IEEE International Conference on Development and Learning and on Epigenetic Robotics (ICDL-EpiRob 2014)
Genoa, Italy, 13-16 October 2014

Learning Intermediate Object Affordances: Towards the Development of a Tool Concept

Afonso Gonçalves, João Abrantes, Giovanni Saponaro, Lorenzo Jamone, Alexandre Bernardino
Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal
{agoncalves, jabrantes, gsaponaro, ljamone, alex}@isr.ist.utl.pt

***Abstract*—Inspired by the extraordinary ability of young infants to learn how to grasp and manipulate objects, many works in robotics have proposed developmental approaches to allow robots to learn the effects of their own motor actions on objects, i.e., the objects affordances. While holding an object, infants also promote its contact with other objects, resulting in object–object interactions that may afford effects not possible otherwise. Depending on the characteristics of both the held object (intermediate) and the acted object (primary), systematic outcomes may occur, leading to the emergence of a primitive concept of tool. In this paper we describe experiments with a humanoid robot exploring object–object interactions in a playground scenario and learning a probabilistic causal model of the effects of actions as functions of the characteristics of both objects. The model directly links the objects' 2D shape visual cues to the effects of actions. Because no object recognition skills are required, generalization to novel objects is possible by exploiting the correlations between the shape descriptors. We show experiments where an affordance model is learned in a simulated environment, and is then used on the real robotic platform, showing generalization abilities in effect prediction. We argue that, despite the fact that during exploration no concept of tool is given to the system, this very concept may emerge from the knowledge that intermediate objects lead to significant effects when acting on other objects.**

I. Introduction

Many important human behaviors require making objects contact with each other. A fundamental cognitive ability to master such skill is to understand the relationships between the physical properties of the objects's surfaces that enter into contact. For instance, to pile objects we must put into contact their flat surfaces to assure stability; to bring closer objects out of reach we pull them with elongated objects; to fit objects together we match concave parts on one object to corresponding convex parts on the other. Infants and toddlers achieve this ability throughout a developmental process spanning several stages. At 6 months of age infants already manipulate objects in a differentiated manner depending on object's properties [1]. During the second half year infants become sensitive not just to objects and surfaces alone, but to the affordances entailed by the relation between the two [2], [3]. At the second year children begin using objects for increasingly complex problem solving and tool use [4]–[6]. While a number of researchers have suggested that tool use requires a cognitive leap beyond information that is directly perceived, thus requiring the ability to engage in novel forms of symbolic or relational thinking [7], a new wave of research proposes an alternative view in which tool use is seen as an extension of the perception-action coupling that infants show during the first year of life; therefore, the concept of tool may emerge from the detection of possible affordances between objects or object parts, based on information that is directly perceptible [5]. From this perspective, the trial and error attempts that precede successful tool use can be seen as exploratory behaviors, providing opportunities for affordance learning.

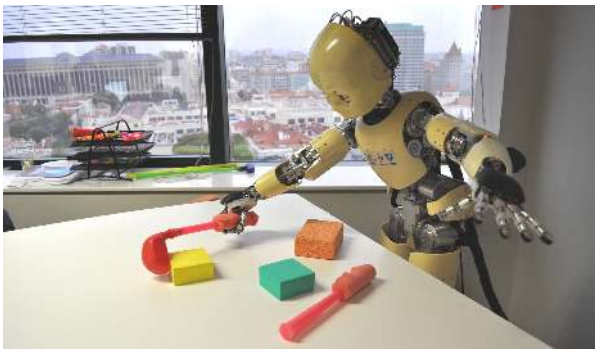


Figure 1. The iCub humanoid playing with objects.

In this paper we investigate how the iCub [8] humanoid robot can learn the affordances between a hand held object (intermediate) and an acted object (primary), in analogy to developmental processes occurring in the second half year of young infants. We adopt a probabilistic model of affordances relating the shape properties of the intermediate and primary objects with the effects of robot's motor actions, measured as relative displacements of the primary object. This model is learned by performing numerous experiments on a set of objects displaced on a table (Fig. 1). One of the objects is selected for grasping (intermediate object) and the other object is selected to be acted on, by making the free surface of the held object impact on it according to a set of possible directions. Objects and actions are randomly selected during the learning process and a causal model of the occurrences is obtained in the form of a Bayesian Network [9].

