Affordance-based imitation learning in robots

Manuel Lopes and Francisco S. Melo and Luis Montesano

Abstract— **In this paper we build an imitation learning algorithm for a humanoid robot on top of a general world model provided by learned object affordances. We consider that the robot has previously learned a task independent affordance-based model of its interaction with the world. This model is used to recognize the demonstration by another agent (a human) and infer the task to be learned. We discuss several important problems that arise in this combined framework, such as the influence of an inaccurate model in the recognition of the demonstration. We illustrate the ideas in the paper with some experimental results obtained with a real robot.**

I. INTRODUCTION

Imitation is a very powerful method to transfer knowledge among different agents. In current complex humanoid robotic systems, this capability provides an efficient tool to program robots by demonstration. Therefore, several systems for imitation in robots have been proposed in the literature (see [1] for a review).

When implementing imitation learning one must consider two fundamental problems: selection of the goal of imitation (imitation metric) and description of the observed motion in terms of the imitator's own motor capabilities (body correspondence). These problems have been addressed in different ways in the literature. Possible approaches include hand-coding of the correspondence between actions [2], defining correspondences between effects instead of between actions [3], learning basic object properties to elicit compatible actions [4] or describing world-state transitions at the trajectory level [5]. However, most such approaches fail to considered sequences of actions. Those that consider them exhibit simple mimicking behavior and not real imitation, since they merely copy the observed actions [6].

In this paper we adopt the formalism in [7]. This approach to imitation provides a unified model that spans several imitation and imitation-like behaviors. Behaviors like emulation, contextual learning and social facilitation are explained in terms of the information extracted from the demonstration. The core of this framework is the Bayesian inverse reinforcement learning algorithm [8] that extracts the reward function from the observed demonstration. As in standard reinforcement learning, all task information is encoded in this reward function. Therefore, this task (reward) must be *explicitly determined* for the agent to be able to generalize (and, in some cases, even improve) the observed behavior to situations that were not demonstrated. Based on this reward, the agent can determine the optimal policy, yielding the final behavior and completing the imitation learning process.

In order to recover the reward function, the robot must possess two capabilities: (i) it must be able to interpret the demonstration in terms of its own action repertoire; and (ii) it must know the world dynamics, *i.e.,* the state transition probabilities. These requirements are typically ensured in a task-specific manner. However, this hinders the re-use of previously acquired knowledge. When learning many different tasks, this *a priori* information can reduce the complexity of the learning problem.

In this paper, these prerequisites (state-action recognition capabilities and known world model) are fulfilled by means of a general task-independent model for affordances [9]. Introduced by Gibson [10], affordances define the relation between an agent and the environment by means of its motor and sensing capabilities. Thus, they relate the agent's actions to their effects on the surrounding objects.

The combination of affordances [11] and imitation [7] endows the robot with learning capabilities that can be classified as *real imitation* in the context of [7], [12], [13]. Real imitation explains complex learning behaviors, where (i) learning the task is only possible given the demonstration; and (ii) there is a generalization of the observed behavior and not a simple copy of the observed motion. In terms of our approach, this means that the robot *determines the task* from the observed demonstration; it then chooses its actions so as to accomplish this task, reproducing the behavior of the demonstrator and generalizing this behavior in situations never observed before.

Decoupling the world description (affordances) and imitation has two main advantages. First, the robot is able to re-use previous knowledge in different tasks. This is important since learning is a very tedious task that requires extensive experience and time. Second, the learning process in itself is simplified.

◇

Our approach, summarized in Figure 1, is part of a more general developmental architecture for social robots [14]. Artificial development, strongly motivated by the motor development in biological systems, suggests that if behaviors are built on top of others, learning complexity is strongly reduced. Similarly, our task independent knowledge (affordances) is used to facilitate learning by imitation.

We assume that the robot is already able to interact with the world by means of several action primitives such as grasping, tapping and touching nearby objects (we refer to [14] for further details). By repeatedly interacting with

This work was partially supported by the POS_C financing program, that includes FEDER funds. All authors are with the Institute for Systems and Robotics, Instituto Superior Técnico, Lisbon, Portugal. {macl,fmelo,lmontesano}@isr.ist.utl.pt

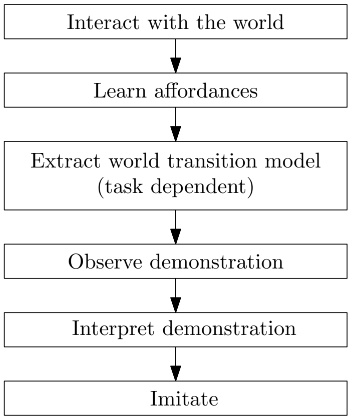


Fig. 1. Diagram describing the approach in this paper. Imitation is possible after learning about the world properties by autonomous interaction with it.

different objects, the robot learns the affordances relating its action primitives and their effects on objects [9], thus acquiring the ability to predict the consequences of its actions and to recognize the actions of other agents. Notice that all information acquired up to this point is task independent.

