IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC 2014)
Espinho, Portugal, 14-15 May 2014

Learning Visual Affordances of Objects and Tools through Autonomous Robot Exploration

Afonso Gonçalves[1], Giovanni Saponaro[1], Lorenzo Jamone[1], Alexandre Bernardino[1]

***Abstract*— Endowing artificial agents with the ability of predicting the consequences of their own actions and efficiently planning their behaviors based on such predictions is a fundamental challenge both in artificial intelligence and robotics. A computationally practical yet powerful way to model this knowledge, referred as objects affordances, is through probabilistic dependencies between actions, objects and effects: this allows to make inferences across these dependencies, such as i) predicting the effects of an action over an object, or ii) selecting the best action from a repertoire in order to obtain a desired effect over an object. We propose a probabilistic model capable of learning the mutual interaction between objects in complex tasks that involve manipulation, where one of the objects plays an active tool role while being grasped and used (e.g., a hammer) while another item is passively acted upon (e.g., a nail).**

We consider *visual affordances*, meaning that we do not model object labels or categories; instead, we compute a set of visual features that represent geometrical properties (e.g., convexity, roundness), which allows to generalize previously-acquired knowledge to new objects. We describe an experiment in which a simulated humanoid robot learns an affordance model by autonomously exploring different actions with the objects present in a playground scenario. We report results showing that the robot is able to i) learn meaningful relationships between actions, tools, other objects and effects, and to ii) exploit the acquired knowledge to make predictions and take optimal decisions.

I. Introduction and Related Work

Humans solve complex tasks on a routine basis, by choosing, amongst a vast repertoire, the most proper actions to apply onto objects in order to obtain certain effects. According to developmental psychology [1], the ability to predict the functional behavior of objects and their interaction with the body, simulating and evaluating the possible outcomes of actions before they are actually executed, is one of the purest signs of cognition, and it is acquired incrementally during development through the interaction with the environment. Neuroscientific evidence [2] supports the idea that, in the brain, these predictions happen during action planning through the activation of sensorimotor structures that couple sensory and motor signals. To reproduce such intelligent behavior in robots is an important, hard and ambitious task.

One possible way to tackle this problem is to resort to the concept of *affordances*, introduced by Gibson in his seminal work [3]. He defines affordances as action possibilities available in the environment to an individual, thus depending on its action capabilities.

From the perspective of robotics, affordances are powerful since they capture the essential world and object properties, in terms of the actions that a robot is able to perform. They can be used to predict the effects of an action, or to plan the actions to achieve a specific goal; by extending the concept further, they can facilitate action recognition and can be exploited for robot imitation [4].

Many computational models have been proposed in the literature in order to equip robots with the ability to learn such affordances (sometimes more generally referred as *sensorimotor representations*) and use them for prediction and planning.

The early work of Fitzpatrick et al. [5] focuses on learning object affordances within a developmental framework, putting forward the idea that a robot can learn about what it can do with an object only by acting on it and observing the effects. More specifically, the robot learns about the rollability affordance of objects, by applying its available actions on the objects several times and observing the resulting changes in the environment.

The concept of affordances and its implications in robotics are discussed by Şahin et al. [6], who propose a formalism to use affordances at different levels of robot control; they apply one part of their formalism for the learning and perception of traversability affordances on a mobile robot equipped with range sensing ability [7].

In the framework presented by Montesano et al. [8], objects affordances are modeled with a Bayesian Network [9], a general probabilistic representation of dependencies between actions, objects and effects; they also describe how a robot can learn such a model from motor experience and use it for prediction, planning and imitation. In order to allow the system to generalize the acquired knowledge across similar objects, they do not consider object labels, but instead they represent objects in terms of a basic set of perceived visual features; for example, the robot learns that spherical objects roll faster than cubic ones when pushed. Since learning is based on a probabilistic model, the approach is able to deal with uncertainty, redundancy and irrelevant information.