Object's surfaces are characterized by pre-categorical shape descriptors. We do not perform any object recognition since it is not the class of an object but its physical characteristics that ultimately determine the affordance. 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. It

is on the basis of human intelligent behavior and creativity the ability to infer function of novel objects from shape similarities with known ones.

For practical purposes, the model (Bayesian Network) is learned on the iCub simulator [10], but the used perceptual features are shown to generalize well to the real robot. We test several ways to learn the structure of the model, and compare to the baseline in [11]. After the model has been learned, we demonstrate its applicability and generalization with the real robot platform, in a prediction task.

The rest of the paper is organized as follows. After reviewing the state of the art (Sec. II), in Sec. III we present our computational model of affordances, proposing different possible structures. Then, in Sec. IV we describe the experimental setup and in Sec. V we report the results of our experiments, that show the estimation and generalization properties of the different structures. Finally, in Sec. VI we draw our conclusions.

II. Related Work

In his highly influential work [12], Gibson defines affordances as action possibilities available in the environment to an individual, therefore depending on its motor abilities. The concept of object affordances can be very powerful in robotics since it allows to capture the most informative object properties in terms of the actions that a robot is able to perform.

A number of computational models have been investigated in the robotics literature to learn object affordances and use them for prediction [13], imitation [13], [14], planning [13], [15], tool use [16]–[19], and language grounding [20], [21].

The early work of Fitzpatrick et al. [22] proposes an ecological approach to affordance learning, putting forward the idea that a robot can learn affordances just by acting on objects and observing the effects: more specifically, they describe experiments in which a robot learns about the rollability affordance of objects, by tapping them several times and observing the resulting motion.

In the framework presented by Montesano et al. [13], 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. To achieve better generalization, 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, [23]); from a functional perspective in robotics, the terms affordances and object–action complexes point to the same general concept.

While most works consider actions that are directly applied to a single object, a few of them deal with multi-objects scenarios, either in terms of tool use [17], [18] or pairwise object interaction [15].

Sinapov and Stoytchev [16], [17] investigate 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. The generalization capabilities of the system are limited to dealing with smaller and larger versions of known tools.

An interesting approach has been proposed by Jain et al. [18], 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 in [16]–[18] the properties of the acted objects are not considered; only the general affordances of tools are learned, regardless of the objects that the tools act upon.

The recent work of Moldovan et al. [15] considers a multi-object scenario in which the relational affordances between objects pairs are exploited to plan a sequence of actions to achieve a desired goal, using probabilistic reasoning. The pairwise interactions are described in terms of the objects relative distance, orientation and contact; however, they do not investigate how these interactions are affected by different geometrical properties of the objects.

III. Probabilistic Model of Object Affordances

We follow the framework of [13], where the relationships between an acted object, the applied action and the observed effect are encoded in a causal probabilistic model, a Bayesian Network (BN)—whose expressive power allows the marginalization over any set of variables given any other set of variables. It considers that actions are applied to a single object using the robot hands, whereas we model inter–object affordances, including new variables that represent the intermediate object as an individual entity, as depicted in Fig. 2. The BN of our approach explicitly models both primary (acted) and intermediate (held) objects, thus we can infer i) affordances of primary objects, ii) affordances of intermediate objects, and iii) affordances of the interaction between intermediate and primary objects. For example, our model can be used to predict effects given both objects and the performed action, or choose the best intermediate object

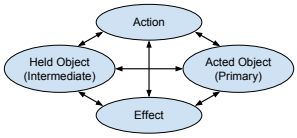
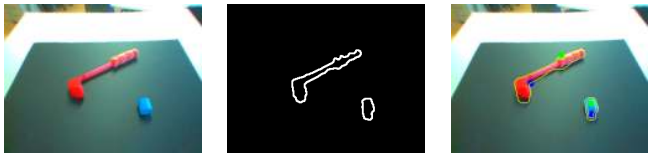


Figure 2. General architecture of affordances, modeled as relations between actions, effects and objects (held and acted).