When learning a specif task, the system will need to adapt the description provided by the affordances to obtain a dynamic model for the specific problem at hand. This is represented in Figure 1 in the block marked as "Extract world transition model". This transition model is already task-specific and is obtained from the affordance-based model with the help of a human operator. The robot can then observe a demonstration of a specific task to be learned, interpret it and imitate.

The method used to determine the task underlying the demonstration is supported in the idea of *inverse reinforcement learning* [15], [16]. Given a demonstration, the imitator determines the underlying task by considering the likelihood of the observed actions. The method used is a basic variation of the algorithm in [8] and accommodates for incomplete/inaccurate demonstrations.

◇

The contribution of this paper consists in the joint architecture for imitation based on an affordances model [11] and a general imitation learning formalism [7], summarized in Figure 1. We argue that affordances provide sufficient information to learn by imitation. In particular, they combine in a single structure an "action interpreter" and a world transition model. As a result, previously learned, task-independent knowledge is used to recover the appropriate information from the observations and elicit a rich imitation behavior. The experimental results using a real humanoid robot platform suggest that the proposed architecture is adequate to implement learning by imitation in real-world tasks.

The paper is organized as follows. We start by showing how the affordances knowledge is acquired in Section II. We describe the framework used for imitation in Section III. In Section IV we combine both approaches in our affordances-based imitation learning paradigm. Finally, in Section V we present the results obtained by implementing imitation in a real-world setting and conclude the paper in Section VI.

II. AFFORDANCE MODELING AND LEARNING

In this section we describe how to model affordances using a Bayesian network (BN). We briefly review standard representation, inference and learning concepts using BNs and describe their application to the problem of affordance learning.

We consider a robot that has available a repertoire of actions to interact with the world. The robot is also able to detect and extract information from the objects around him. We pose the affordance learning problem at this level of abstraction, where the main entities are the available *actions* $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$, the *objects*, described by their observable *features* $\mathcal{F} = \{f_1, \ldots, f_m\}$ and the *effects* $\mathcal{E} = \{e_1, ..., e_p\}$. The final goal is to determine the multiple relations between the random variables representing actions, features of objects and effects (see Figure 2). Notice the difference between object features and effects: object features can be acquired by simple observation, whereas effects can only be observed as a consequence of interaction. The relations between these entities are inferred as the robot acts upon each object and observes the resulting effects.

A. Learning affordances

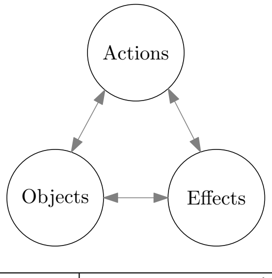
We use a probabilistic graphical model known as Bayesian networks [17] to encode the dependencies between actions, object features and the effects of those actions. A BN is a probabilistic directed graphical model where the nodes represent random variables and the (lack of) arcs represent conditional independence assumptions.

The learning of affordances with a BN is performed in two phases. First the structure is learned using Markov Chain Monte Carlo (MCMC) [18]. Once the structure of the network has been established, the parameters of each node are estimated using a Bayesian approach [18]. The estimated parameters can be subsequently updated on-line, allowing to incorporate information provided by new experiences.

Affordances encode the probabilistic relations between actions and perceptions (object features and effects). Figure 4 shows how the learned network captures the structural dependencies between actions, object features and effects. The model is able to distinguish irrelevant properties of the objects, *i.e.,* object features not influencing action outcomes. This "feature selecting" effect of the structure learning method is fundamental in planning because task execution is often linked to the object properties and only to a lesser extent to the objects themselves.

B. Using affordances

Since the structure of the BN encodes the relations between actions, object features and effects, it is now possible to compute the distribution of a variable or group of variables given the value of other variables. To this, we use the junction tree algorithm [19] to compute the distribution of the variables of interest. We emphasize that it is not necessary to know the values of all the variables to perform inference.