The concept of affordances has also been formalized under the name of object-action complexes (OACs, [10]); from a functional perspective in robotics, the terms affordances and object-action complexes point to the same general concept.

All these approaches consider actions that are directly applied to an object. However, many daily actions involve

[1] A. Gonçalves, G. Saponaro, L. Jamone and A. Bernardino are with the Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal. {agoncalves, gsaponaro, ljamone, alex}@isr.ist.utl.pt

This work was partially supported by the POETICON++ project from the European FP7 program (grant agreement no. 288382). G. Saponaro is supported by a doctoral grant (SFRH/BD/61910/2009) from the Portuguese Government – Fundação para a Ciência e a Tecnologia.

the use of *tools*. Humans develop the ability to use tools in a meaningful way following a long and complex process that begins at birth. Guerin et al. [11] provide a discussion about the mechanisms underlying the development of tool use in human newborns, from the initial sensorimotor learning to the emergence of planning capabilities. In this paper, we look at how a robot can learn that certain visual features of a tool make it appropriate to perform a specific action, and to obtain desired effects on the environment. The problem has been investigated within the robotics community, with a few notable examples.

Stoytchev [12] investigates the learning of tool affordances as tool–behavior pairs that provide a desired effect. The learned representation is said to be grounded in the behavioral repertoire of the robot, which knows what it can do with an object using each behavior. However, what is learned are the affordances of specific tools (i.e., considered as individual entities), and no association between the distinctive features of a tool and its affordances is made. Therefore, the generalization capabilities of the system with respect to novel tools are limited.

A recent work by Tikhanoff et al. [13] focuses on learning a specific affordance (i.e., pulling) during tool use. The robot learns what is the best position relative to the object in which the tool-effector must be placed to allow a successful drawing action (bringing the object closer to the robot), as a function of the tool dimension and the shape of the tool-effector. Although useful for robot operations, this knowledge is specific for the tool that is experienced, and cannot be easily generalized to unseen tools.

An interesting approach has been proposed by Jain et al. [14], in which a Bayesian Network is used to model tool affordances as probabilistic dependencies between actions, tools and effects. To address the problem of predicting the effects of unknown tools, they propose a novel concept of tool representation based on the functional features of the tool, arguing that those features can remain distinctive and invariant across different tools used for performing similar tasks. However, it is not clear how those features are computed or estimated, if they can be directly obtained through robot vision and if they can be applied to different classes of tools.

Moreover, it is worth noting that both in [12] and [14] the properties of the affected objects are not considered; only the general affordances of tools are learned, regardless of the objects that the tools act upon.

Our proposed approach builds upon the probabilistic models of [8] and [14]. We model affordances with Bayesian Networks, and we do not consider objects labels, but instead we reason upon the perceived visual features. A distinctive characteristic of our approach is that we do not model just the general affordances of individual objects, but also the affordances of tools and affected objects together, hence modeling the interaction between tools and objects based on their physical (geometrical) properties.

The rest of the paper is organized as follows. In Section II we illustrate our affordance model in detail, describing the visual features that we use to encode objects and effects (Section II-A), and how the model parameters can be learned by the robot through autonomous exploration (Section II-B). In Section III we provide experimental results that support the effectiveness of our approach, showing some examples of how the acquired affordance knowledge can be exploited to instruct the robot behaviors. Finally, in Section IV we report our conclusions and sketch future work directions.

