is what you do. In *to appear in International Conference on Robotics and Automation*, Taiwan, 2003.

[28] Philippe Rochat. Ego function of early imitation. In Andrew N. Meltzoff and Wolfgang Prinz, editors, *The Imitative Mind*. Cambridge University Press, 2002.

[29] C. Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. *Computer Vision and Image Understanding*, 80, 2000.

[30] Paul M. Baggenstoss. Statistical modeling using gaussian mixtures and hmms with matlab. http://www.npt.nuwc.navy.mil/Csf/htmldoc/pdf/.

[31] N. Vlassis and A. Likas. A kurtosis-based dynamic approach to gaussian mixture modeling. *IEEE Trans. Systems, Man, and Cybernetics, Part A*, 29:393–399, 1999.

[32] Matteo Schenatti, Lorenzo Natale, Giorgio Metta, and Giulio Sandini. Object grasping data-set. Lira Lab, University of Genova, Italy, 2003.