Inputs	Outputs	Function
(O, A)	E	Effect prediction
(O, E)	A	Action recognition/planning
(A, E)	O	Object recognition/selection

Fig. 2. Affordances as relations between (A)ctions, (O)bjects and (E)ffects, that can be used to solve different problems: prediction, action selection or object selection.

We are now able to use the affordance knowledge to solve the several problems described in Figure 2 simply by computing the appropriate distributions. In particular, for purposes of imitation, we are interested in

1) interpret actions performed by others in terms of the agent's own actions, i.e. by matching the effects;
2) estimate a dynamic world model by predicting the effects of each action in terms of the world state.

It is important to remark that the effect prediction capabilities do not yield immediately the transition model, but allow the construction of one such model.

III. IMITATION LEARNING

In this section we describe the fundamental process by which the robot perceives the task to be learnt after observing the demonstration by another agent (human). To this purpose we adopt the formalism described in [7] that we now briefly describe.

Formalism: At each time instant, the robot must choose an action from its repertoire of action primitives $\mathcal{A}$, depending on the state of the environment. We represent the state of the environment at time t by X_t and let $\mathcal{X}$ be the (finite) set of possible environment states. This state evolves according to the transition probabilities

$$\mathbb{P}\left[X_{t+1} = y \mid X_t = x, A_t = a\right] = \mathsf{P}_a(x, y), \qquad (1)$$

where A_t denotes the robot's action primitive at time t. The action-dependent transition matrix P thus describes the dynamic behavior of the process $\{X_t\}$.

We assume that the robot is able to recognize the actions performed during the demonstration.[1] Baring this assumption in mind, we consider that the demonstration consists of a sequence $\mathcal{H}$ of state-action pairs

$$\mathcal{H} = \{(x_1, a_1), (x_2, a_2), \ldots, (x_n, a_n)\}.$$

Each pair (x_i, a_i) exemplifies to the robot the expected action (a_i) in each of the states visited during the demonstration (x_i). From this demonstration, the robot is expected to perceive what the demonstrated task is and, eventually by experimentation, learn how to perform it optimally. A decision-rule determining the action of the robot in each state of the environment is called *a policy* and is denoted as a map $\delta : \mathcal{X} \longrightarrow \mathcal{A}$. The robot should then *infer the task* from the demonstration and *learn the corresponding optimal policy*, that we henceforth denote by δ^*.

In our adopted formalism, the task can be defined using a function $r : \mathcal{X} \longrightarrow \mathbb{R}$ describing the "desirability" of each particular state $x \in \mathcal{X}$. This function r works as a *reward* for the robot and, once r is known, the robot should choose its actions to maximize the functional

$$J(x, \{A_t\}) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t r(X_t) \mid X_0 = x\right],$$

where γ is a discount factor between 0 and 1 that assigns greater importance to those rewards received in the immediate future than to those in the distant future. We remark that, once r is known, the problem falls back to the standard formulation of reinforcement learning [20].

The relation between the function r describing the task and the optimal behavior rule can be evidenced by means of the function V_r given by

$$V_r(x) = \max_{a \in \mathcal{A}} \left[r(x) + \gamma \sum_{y \in \mathcal{X}} \mathsf{P}_a(x, y) V_r(y)\right] \qquad (2)$$

The value $V_r(x)$ represents the expected (discounted) reward accumulated along a path of the process $\{X_t\}$ starting at state x, when the optimal behavior rule is followed. The optimal policy associated with the reward function r is thus given by

$$\delta_r(x) = \arg\max_{a \in \mathcal{A}} \left[r(x) + \gamma \sum_{y \in \mathcal{X}} \mathsf{P}_a(x, y) V^*(y)\right]$$

The computation of δ_r (or, equivalently, V_r) given P and r is a standard problem and can be solved using any of several standard methods available in the literature [20].

Methodology: In the formalism just described, the fundamental imitation problem lies in the estimation of the function r from the observed demonstration $\mathcal{H}$. Notice that this is closely related to the problem of *inverse reinforcement learning* as described in [16]. We adopt the method described in [7], which is a basic variation of the *Bayesian inverse reinforcement learning* (BIRL) algorithm in [8].