II. A PROBABILISTIC MODEL OF TOOLS AND OBJECTS AFFORDANCES

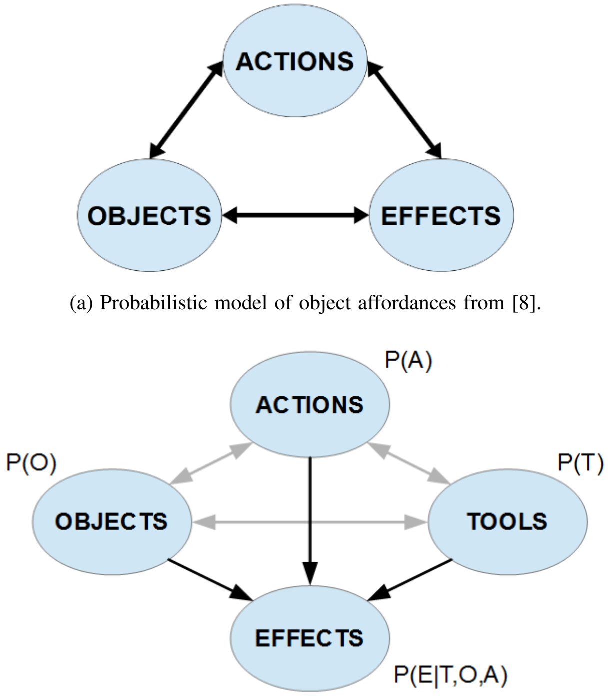
The model of affordances proposed in [8] considers actions that are directly applied to an object: a conceptual schema is depicted in Fig. 1a. However, many actions involve the use of tools. In a sense, we could say that all actions are performed exploiting tools, if we assume that hands and feet can also be viewed as tools. The extension we propose is to include the notion of tools into the probabilistic model of affordances, as depicted in Fig. 1b. The grey arrows identify all the possible connections (the most general architecture), while the black ones are those that we selected for our implementation, in which we consider only the causal relationships from actions/tools/objects to effects.

This model considers the notion of (affected) objects explicitly, unlike [14], therefore it can be used to infer i) affordances of affected objects, ii) affordances of tools, and iii) affordances of the interaction between tools and objects. In our framework, there is not an explicit “toolness” property, neither specified a priori nor learned. All objects are represented by a set of physical properties, so that any object can potentially be used by a robot actively (as a “tool”) or passively (as an “affected object”).

A. Visual descriptors

A great deal of information about the shape of an object can be extracted from its visible silhouette. For this, we need to be able to segment the object from the background, that is, the ability to partition a visual scene into multiple distinct segments. This might be a difficult task, especially if we deal with unknown objects in cluttered scenes. As we are not addressing this general problem in the context of this work, we consider a simple playground environment, consisting of a table with colored objects on top; in this case we can apply simple color-based segmentation to retrieve connected components of pixels in 2D, that we call “blobs”. We assume that each blob corresponds to an object, which in turn can potentially be used as an active tool or as a passive, affected object.

From each segmented blob, we extract the visual features reported in Table I, which we call shape descriptors: an example is shown in Fig. 2. They are computed in 2D as relationships between: blob contour perimeter, blob area, external contour perimeter (polygonal approximation), convex hull, approximating ellipse, minimum-enclosing circle and minimum-enclosing rectangle [15]. All objects can potentially act as active “tools” or passive “affected objects”. For instance, given a visual context with some blobs (e.g., two



(a) Probabilistic model of object affordances from [8].

(b) Extension proposed in this paper, which models the notion of active objects ("tools") and passive ones (affected "objects"). Grey arrows represent the connections of the most general architecture, black ones are those we selected for our implementation.

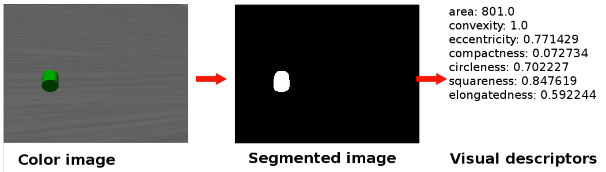
Fig. 1: Conceptual schemas of probabilistic object affordances.

balls and a baseball bat), in general the robot can try out all possible combinations of tool usage with the available items: move ball #1 using the bat, move ball #2 with ball #1, move the bat with a ball, etc. Some of these combinations will produce physical displacement of the affected object, others will not—in other words, some objects *afford* an action onto other objects (we call them "tools" and "affected objects", respectively), and a robot can learn this from self-exploration (Section II-B). For simplicity, in this work we consider as tools two items which have a high elongatedness value (so that their handle can be grasped by the robot), and as affected objects the remaining items of Fig. 5.