(a) Color input. (b) Binary segmentation. (c) Color image with overlay contours.

Figure 3. Visual processing pipeline. For each segmented blob, we compute the visual features of Table I.

(tool) to achieve a goal (effect to be produced on a primary object).

A. Visual Descriptors

We use the shape characteristics of objects in terms of descriptors of their 2D silhouette, segmented from the background; the objects that we employ are almost flat and they are put on a table in favorable perspectives. In future work we aim at introducing 3D features; still, we believe that a great deal of information about the shape of an object can be extracted from its visible silhouette. As we are not addressing the general segmentation problem, we consider a simple playground environment, consisting of a table with colored objects on top. Thus, we 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 the intermediate object or the primary object. Furthermore, we compute not only the descriptors of *whole* blobs, but also the ones of blob *halves*: we divide each blob in two parts along its main axis; this division is helpful to capture affordances of intermediate objects, for which only the tip part (upper half, or not grasped part) is relevant in interactions. For held objects, the variables encode tip part visual descriptors; for acted objects, they encode whole blob descriptors.

Each segmented blob is described by the visual features reported in Table I. An example is shown in Fig. 3. These visual descriptors include information about blob contour perimeter, blob area, external contour perimeter (polygonal approximation), convex hull, approximating ellipse, minimum-enclosing circle and minimum-enclosing rectangle [24]. These shape primitives are interesting from the point of view of generalization abilities. Because they are very general and do not demand for a categorization of the object, they can be used to assess similarity between objects even if they belong to different classes. For instance a thermometer and a pencil are

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

categorically different but, in terms of physical properties, they are very similar and afford some common actions (e.g., stir the coffee). If fact, it is one of the main characteristics of human intelligence the ability to use the properties of available objects to produce effects that were not obvious, on a first analysis. Contrary to object recognition approaches, which say little if any about the physical properties of an object, our proposed descriptors are very much related to the shape of surfaces and change smoothly across similar objects.

For visual features of held and acted objects, we consider three empirically-defined discrete levels: Low (L), Medium (M) and High (H).

B. Actions and Effects

For the actions used in this work, we consider four directional pushes (left, right, pull closer, push away), performed with the robot end effector extension (tip of the hand held object). The IDs of these four actions are the observed values of the action node in the affordance networks.

As for effects, we further divide them in EffectX and EffectY, and we consider the 2D displacement (along lateral and longitudinal direction, respectively) of the acted object on the table plane, from the time when the robot performs the action, until a fixed duration of a few frames afterwards.

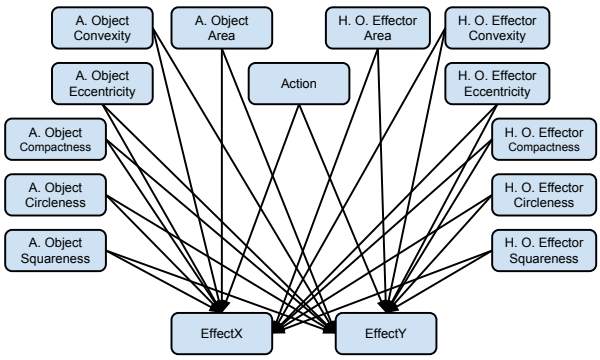
For the effect nodes, we consider five discrete levels: Very Positive (VP), Low Positive (LP), No Movement (NM), Low Negative (LN) and Very Negative (VN). In lateral movement, positive means to the right and negative to the left. For longitudinal movements, positive means closer to the robot (down in image space), negative means farther from the robot (up in image space).