For a given r-function, the *likelihood of a pair* $(x, a) \in \mathcal{X} \times \mathcal{A}$ is defined as

$$L_r(x, a) = \mathbb{P}\left[(x, a) \mid r\right] = \frac{e^{\eta Q_r(x,a)}}{\sum_{b \in \mathcal{A}} e^{\eta Q_r(x,b)}},$$

where $Q_r(x, a)$ is defined as

$$Q_r(x, a) = r(x) + \gamma \sum_{y \in \mathcal{X}} \mathsf{P}_a(x, y) V_r(y)$$

[1] We will discuss the validity of this assumption further ahead.

and V_r is as in (2). The parameter η is a user-defined *confidence parameter* that we describe further ahead. The value $L_r(x, a)$ translates the plausibility of the choice of action a in state x when the underlying task is described by r. Given a demonstration sequence

$$\mathcal{H} = \{(x_1, a_1), (x_2, a_2), \ldots, (x_n, a_n)\}.$$

the corresponding likelihood is

$$L_r(\mathcal{H}) = \prod_{i=1}^{n} L_r(x_i, a_i).$$

The method uses MCMC to estimate the distribution over the space of possible r-functions (usually a compact subset of $\mathbb{R}^p, p > 0$), given the demonstration [8]. It will then choose the maximum *a posteriori* r-function. Since we consider a uniform prior for the distribution, the selected reward is the one whose corresponding optimal policy "best matches" the demonstration. The confidence parameter η determines the "trustworthiness" of the method: it is a user-defined parameter that indicates how "close" the demonstrated policy is to the optimal policy [8].

Some important remarks are in order. First of all, to determine the likelihood of the demonstration for each function r, the algorithm requires the transition model in P. If such transition model is not available, then the robot will only be able to *replicate particular aspects of the demonstration*. However, as argued in [7], the imitative behavior obtained in these situations may not correspond to actual imitation.

Secondly, it may happen that the transition model available is *inaccurate*. In this situation (and unless the model is significantly inaccurate) the robot should still be able to perceive the demonstrated task. Then, given the estimated r-function, the robot may only be able to determine a *sub-optimal policy* and will need to resort to *experimentation* to improve this policy. We discuss these aspects in greater detail in the continuation.

IV. COMBINING AFFORDANCES WITH IMITATION LEARNING

In this section we discuss in greater detail how the information provided by the affordances described in Section II can be combined with the imitation learning approach described in Section III. We discuss the advantages of this approach as well as several interesting issues that arise from this combination.

In the previous section, we assumed the robot to be able to recognize the actions performed by the demonstrator. This action recognition needs not to be explicit, *i.e.,* the agent needs not to determine the action taken by the demonstrator. Instead, it needs only to *interpret* the observed action in terms of its own action repertoire. This interpretation may rely on the observed state transition or in the corresponding effects. It is important to emphasize that transitions and effects are different concepts: the same transition may occur from different actions/effects and the same effect can be observed in different transitions. To clarify this distinction, consider moving or jumping from one place to the other, the effects are different but the transition is the same. Or motions with different speeds that can result in the same effect, i.e. motion, and different transition.

We should emphasize that if no action recognition/interpretation takes place, the robot will generally be able to learn only how to replicate particular elements of the observed demonstration. In our approach we want the robot to *learn the task* more than to replicate particular aspects of the observed demonstration. As seen in Section II, affordances provide a functional description of the robot's interaction with its surroundings as well as the action-recognition capabilities necessary to implement imitation.

Affordance-based action recognition/interpretation works as follows. For each demonstrated action, the robot observes the corresponding effects. The affordance network is then used to estimate the probability of each action in the robot's action repertoire given the observed effects, and the action with greatest probability is picked as the observed action. Clearly, there will be some uncertainty in the identification of the demonstrated action, but as will be seen in the experimental section, this does not significantly affect the performance of the learning algorithm.

On the other hand, given the demonstration—consisting on a sequence of state-action pairs—the robot should be able to *infer* the task to be learnt. This means, in particular, that once the robot realizes the task to be learnt, it should be able to learn how to perform it *even in situations that were never demonstrated*.

Choosing between two policies generally requires the robot to have a model of the world. Only with a model of the world will the robot have the necessary information to realize *what task is more suitably accomplished by the demonstrated policy*. If no model of the world is available, then the robot will generally only *repeat the observed action pattern*, with no knowledge on what the underlying task may be. Also, the absence of a model will generally prevent the robot from *generalizing* the observed action pattern to situations never demonstrated.

As argued in Section II, affordances empower the robot with the ability to predict the effect of its actions in the surrounding environment. Once the adequate state-space for a particular task is settled, the information embedded in the affordance network can be used to extract the dynamic model describing the state evolution for the particular task at hand. This action-dependent dynamic model consists of the transition matrix P described in Section III.