The idea is that the set of descriptors captures physical (geometrical) properties of objects and tools that affect the way in which they interact with each other, and the effects that are generated due to specific actions; in other words, the descriptors capture the affordances of objects and tools. The robot can learn these visual affordances by exploring actions on affected objects using tools (while measuring the visual descriptors of the affected object and the tool-effector), and by measuring the visual effects on the affected objects. For example, the robot can learn that objects with a high circleness value (sphere-like objects), when pushed laterally by a tapping action, are likely to roll on the table for longer distances compared to objects with high squareness (cuboids) and that high tool-effector eccentricity (e.g., a rake) is required to draw objects closer.

TABLE I: Shape descriptors.

Descriptor	Definition
Area	Number of pixels
Convexity	Ratio between convex hull perimeter and object perimeter
Eccentricity	Ratio between minor and major axes of best-fit ellipse
Compactness	Ratio between object area and squared external contour perimeter
Circleness	Ratio between object area and area of minimum-enclosing circle
Squareness	Ratio between object area and area of minimum-enclosing rectangle
Elongatedness	Linear combination of *eccentricity* and *compactness*



```
area: 801.0
convexity: 1.0
eccentricity: 0.771429
compactness: 0.072734
circleness: 0.702227
squareness: 0.847619
elongatedness: 0.592244
```

Fig. 2: Example of visual descriptors extracted from a segmented blob.

B. Autonomous robot exploration

From a philosophical point of view, one might argue whether affordances are properties either of the world (cups afford to be grasped because they typically have a handle), of the agent (cups afford to be grasped because humans are typically able to grasp them), or of both of them. However, a great deal of affordance knowledge results from the ecological exploratory interaction between the agent (human or robot) and the environment, thus depending both on the physical properties of the objects and on the motor and perceptual capabilities of the agent.

This learning process requires that the cognitive system knows how to perform a number of actions and has developed some perceptual capabilities: then, affordances can be learned through *self-exploration* (i.e., an exploration of the possibilities of the self in the environment). In practical terms, this means to try out the available actions on the available objects, using the available tools, monitoring and recording the effects: as we focus on visual affordances, the recorded data consists of visual geometric features of both passive objects and active tools, as well as visually

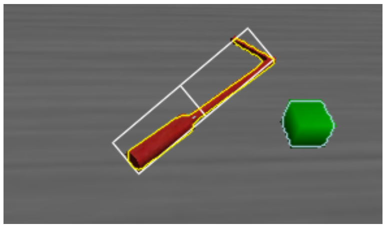


Fig. 3: The item on the left is a possible tool, its blob is divided in two parts (bottom and top); the visual descriptors can be computed on both parts, handle and effector. The item on the right is a possible affected object.

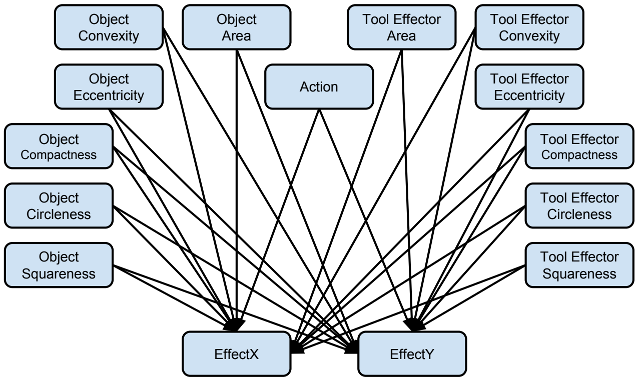


Fig. 4: The detailed structure of the Bayesian Network model for objects and tools affordances used in the experiments.

measurable effects caused by actions on objects. We consider the Bayesian Network structure depicted in Fig. 4; note that this is the most general structure in which all the objects, tools and actions nodes are connected to the effects.