C. Fully Connected Network

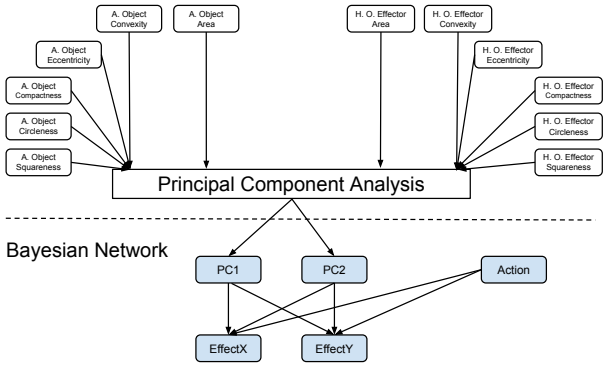
In this article, we will evaluate different Bayesian Networks[1] for their space complexity, speed of training, prediction ability and generalization capability – or lack of.

The baseline structure for our comparisons is a manually defined *fully connected* network, shown in Fig. 4a. This is

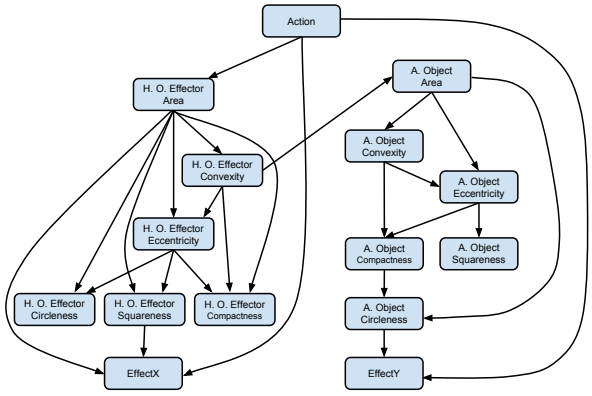
[1]In all the Bayesian Network structures that we discuss, we use discrete variables and Maximum A Posteriori probability estimates to learn the conditional probability distribution (CPD) table parameters.



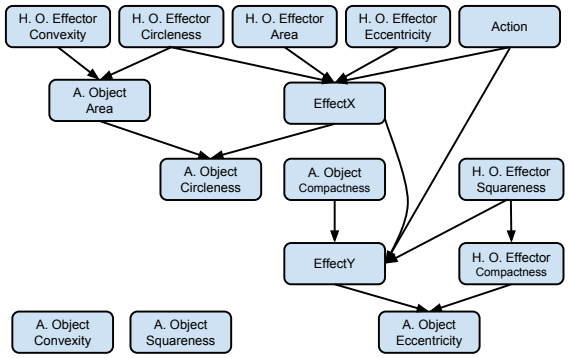
(a) Fully connected network, manually defined.



(b) Reduced network.



(c) K2 Structure Learning network.



(d) BDe Structure Learning network.

Figure 4. Proposed Bayesian Network structures to encode inter–object affordances. Figs. 4b, 4c, 4d were obtained from experimental data (Sec. V-B). A. Object: Acted Object; H. O.: Held Object.

the most general structure, in which all the object and action nodes of the conceptual diagram of Fig. 2 are connected to the effect nodes.

The fully connected network suffers from a number of limitations, further explained in Sec. V-A: low performance, overfitting, large number of parameters. Basically, this network structure suffers from the curse of dimensionality: each effect node has 13 parents, resulting in big CPD tables, which makes the network hard to train and unfit to generalize the trained observations to unseen situations.

D. Reduced Network

To reduce the dimensionality of the feature space, we apply Principal Component Analysis to the features seen on our training data, as shown in the upper part of Fig. 4b. Using 80% of our experimental data for training and 20% for testing (see Sec. V-B), PCA provides 12 principal components; however, we only need 2 principal components to explain almost 100% of the data variance. Therefore, we created two nodes, each corresponding to a principal component, and these, along with the action node, are now the parents of the effect nodes of a *reduced* Bayesian Network, displayed in the lower part of Fig. 4b. The values of these nodes are the coefficients of each eigenvector given the observable features. These coefficients are then discretized, based on the training data, into two values. We also tried to discretize each node into more values, but the performance of the network when predicting effects of unseen data was significantly worse.