Figure 3 depicts the fundamental elements in the imitation learning architecture described, corresponding to the block "Interpret demonstration" in Figure 1.

Several remarks are in order. First of all, the affordance network is *task independent* and can be used to provide the required information for different tasks. Notice that the interaction model described in the affordance network could be enriched with further information concerning the state of the system for a specific task. This would make the extraction of the transition model automatic, but would render the affordance network task-dependent. This and the

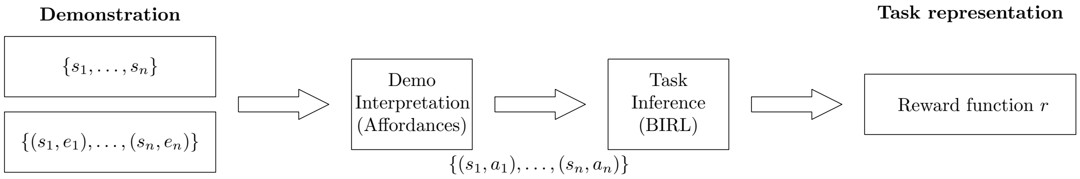

Demonstration
$\{s_1, \ldots, s_n\}$
$\{(s_1, e_1), \ldots, (s_n, e_n)\}$
Demo Interpretation (Affordances)
$\{(s_1, a_1), \ldots, (s_n, a_n)\}$
Task Inference (BIRL)
Task representation
Reward function r

Fig. 3. Representation of the fundamental elements of an imitation learner.

very definition of affordances justifies the use of a more general affordance model, even if requiring the transition model to be separately extracted for each particular task. This means that imitation can be successfully implemented in different tasks, provided a single, sufficiently general and task-independent model of interaction is available (such as the one provided by the affordances).

The second important observation is concerned with the fact that the affordance network is learnt from interaction with the world. The combination of both learning blocks (affordance learning and imitation learning) gives rise to a complete architecture that allows the acquisition of skills ranging from simple action-recognition to complex sequential tasks.

In the next section, we implement this combined architecture in a real-robot. We illustrate the learning of a sequential task that relies on the interaction model described in the affordance network. We discuss the sensitivity of the imitation learning to action recognition errors.

V. EXPERIMENTAL RESULTS

In this section, we implement our imitation learning methodology in a sequential task. For all experiments we used BALTAZAR [21], a robotic platform consisting of a humanoid torso with one anthropomorphic arm and hand and a binocular head (see Figure 7).

Prior to the experiments on imitation, the robot had already interacted with different objects and learned the affordance network for each of the 3 action primitives “Grasp”, “Tap” and “Touch”. The objects were classified according to the visual features “Color”, “Shape” and “Size” and the effects of each action in the objects were described in terms of “Velocity”, “Contact” and “Object-hand distance”. The structure of the corresponding affordances network is depicted in Figure 4 (see [9] for further details). To implement the imitation learning algorithm in the robot we considered a simple recycling game, where the robot must separate different objects according to their shape (Figure 5). In front of the robot are two slots (Left and Right) where 3 types of objects can be placed: Large Balls, Small Balls and Boxes. The boxes should be dropped in a corresponding container and the small balls should be tapped out of the table. The large balls should be touched upon, since the robot is not able to efficiently manipulate them. Every time a large ball is touched, it is removed from the table by an external user. Therefore, the robot has available a total of 6 possible actions: Touch Left (TcL), Touch Right (ThR), Tap Left (TpL), Tap Right (TpR), Grasp Left (GrL) and Grasp Right (GrR).

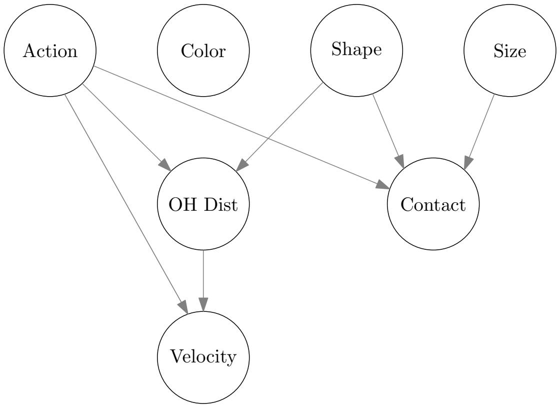


Fig. 4. Bayesian network describing the learned affordances. The general topology of the network is action-independent; only the network parameters change with the actions.