The nodes of Fig. 4 do not encode object labels or categories, but the features that we compute on them. As mentioned in Section II-A, tools are divided into the handle and the effector, and the features of the effector part are the only ones used in the context of this work. Object and tool-effector properties are described by a vector containing the visual descriptors defined in Table I. The two effect nodes in the figure refer to object motion in the horizontal and vertical directions in the image space, respectively.

The nodes in the Bayesian Network can be seen as the perceptual and motor channels available in our system. Therefore, the number of nodes will remain the same if the number of tools and objects increases. Only if more perceptual or motor abilities are added to the system, will the network grow.

III. Experimental Results

We report the results we obtained in learning the visual affordances of objects and tools, and we provide some examples that show how this learned basis of knowledge can be used to support the behavior of a robot in its environment.

A. The robotic platform

The iCub [16] is an open-source humanoid robot for research in embodied cognition, developed in the context of the EU project RobotCub (2004-2010) and adopted by more than 20 laboratories worldwide. It has 53 motors that move the eyes, neck, arms and hands, waist, and legs. It is equipped with stereo vision, proprioception, vestibular system, force and tactile sensing.

In this work, we adopt the iCub Simulator [17], a realistic software that uses ODE (Open Dynamic Engine) for simulating rigid body dynamics and collision detection algorithms to model physical interaction with objects. YARP [18] and iCub software libraries are employed to provide the simulated robot with motor control capabilities to perform several actions using various tools [13]. We implemented the software modules that compute the visual descriptions of tools and affected objects (Section II-A), and those that coordinate autonomous exploration (Section III-B). Our software is publicly available from the iCub repository (`http://www.icub.org`), and it runs both on the real iCub and on the simulator.

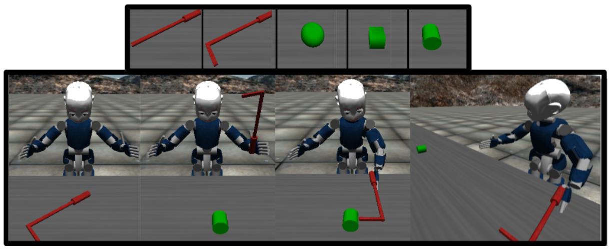


Fig. 5: Exploration using the iCub simulator. Top image: the tools and objects explored. Bottom image: the exploration sequence that is repeated multiple times for each tool-object-action combination.

B. Learning the affordances through exploration

To show the effectiveness of our proposed method for learning tools and objects affordances, we consider an exploration scenario in which the robot interacts with three different objects (Sphere, Cube, Cylinder – $\mathrm{Sp}, \mathrm{Cu}, \mathrm{Cy}$) located on a table, using two different tools (Stick, Rake – St, Ra), applying four possible actions (Tap from Left, Tap from Right, Push, Draw – $\mathrm{TL}, \mathrm{TR}, \mathrm{Pu}, \mathrm{Dr}$), and measuring two effects (horizontal and vertical motion of an object on the table – $\mathrm{EffectX}, \mathrm{EffectY}$). One single exploration step is defined as an action over an affected object using a tool, as in Fig. 5; for each step, the visible effects of the action are recorded. The robot tracks affected objects by moving its eyes, head and torso, in order to compute the object displacement on the table resulting from the action; the four actions involve motion of the robot torso and arms.

The structure of the Bayesian network is depicted in Fig. 4. The information inside each of the perceptual nodes is clustered, depending on the distribution of the data. For affected object and tool-effector features, we consider three discrete levels: low (L), medium (M) and high (H). Given the items that we defined in the simulator (without noise), we typically obtain the following sets of features:

- Sphere (Sp) = [H,H,H,H,H,L];
- Cube (Cu) = [L,H,L,L,L,H];
- Cylinder (Cy) = [M,H,M,M,M,M].
- Stick (St) = [L,H,L,L,L,L];
- Rake (Ra) = [H,H,M,M,M,H].