E. Structure Learning

In Bayesian Network structure learning, the search space contains all possible structures of directed acyclic graphs (DAGs), given the number of variables in the domain. [9] Because the number of DAGs is super-exponential in the number of nodes, it is unfeasible to enumerate all possible network structures and assign them a score, even for a low number of nodes. This justifies the usage of heuristics to find a (local) maximum in the structure space. We employ two heuristic-based approaches: K2 [25], [26] and BDe (Bayesian Dirichlet likelihood-equivalence, [27]). In both cases, we use 80% of our experimental data for training and 20% for testing (see Sec. V-B).

The K2 algorithm searches for the structure that maximizes the joint probability $p(\text{structure}, \text{data})$, for this it assumes a known ordering on the domain variables and that all possible structures are equally likely. It starts from the lowest-order node and makes its way sequentially to the highest. At each node it first assumes that it has no parents, then it uses a

Table II
COMPLEXITY OF BAYESIAN NETWORKS, COMPUTED AS THE SUM OF THE ELEMENTS IN THE CPDS OF ALL NODES.

Baseline	PCA	Structure Learning BDe	Structure Learning K2
21257680	**168**	1594	535

Figure 5. Exploration sequence on the iCub simulator.

greedy-search method over the K2 score [25] of the lower-order nodes to incrementally add them as its parents. Fig. 4c shows the learned K2 structure.

With BDe, the structure of the networks is maximized by using greedy search and simulated annealing. All the nodes except for EffectX and EffectY were entered as interventional variables, and the resulting network is shown in Fig. 4d.

The measure of complexity in Table II is computed as the number of elements in the largest CPD of a network. Complexity depends only on the discretization and on the network structure, independently of data and learning.

IV. EXPERIMENTAL SETUP

In this section we present the iCub robot and the experimental setup implemented for inter–object affordances exploration.

A. Robotic Platform

The iCub [8] 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 both the iCub Simulator [10] and the real robot. YARP [28] and iCub software libraries are employed to provide the simulated robot with motor control capabilities to perform several actions using various tools [19]. We implemented the software modules that compute the visual descriptions of intermediate and acted objects (Section III-A), and those that coordinate autonomous exploration (Section IV-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.

B. Autonomous Exploration of Affordances

To explore the multitude of possible values for the nodes in the Bayesian Network, data was gathered from 2353 experiments in the iCub simulator and 21 in the iCub robot.

The experiments in the simulator consisted in, for each experimental trial, performing 1 of the 4 directional push actions upon the primary object while holding an intermediate object, where both objects were chosen from a set of 8 possibilities (shown in Figs. 5 and 6). Of these experiments, some were used to learn the proposed Bayesian Network models and some for testing, as further detailed in Sec. V-B.

In the real robot the experiments consisted on performing the "tap to the left" action while holding a straight stick 10 times on a ball and 11 times on a box. These experiments were used to evaluate the generalization abilities of the model going from simulation to the real setup.

V. RESULTS

In this section, we show results of the affordance networks, both in *simulation* and on the *real robot*.

A. Evaluation Scores

For our tests, the score criteria that we employ are the following: gambling score, accuracy, distance.

Gambling score: in this scoring system the robot makes a prediction for each effect given the observation of all the other variables (nodes). To make this prediction the posterior probability, $p(\text{EffectX}|\boldsymbol{O} = \boldsymbol{v})$, is computed, where $\boldsymbol{O}$ are the object and action nodes and $\boldsymbol{v}$ are their values. The predicted effect is then the value of the Effect that maximizes the posterior probability.

If the predicted effect is equal to the real effect then we add 4 points to the score, otherwise we subtract 1 point.

If we had a random machine predicting random effects and comparing against observations, its estimated score would be zero. If we had a perfect predictor, we would obtain the score of four times the number of test data observations.

In Table III, the score is presented as a percentage of the score obtained by the network, divided by the perfect score. With this score, we can easily see how much better the robot performance is, versus a random machine and versus a perfect machine.