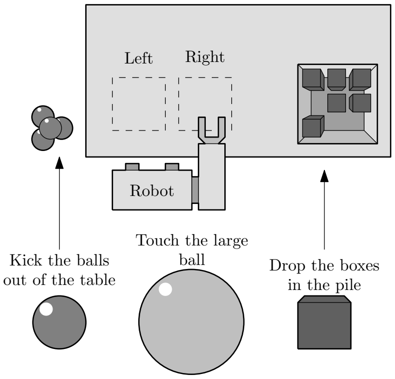


Fig. 5. Simple recycling game.

For the description of the process $\{X_t\}$ for the task at hand, we considered a state-space consisting of 17 possible states. Of these, 16 correspond to the possible combinations of objects in the two slots (including empty slots). The 17th state is an invalid state that accounts for the situations where the robot’s actions do not succeed. As described in Section III, determining the dynamic model consists of determining the transition matrix P by considering the possible effects of each action in each possible object. From the affordances in Figure 4 the transition model for the actions on each object are shown in Figure 6. Notice that, if the robot taps a ball on the right while an object is lying on the left, the ball will most likely remain in the same spot. However, since this behavior arises from the presence of two objects, *it is not captured in the transition model* obtained from the affordances. This means that the transition model extracted from the affordances necessarily includes some inaccuracies.

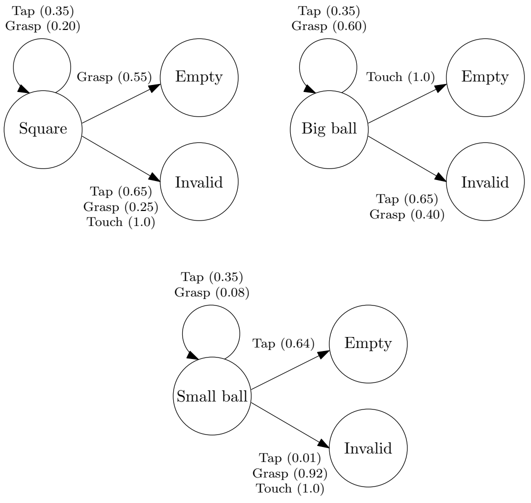


Fig. 6. Transition diagrams describing the transitions for each slot/object.

To test the imitation, we provided the robot with an error-free demonstration of the optimal behavior rule. As expected, the robot was successfully able to reconstruct the optimal policy. We also observed the learned behavior when the robot was provided with *two* different demonstrations, both optimal, as described in Table I. Each state is represented as a pair (S_1, S_2) where each S_i can take one of the values "Ball" (Big Ball), "ball" (Small Ball), "Box" (Box) or $\emptyset$ (empty). The second column of the table lists the observed actions for each state, and the third column lists the learned policy. Notice that, once again, the robot was able to reconstruct an optimal policy, by choosing one of the demonstrated actions in those states where different actions were observed.

In another experiment, we provided the robot with an *incomplete and inaccurate* demonstration. In particular, the action at state ($\emptyset$, Ball) was never demonstrated and the action at state (Ball, Ball) was *wrong*. Table I shows the demonstrated and learned policies. Notice that in this particular case the robot was able to recover the *correct policy*, even with an incomplete and inaccurate demonstration,.

In Figure 7 we illustrate the execution of the optimal learned policy for the initial state (Box, SBall).[2]

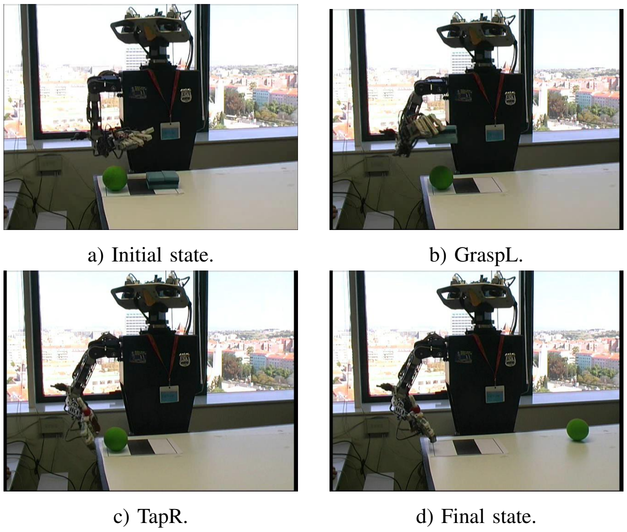
We then tested the action recognition capabilities of the robot when using the information provided by the affordances. A demonstrator performed several actions upon different objects and the robot classified these actions according to the observed effects (see Figure 8). The accuracy of the recognition varied, depending on the performed action, on the demonstrator and on the speed of execution, but for all actions the recognition was successful with an error rate between 10% and 15%. The errors in action recognition are not surprising and are justified by the different viewpoints during the learning of the affordances and during the demonstration. In other words, the robots learns the affordances by looking at its own body motion, but the action recognition is conducted from an external point-of-view. In terms of the image, this difference in viewpoints translates in differences on the observed trajectories and velocities, leading to some occasional mis-recognitions. We refer to [6] for a more detailed discussion of this topic.