As for effects, we consider 5 discrete levels: Very Positive (VP), Low Positive (LP), No Movement (NM), Low Negative (LN) and Very Negative (VN). In horizontal movements, positive means to the right and negative to the left. For vertical movements, positive means closer to the robot (going down in image space) and negative means farther from the robot (up in image space).

TABLE II: Conditional probability of the horizontal effect given the squareness property of the affected object: $P(\text{EffectX}|\text{Object Squareness})$.

	High Right	Low Right	No Movement	Low Left	High Left
Low	0.292	0.147	0.275	0.129	0.155
Medium	0.301	0	0.353	0.153	0.191
High	0	0.034	0.879	0.086	0

TABLE III: Conditional probability of the horizontal effect given the action: $P(\text{EffectX}|\text{Action})$.

	High Right	Low Right	No Movement	Low Left	High Left
Tap Left	0	0	0.264	0.35	0.385
Tap Right	0.534	0.12	0.345	0	0
Draw	0.092	0	0.671	0.142	0.093
Push	0	0.024	0.942	0.032	0

At the current stage, actions are not parameterized: they are always executed in the same way, with the same arm. Therefore, the Action node simply contains the IDs of the four actions. Adding additional parameters (e.g., movement speed, approaching direction) is a possible direction for future extensions of the model.

We collected data from 144 exploration steps in simulation. From these measurements we learn the conditional probability distribution tables (CPDs) that describe the relationships between parent and children nodes contained in the links of the Bayesian network, $P(\text{child}|\text{parent})$.

Tables II and III show some of the CPDs that result from the exploration. There, we can observe that "high object squareness" produces higher probability of No Movement, and low squareness higher probability of any kind of movement. Another case is that of $P(\text{EffectX}|\text{Action})$, which yields higher probabilities of movement in the direction of the actions. This allows to qualitatively assess the validity of the methodology, proving that the robot has learned meaningful relationships between the perceived visual features of tools and affected objects in the environment. In Table III, the Tap Left/Right action probabilities have asymmetric values across Left/Right movement effects because in this work we are using the same robot arm throughout all the experiments.

C. *Probabilistic inferences for prediction and planning*

After learning, the Bayesian Network can be used for prediction and planning.

Table IV presents some illustrative *inferences* (queries to the Bayesian Network) that show how the information encoded in the network can be used to support the robot behaviors. The first two lines of the table predict the effect (in this case, the vertical motion) of a draw action applied to a sphere and a cylinder, using either a stick or a rake: $P(\text{EffectY}|T = [\text{St}, \text{Ra}], O = [\text{Sp}, \text{Cy}], A = \text{Dr})$. For the sphere, we observe that a stick produces a high entropy results, whereas the rake has 60% probability of drawing the sphere closer. For the cylinder, the result is that the rake can reliably help bring it closer to the robot, whereas the stick has a contrary effect to the one desired. It is worth noting that in Table IV, despite the fact that we are displaying the name of tools and objects, the robot does not know their labels. It only has access to a set of visual shape descriptors which, in this case, we can map to a specific object for human interpretation: however, they could come from a different, novel object that has similar geometric properties. It is via the the object shape properties affording specific actions that the system makes its predictions, not via object categories. This allows *generalization* of the method to unknown objects.

TABLE IV: Query table for the vertical effect of a draw action with a stick and a rake applied to a sphere and a cylinder: $P(\text{EffectY}|T, O, A = \text{Dr})$.