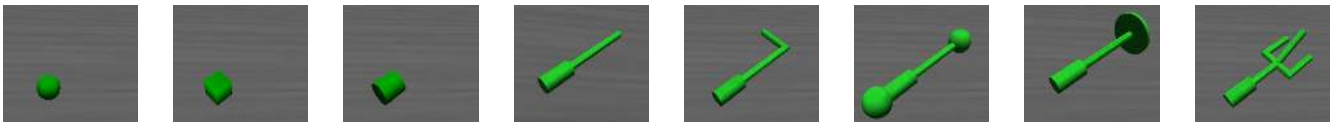
Accuracy: defined as the number of correct predictions over the number of total predictions.

Distance: defined as the absolute difference between prediction and real value. In Table III it is shown as a percentage, relative to the maximum possible distance.

B. Performance Evaluation and Discussion

We evaluate the Bayesian Networks regarding their capability of predicting effects, given two objects' visual descriptors and the action performed with them, with previously unseen test data. To do this we perform two tests, outlining advantages and disadvantages of each of the networks.

The first test consists of randomly *splitting* the data in a training set with 80% of observations, the remaining 20% for testing. The exploration data is relative to the 1663 trials corresponding to the seven objects of Fig. 6a–6g. Results are presented in Table III. The original baseline network is the



(a) Ball. (b) Cube. (c) Cylinder. (d) Stick. (e) L-stick. (f) Bone. (g) Umbrella. (h) Fork.

Figure 6. Objects used in robot simulation, to train the Bayesian Networks. Object 6h is only used in leave-one-out test, see Sec. V-B.

Table III
SCORES WHEN RANDOMLY SELECTING 80% OF OBSERVATIONS AS TRAINING DATA, REMAINING OBSERVATIONS AS TEST DATA. R.P. STANDS FOR RANDOM PREDICTIONS.

	Baseline (13.55% r.p.)	PCA (0% r.p.)	Structure Learning BDe (0% r.p.)	Structure Learning K2 (0% r.p.)
Gambling Sc.	69.88%	75.72%	79.10%	**79.67%**
Accuracy	75.90%	80.57%	83.28%	**83.73%**
Distance	9.11%	6.10%	**5.12%**	**5.12%**

Table IV
LEAVE-ONE-OUT SCORES, TESTING NETWORKS AGAINST AN OBJECT UNSEEN DURING TRAINING. R.P. STANDS FOR RANDOM PREDICTIONS.

	Baseline (57.25% r.p.)	PCA (0% r.p.)	Structure Learning BDe (52.61% r.p.)	Structure Learning K2 (53.04% r.p.)
Gambling Sc.	30.25%	**67.39%**	35.53%	34.91%
Accuracy	44.20%	**73.91%**	48.42%	47.93%
Distance	25.60%	**7.28%**	23.72%	23.97%

one with the lowest performance: due to its huge complexity, this network does not generalize well what it learned. 13.55% of the time, this network made a random prediction because an event where all the instantiated variables were seen with the exact same values observed in the test data was never seen during training. The PCA network yields a good score, because it has the smallest complexity of all the networks considered. However, the two networks obtained with Structure Learning, BDe and K2, provided very similar results and were the networks with the best performance for the test data.

The second test is a *leave-one-out* cross-validation, using the same networks as the previous test, but the unseen object of Fig. 6h as test data (690 samples). Results are shown in Table IV. The PCA network has the best performance: being the least complex network makes it the most capable network for generalization to unseen objects. The performance of the other networks got significantly worse, showing that these networks are too dependent on the training data (overfitting), so their use on the real robot with a changing environment should be accompanied with an online Structure Learning and parameter learning algorithm.

C. Generalization from Simulation to Real Robot

In this experiment, the robot performed the "tap to the left" action while holding a straight stick. It repeated this action 10 times acting on a ball, 11 times acting on a box. From these iterations, we computed the statistics to be used as ground truth, and we compared them to the prediction of the resulting effect, given acted object and intermediate object, by the K2 and PCA network.[2] Results for the query $p(\text{Effect}|\ \text{parents}(\text{Effect}))$, where Effect is EffectX or EffectY, and the ground truths, are shown together in Table V.