To assess the sensitivity of the imitation learning module to the action recognition errors, we tested the learning algorithm for different error recognition rates. For each error rate, we ran 100 trials. Each trial consists of 45 state-action pairs, corresponding to three optimal policies. The obtained

TABLE I

EXPERIMENT 1: ERROR FREE DEMONSTRATION (DEMONSTRATED AND LEARNED POLICIES). EXPERIMENT 2: INACCURATE, INCOMPLETE DEMONSTRATION (DEMONSTRATED AND LEARNED POLICIES), THE BOXED DEMONSTRATION CORRESPOND TO THE INCOMPLETE AND INACCURATE DEMONSTRATIONS.

State	**Demo1**	**Learned**	**Demo2**	**Learned**
($\emptyset$, Ball)	TcR	TcR	$\boxed{-}$	TcR
($\emptyset$, Box)	GrR	GrR	GrR	GrR
($\emptyset$, ball)	TpR	TpR	TpR	TpR
(Ball, $\emptyset$)	TcL	TcL	TcL	TcL
(Ball, Ball)	TcL,TcR	TcL,TcR	$\boxed{\text{GrR}}$	TcL
(Ball, Box)	TcL,GrR	GrR	TcL	TcL
(Ball, ball)	TcL	TcL	TcL	TcL
(Box, $\emptyset$)	GrL	GrL	GrL	GrL
(Box, Ball)	GrL,TcR	GrL	GrL	GrL
(Box, Box)	GrL,GrR	GrR	GrL	GrL
(Box, ball)	GrL	GrL	GrL	GrL
(ball, $\emptyset$)	TpL	TpL	TpL	TpL
(ball, ball)	TpL,TcR	TpL	TpL	TpL
(ball, Box)	TpL,GrR	GrR	TpL	TpL
(ball, ball)	TpL	TpL	TpL	TpL



a) Initial state.

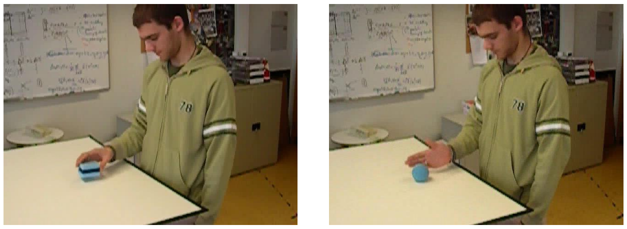
b) GraspL.

c) TapR.

d) Final state.

Fig. 7. Execution of the learned policy in state (Box, SBall).

[2] For videos showing additional experiences see http://vislab.isr.ist.utl.pt/baltazar/demos/



a) Grasp.

b) Tap.

Fig. 8. Testing action recognition from a demonstrator.

results are depicted in Figure 9.

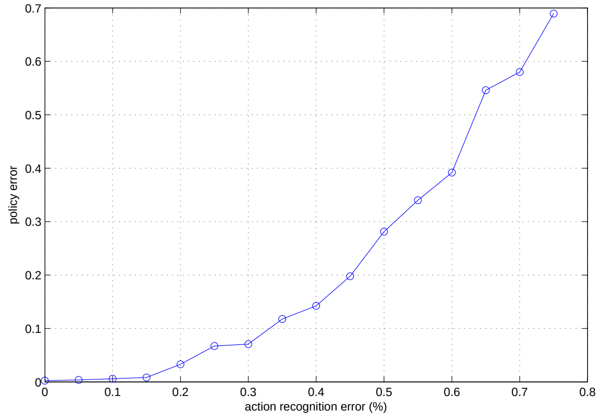


Fig. 9. Percentage of wrong actions in the learned policy as the action recognition errors increase.