	High Closer	Low Closer	No Movement	Low Farther	High Farther
T=Stick, O=Sphere	0.166	0.166	0.5	0.166	0
T=Rake, O=Sphere	0	0.6	0.4	0	0
T=Stick, O=Cylinder	0	0	0	0.5	0.5
T=Stick, O=Cylinder	0	1	0	0	0

Another possible use of a Bayesian Network model of affordances is for *planning*. For instance, given a set of tool-like objects available (possibly unknown) and a desired effect (e.g., bring closer) in an object (e.g., a cylinder), the robot can reason about which of the available tools and action on its repertoire are better to achieve the desired effect. The query is formulated as:

$$(T^*, A^*) = \arg\max_{T,A} P(T, A|\text{EffectY} = \text{LP}, O = \text{Cy}). \quad (1)$$

Using the learned data and having the tools Rake and Stick available, the above query results in the choice of a Rake and action Draw:

$$P(T = \text{Ra}, A = \text{Dr}|\text{EffectY} = \text{LP}, O = \text{Cy}) = 0.789. \quad (2)$$

The estimated probability of actually obtaining the desired effect (EffectY = LP) is 0.789. More precisely, the query in Eq. 1 produces $A = \text{Dr}$ (draw action) and $T = [H, H, M, M, M, H]$ (the values that the descriptors of the tool-effector should have) as best result (i.e., the result with the highest probability). If the robot is looking at a set of tools available in the environment it can compute the visual features on them, and choose the tool whose features are closer to the ones provided by the query. Another possibility is to directly use the computed features within a query, and ask to the Bayesian Network what is the probability of obtaining the desired effect with each of the available tools; the probabilities can be then ranked, and the best tool (among the available ones) chosen.

Given the experience our simulated robot made over different objects, the results of this query are the same if the object is a sphere (the selected tool and action are still

Rake and Draw), but the estimated probability of obtaining the desired effect is lower:

$$P(T = \text{Ra}, A = \text{Dr} | \text{EffectY} = \text{LP}, O = \text{Sp}) = 0.716. \quad (3)$$

This is a consequence of the fact that spherical objects can rotate more in unexpected directions with respect to cylindrical objects, due to their shape. The same query can be performed also if no information about the object is available: in this case, we still get the same result, but with a lower estimated probability:

$$P(T = \text{Ra}, A = \text{Dr} | \text{EffectY} = \text{LP}) = 0.565. \quad (4)$$

This reflects the fact that during exploration some objects could not be brought closer even using the Rake tool and the Draw action (e.g., a cube, which typically does not move much).

A different query can show that, for example, in order to displace an object (e.g., a cylinder, $O = \text{Cy}$) far to the right (desired effect EffectX $=$ VP), the Tap Right action is the best one, but both a Rake-like and a Stick-like tool can be equally suited. Indeed, if we query

$$(T^*, A^*) = \underset{T,A}{\arg\max}\, P(T, A | \text{EffectX} = \text{VP}, O = \text{Cy}), \quad (5)$$

the two highest probabilities that we obtain are:

$$P(T = \text{St}, A = \text{TR} | \text{EffectX} = \text{VP}, O = \text{Cy}) = 0.371; \quad (6)$$
$$P(T = \text{Ra}, A = \text{TR} | \text{EffectX} = \text{VP}, O = \text{Cy}) = 0.261. \quad (7)$$

Interestingly, all other non-zero probabilities that we obtain also suggest the Tap Right action and tool descriptors that do not correspond to the explored Rake or Stick ($\text{St} = [L, H, L, L, L, L], \text{Ra} = [H, H, M, M, M, H]$), but instead are slight variations of them (namely $[L, M, L, L, L, L]$, $[H, M, M, M, M, H]$, $[L, H, L, L, L, M]$). Any other action among the available ones produces zero probability of displacing the object to the right.

IV. Conclusions and Future Work

We propose a novel probabilistic framework of visual affordances that considers tools and affected objects in manipulation scenarios. Such a model can be learned by a robot through autonomous exploration of the environment, and it can be used to support its intelligent behaviors in many ways, including: i) making predictions about the effects gererated by an action, and ii) choosing the best action and tool to obtain a desired effect on an object. We look at the visual features of tools and affected objects, not at their identities or semantic labels: in this way we can generalize the acquired knowledge to new, unknown tools and objects. Compared to previous works, we consider complex actions in which tools are used to act upon other objects, and we keep the notion of both tools and objects in the model, therefore being able to learn not just the general affordances of either tools or objects, but also the affordances resulting from their joint interaction.