We evaluate how well the predictions match the ground truth by computing the match distance [29] between their histogram distributions. Being a cross-bin dissimilarity measure, the match distance is suited to cases where the bin order matters. Our bin order for the effects (VN,LN,NM,LP,VP) places more similar displacements in neighbor bins. The maximum value of the distance, in our case, is $d_{\text{MAX}} = 4$, the distance between histograms $(1, 0, 0, 0, 0)$ and $(0, 0, 0, 0, 1)$. It is a special case of the Earth Mover's Distance, so it can be interpreted as the amount of mass transported between bins times their distance, to transform one histogram to the other.

Both the PCA network and the K2 structure provide acceptable results (average match distances below 10% of d_{MAX}), with K2 being slightly more accurate (about 7% lower match distances), although the K2 structure of Fig. 4c has the peculiarity of the EffectX node being conditionally independent from acted object features. This explains why the K2 EffectX rows of Table V have equal values, regardless of the acted object.

VI. CONCLUSIONS

We presented a novel computational model of multi-object affordances using Bayesian Networks. Our model considers actions performed using an intermediate object over a primary one while *relating the visual features of both* of them to the effects, unlike previous works on affordances, which deal with either primary objects only (no intermediate object), or with intermediate objects yet ignoring the characteristics of the primary one. Different structures of the Bayesian Network, obtained either through structure learning (K2 and BDe algorithms) or dimensionality reduction (PCA), are investigated and compared in terms of complexity, representation capability and generalization, with respect to a baseline fully-connected structure. The results show that both structure learning and dimensionality reduction techniques allow to

[2]In the effect prediction experiment, the baseline network and the BDe network provided random answers (equal probability for all values) because their structure did not represent well the exact combination of observations in the experiment.

Table V
COMPARISON BETWEEN GROUND TRUTH (GT) AND EFFECT PREDICTION BY K2 AND PCA NETWORKS. PCA PROVIDES BETTER MATCHES FOR THE BALL EXPERIMENTS, K2 FOR THE BOX ONES. OVERALL, PCA HAS A MATCH DISTANCE 7.3% HIGHER THAN K2.

	VN			LN			NM			LP			VP			match distance	
	GT	K2	PCA	GT	K2	PCA	GT	K2	PCA	GT	K2	PCA	GT	K2	PCA	K2	PCA
ball EffectX	0	0	0	0.3	0.0233	0.0137	0.5	0.8372	0.7945	0.1	0.1395	0.1644	0.1	0	0.0274	0.4372	**0.3671**
ball EffectY	0	0.01	0	0.1	0	0.0137	0.5	0.2	0.3699	0	0.23	0.3014	0.4	0.56	0.3151	0.65	**0.3872**
box EffectX	0	0	0	0.0909	0.0233	0.0337	0.9091	0.8372	0.7416	0	0.1395	0.2247	0	0	0	**0.2071**	0.2819
box EffectY	0	0	0.0112	0	0.0204	0.0449	0.4545	0.5306	0.7079	0.5455	0.4490	0.1348	0	0	0.1011	**0.1169**	0.4781

reduce the complexity of the model while improving the estimation performance; more specifically, the PCA model is characterized by the lowest complexity and the best performance in generalization to completely new objects (Table II and Table IV), while the K2 model performs slightly better in representing the experienced data (Table III). Moreover, the model learned in simulation can be used to reasonably predict the effects of the actions on the real robot; in this case, the structure obtained with the K2 algorithm shows the best average performance (Table V).

ACKNOWLEDGMENTS

This work was supported by the European Commission under the POETICON++ (FP7-ICT-288382) and LIMOMAN (PIEF-GA-2013-628315) projects, and by the Portuguese Government – Fundação para a Ciência e a Tecnologia (PEst-OE/EEI/LA0009/2013). G. Saponaro was supported by an FCT doctoral grant (SFRH/BD/61910/2009).