As expected, the error in the learned policy increases as the number of wrongly interpreted actions increases. Notice, however, that for small error rates ($\leq 15\%$) the robot is still able to recover the demonstrated policy with an error of only 1%. In particular, if we consider the error rates of the implemented action recognition method (between 10% and 15%), the optimal policy is accurately recovered. This allows us to conclude that action recognition using the affordances is sufficiently precise to ensure the recovery of the demonstrated policy.

VI. Conclusions

In this paper we presented a combined architecture for robotic imitation, based on an affordances model [9], [11] and a general imitation learning method/formalism [7]. The model of interaction provided by the affordances endows the robot with sufficient knowledge to be able to learn complex behaviors by imitation.

We implemented our methodology in humanoid robotic torso. The robot had to learn a sequential task after observing a person execute it. We emphasize that there is no reinforcement given to the robot by any external user and no supervision is conducted on any step of the learning process. The task description is extracted by observing the demonstrator execute it. In the conducted experiments, the robot was able to successfully determine the underlying task by relying on the knowledge provided by the affordances, relating the actions of the robot with the resulting effects on objects.

The results showed the method to be robust even in the presence of incomplete and incoherent demonstrations and also under action-recognition errors.

Future work should address the problem of recovering the (task-specific) transition model from the (task-independent) model provided by the affordances. At the present stage, this is accomplished by an external user. We are interested in developing an automated method to perform this task.

References

[1] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," *Phil. Trans. of the Royal Society of London: Series B, Biological Sciences*, vol. 358, no. 1431, 2003.

[2] A. Alissandrakis, C. L. Nehaniv, and K. Dautenhahn, "Action, state and effect metrics for robot imitation," in *15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 06)*, Hatfield, United Kingdom, 2006, pp. 232–237.

[3] H. Kozima, C. Nakagawa, and H. Yano, "Emergence of imitation mediated by objects," in *2nd Int. Workshop on Epigenetic Robotics*, 2002.

[4] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini., "Learning about objects through action: Initial steps towards artificial cognition," in *IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, 2003.

[5] A. Billard, Y. Epars, S. Calinon, G. Cheng, and S. Schaal, "Discovering optimal imitation strategies," *Robotics and Autonomous Systems*, vol. 47:2-3, 2004.

[6] M. Lopes and J. Santos-Victor, "Visual transformations in gesture imitation: What you see is what you do," in *IEEE Int. Conf. Robotics and Automation*, 2003.

[7] F. Melo, M. Lopes, J. Santos-Victor, and M. I. Ribeiro, "A unified framework for imitation-like behaviors," in *4th International Symposium in Imitation in Animals and Artifacts*, Newcastle, UK, April 2007.

[8] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning," in *20th Int. Joint Conf. Artificial Intelligence*, 2007.

[9] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor, "Modeling affordances using bayesian networks," in *IEEE - Intelligent Robotic Systems (IROS'06)*, USA, 2007.

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

[11] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor, "Affordances, development and imitation." in *IEEE - International Conference on Development and Learning*, London, UK, July 2007.

[12] S. Schaal, "Is imitation learning the route to humanoid robots," *Trends in Cognitive Sciences*, vol. 3(6), pp. 233–242, 1999.

[13] R. W. Byrne, "Imitation of novel complex actions: What does the evidence from animals mean?" *Advances in the Study of Bahaviour*, vol. 31, pp. 77–105, 2002.

[14] M. Lopes and J. Santos-Victor, "A developmental roadmap for learning by imitation in robots," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 37, no. 2, April 2007.

[15] A. Y. Ng and S. J. Russel, "Algorithms for inverse reinforcement learning," in *Proc. 17th Int. Conf. Machine Learning*, 2000.

[16] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the 21st International Conference on Machine Learning (ICML'04)*, 2004, pp. 1–8.

[17] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[18] D. Heckerman, "A tutorial on learning with bayesian networks," in *In M. Jordan, editor, Learning in graphical models.* MIT Press, 1998.

[19] C. Huang and A. Darwiche, "Inference in belief networks: A procedural guide," *International Journal of Approximate Reasoning*, vol. 15, no. 3, pp. 225–263, 1996.

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

[21] M. Lopes, R. Beira, M. Praça, and J. Santos-Victor, "An anthropomorphic robot torso for imitation: design and experiments." in *International Conference on Intelligent Robots and Systems*, Sendai, Japan, 2004.