We report experimental results in which a humanoid robot learns the model through autonomous exploration, and we offer several examples on how the acquired knowledge can be used for prediction and planning. We are currently conducting experiments on the real iCub robot. We plan to increase the number of experienced objects, so that we collect data that cover more densely the space of their visual features; moreover, we are interested in scaling up the complexity of the action representation by introducing different action parameters that can vary during the exploration, like the speed of the motion or the trajectory profile.

References

[1] C. von Hofsten, "An action perspective on motor development," *Trends in Cognitive Sciences*, vol. 8, pp. 266–272, 2004.

[2] V. Gallese, L. Fadiga, L. Fogassi, and G. Rizzolatti, "Action recognition in the premotor cortex," *Brain*, vol. 119, pp. 593–609, 1996.

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

[4] M. Lopes, F. S. Melo, and L. Montesano, "Affordance-based imitation learning in robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2007.

[5] P. Fitzpatrick and G. Metta, "Grounding vision through experimental manipulation," *Phil. Trans. R. Soc. A: Mathematical, Physical and Engineering Sciences*, pp. 2165–2185, 2003.

[6] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To Afford or Not To Afford: A New Formalization of Affordances Toward Affordance-Based Robot Control," *Adaptive Behavior*, vol. 15(4), pp. 447–472, 2007.

[7] E. Uğur and E. Şahin, "Traversability: A Case Study for Learning and Perceiving Affordances in Robots," *Adaptive Behavior*, vol. 18(3-4), pp. 258–284, 2010.

[8] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor, "Learning Object Affordances: From Sensory Motor Coordination to Imitation," *IEEE Transactions on Robotics*, vol. 24(1), pp. 15–26, 2008.

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

[10] N. Krüger, C. Geib, J. Piater, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrčen, A. Agostini, and R. Dillmann, "Object–Action Complexes: Grounded abstractions of sensory–motor processes," *Robotics and Autonomous Systems*, vol. 59(10), pp. 740–757, 2011.

[11] F. Guerin, N. Krüger, and D. Kraft, "A Survey of the Ontogeny of Tool Use: From Sensorimotor Experience to Planning," *IEEE Transactions on Autonomous Mental Development*, vol. 5(1), pp. 18–45, 2013.

[12] A. Stoytchev, "Learning the Affordances of Tools using a Behavior-Grounded Approach," *Affordance-Based Robot Control, Springer Lecture Notes in Artificial Intelligence (LNAI)*, pp. 140–158, 2008.

[13] V. Tikhanoff, U. Pattacini, L. Natale, and G. Metta, "Exploring affordances and tool use on the iCub," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2013.

[14] R. Jain and T. Inamura, "Bayesian learning of tool affordances based on generalization of functional feature to estimate effects of unseen tools," *Artificial Life and Robotics*, vol. 18(1-2), pp. 95–103, 2013.

[15] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, no. 1, pp. 1–19, Jan. 2004.

[16] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. von Hofsten, K. Rosander, M. Lopes, and J. Santos-Victor, "The iCub humanoid robot: An open-systems platform for research in cognitive development," *Neural Networks*, vol. 23, pp. 1125–1134, 2010.

[17] V. Tikhanoff, P. Fitzpatrick, G. Metta, L. Natale, F. Nori, and A. Cangelosi, "An Open Source Simulator for Cognitive Robotics Research: The Prototype of the iCub Humanoid Robot Simulator," in *Workshop on Performance Metrics for Intelligent Systems*, National Institute of Standards and Technology, Washington DC, August 19-21 2008.

[18] G. Metta, P. Fitzpatrick, and L. Natale, "YARP: Yet Another Robot Platform," *International Journal on Advanced Robotics Systems*, March 2006, special Issue on Software Development and Integration in Robotics.