REFERENCES

[1] E. W. Bushnell and J. P. Boudreau, "Motor Development and the Mind: The Potential Role of Motor Abilities as a Determinant of Aspects of Perceptual Development," *Child Development*, vol. 64, no. 4, pp. 1005–1021, 1993.

[2] K. S. Bourgeois, A. W. Khawar, S. A. Neal, and J. J. Lockman, "Infant Manual Exploration of Objects, Surfaces, and Their Interrelations," *Infancy*, vol. 8, no. 3, pp. 233–252, 2005.

[3] L. Rat-Fischer, K. O'Regan, and J. Fagard, "The emergence of tool use during the second year of life," *Journal of Experimental Child Psychology*, vol. 113, no. 3, pp. 440–446, 2012.

[4] A. M. Damast, C. S. Tamis-LeMonda, and M. H. Bornstein, "Mother–Child Play: Sequential Interactions and the Relation between Maternal Beliefs and Behaviors," *Child Development*, vol. 67, no. 4, pp. 1752–1766, 1996.

[5] J. J. Lockman, "A Perception–Action Perspective on Tool Use Development," *Child Development*, vol. 71, no. 1, pp. 137–144, 2000.

[6] 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.

[7] E. Bates, "The biology of symbols: Some concluding thoughts," in *The Emergence of Symbols: Cognition and Communication in Infancy*, E. Bates, L. Benigni, I. Bretherton, and L. C. . V. Volterra, Eds. New York: Academic Press, 1979, pp. 315–370.

[8] 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.

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

[10] 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.

[11] A. Gonçalves, G. Saponaro, L. Jamone, and A. Bernardino, "Learning Visual Affordances of Objects and Tools through Autonomous Robot Exploration," in *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2014.

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

[13] 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.

[14] 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.

[15] B. Moldovan, P. Moreno, and M. van Otterlo, "On the use of probabilistic relational affordance models for sequential manipulation tasks in robotics," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013, pp. 1290–1295.

[16] J. Sinapov and A. Stoytchev, "Learning and Generalization of Behavior-Grounded Tool Affordances," in *IEEE International Conference on Development and Learning (ICDL)*, 2007.

[17] 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.

[18] 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.

[19] 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.

[20] G. Salvi, L. Montesano, A. Bernardino, and J. Santos-Victor, "Language Bootstrapping: Learning Word Meanings From Perception-Action Association," *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 3, pp. 660–671, 2012.

[21] O. Yürüten, K. F. Uyanık, Y. Çalışkan, A. K. Bozcuoğlu, E. Şahin, and S. Kalkan, "Learning Adjectives and Nouns from Affordances on the iCub Humanoid Robot," in *From Animals to Animats 12*. Springer, 2012, pp. 330–340.

[22] P. Fitzpatrick and G. Metta, "Grounding Vision Through Experimental Manipulation," *Phil. Trans. R. Soc. A: Mathematical, Physical and Engineering Sciences*, pp. 2165–2185, 2003.

[23] 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.

[24] D. Zhang and G. Lu, "Review of Shape Representation and Description Techniques," *Pattern Recognition*, vol. 37, no. 1, pp. 1–19, Jan. 2004.

[25] G. F. Cooper and E. Herskovitz, "A Bayesian Method for the Induction of Probabilistic Networks from Data," *Machine Learning*, vol. 9, no. 4, pp. 309–347, 1992.

[26] C. Bielza, G. Li, and P. Larrañaga, "Multi-Dimensional Classification with Bayesian Networks," *Int. J. Approx. Reasoning*, vol. 52, pp. 705–727, 2011.

[27] A. Shah and P. Woolf, "Python Environment for Bayesian Learning: Inferring the Structure of Bayesian Networks from Knowledge and Data," *J. Mach. Learn. Res.*, vol. 10, pp. 159–162, Feb. 2009.

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

[29] Y. Rubner, C. Tomasi, and L. J. Guibas, "The Earth Mover's Distance as a Metric for Image Retrieval," *Int. J. Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.