

# Predicting future object states using learned affordances

Emre Ugur<sup>1,2,3</sup>, Erol Şahin<sup>3</sup>, and Erhan Oztop<sup>4,1,2</sup>

<sup>1</sup> NICT, Biological ICT Group, Kyoto, Japan

<sup>2</sup> ATR, Computational Neuroscience Labs., Kyoto, Japan

<sup>3</sup> METU, CENG, Kovan Lab., Ankara, Turkey

<sup>4</sup> JST, ICORP, Computational Brain Project, Saitama, Japan

**Abstract**—The notion of *affordances* that was proposed by J.J. Gibson, refers to the action possibilities offered to the organism by its environment. In a previous formalization, affordances are defined as general relations that pertain to the robot-environment interaction and they are represented as triples which consist of the initial percept of the environment, the behavior applied and the effect produced. In this paper, we focus on the object affordances and propose a developmental method that enables the robot to ground symbolic object-based operators in its own continuous sensory-motor experiences. The method allows the robot to learn the object affordance relations which can be used to predict the change in the percept of the object when a certain behavior is executed.

## I. INTRODUCTION

The notion of *affordances* was proposed by J.J. Gibson, to refer to the action possibilities offered to the organism by its environment[1]. A horizontal and rigid surface affords walkability, a flat surface at a certain height affords sit-tability. This notion emphasizes the complementarity of robot and its environment and claims that affordances are determined by both the properties of the objects and the properties (and capabilities) of the organism. A small cobblestone may afford hide-ability to a mouse, while affording throw-ability (but not hide-ability) to a human.

Recently, in [2], a formalization for affordances was developed as a framework over which affordances can be utilized at different levels of robot control. According to this formalization, affordances are general relations that pertain to the robot-environment interaction and can be represented as a triple which consist of the initial percept of the object, the behavior applied and the effect produced. For instance, the lift-ability affordance is a relation between the properties of an object, the behavioral capabilities of the robot and the type of effect produced by the lift behavior. In this paper, we use this framework to propose a developmental method that enables a robot to learn the symbolic relations that pertain to its interactions with the objects and show that they can be used to predict the next perceptual state of the objects.

## II. RELATED WORK

Learning of object affordances have been studied in [3] where the objects in the environment were differentiated using their colors only, and the real distinctive features of

the objects are not learned. In another work, [4] proposed a general probabilistic model based on Bayesian networks to learn the relationship between actions, objects, and effects through interaction with the environment. In [5] the robot learned tool affordances by building an adaptive hierarchical outcome model for each behavior. These models were later used to detect the functional similarities between the tools and to predict the effect a tool-behavior pair is expected to create. Although all these studies investigated the problem of predicting the future effects based on discovered affordance relations, none of them used learned affordances in planning.

Traditional AI planners work on a symbolic domain, but robot control and perception usually occurs on a much lower level. In order to bridge this gap [6] used self-organizing maps to cluster low-level perception data, and then mapped one perceptual state (cluster) to the next outcome state (cluster). Direct prediction of the outcome state rather than the prediction of the effect have some drawbacks on the flexibility of the system. While there is no need to represent the non-changing features (for example shape) when predicting the effect, such features must also be taken into account in a system predicting the outcome. In another study, [7] object-action complexes are used as generalized relations between objects, actions, the current and the next states of the world. However, the effects and preconditions are not discovered from scratch in a bottom-up manner, instead they are pre-defined as high-level domain properties. In [8], the robot learned *liftability* of an object based on the parameters of a pre-specified sequence of actions. But since the plan in this case was fixed, the learned knowledge could not be extended to a generic sequence of actions. In [9], the robot learned goal-free action structures, in relation with the objects and the effects of these actions on the objects. Object concepts and action structures were then used in generating a sequence of actions in a goal-directed manner. However, as opposed to the approach presented here, the objects were categorized in an unsupervised manner independent of the effects of robot actions.

### A. Formalization of affordances

Our formalization [2] is based on relation instances of the form (effect, (entity, behavior)), meaning that there exists a

potential to generate a certain effect when the behavior is applied on the entity by the agent. The entity represents the state of the environment (including the perceptual state of the agent) as perceived by the agent. The behavior represents the physical embodiment of the interaction of the agent with the environment, and the effect is the result of such an interaction. For instance, the lift-ability affordance implicitly assumes that, when the lift behavior is applied on a stone entity, it produces the effect lifted, meaning that the position of the stone, as perceived by the agent, is elevated.

A single (effect, (entity, behavior)) relation instance is acquired through a single interaction with the environment. But this single instance does not constitute an affordance relation by itself, since it does not have any predictive ability over future interactions. Affordances should be generic relations with predictive abilities. To achieve the prediction abilities, the similar relation instances can be grouped in the same classes so that the robot can generalize particular instances to general relations. This grouping strategy will be described in Section III.

### B. Representation of affordance relations

*Behavior:* Robot interacts with the world through its pre-coded behaviors. Although behaviors are pre-coded by programmers, they can be in different complexity levels. The affordances framework supports open-loop non-parametric behaviors like *move robot forward for 1 meters*, or open-loop parametric behaviors like *rotate robot  $\theta$  degrees and move it  $d$  meters*, or closed-loop behaviors like *move robot towards the object in position  $(x,y)$* . In general the behaviors are represented by  $b(\nu)$  where  $\nu$  corresponds to the parameter vector. In cases where non-parametric behaviors are used,  $b$  notation is used.

*Entity and Effect:* In object-free representation, an entity corresponds to the features that are perceived in the environment. In this paper, entities are represented by the initial feature vectors ( $f_o$ ) computed for the detected object  $o$ . The effect observed on this object during execution of a behavior is computed by taking the vectorial difference between the final and initial features of the object:

$$\xi_o^{b(\nu)} = f'_o - f_o$$

where  $\xi_o^{b(\nu)}$ ,  $f'_o$  and  $f_o$  represents the effect, final and initial feature vectors, and  $b(\nu)$  corresponds to the behavior executed.

*Affordance Relation Instances:* In the interaction phase, in order to learn affordances and develop generic affordance relations, a number of (effect, (entity, behavior)) relation instances are acquired through interactions with the environment. For different parameters of behaviors, the robot interacts with the environment, and stores the entities and effects together with the corresponding behavior parameters in the set of relation instances (note that we misused the set notation for brevity):

$$\Gamma = \{(\xi_o^{b(\nu)}, (f_o, b(\nu)))\}$$

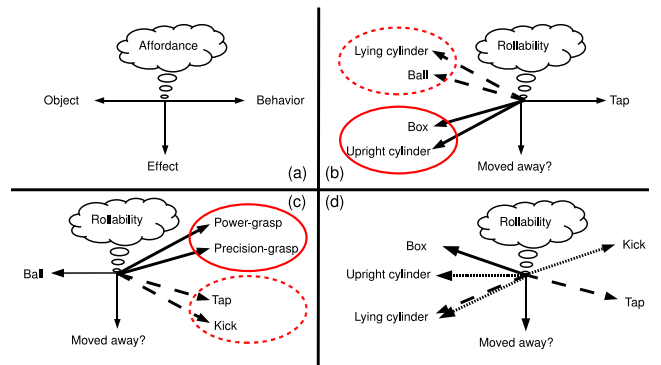


Fig. 1. (a) An affordance represents the general relationship among entities (shown as objects here), behaviors and effects. (b) The objects shown with the same type of lines are grouped in the same entity equivalence classes which correspond to the same type of ‘moved-away’ effects. (c) The behaviors with the same type of lines create similar effects and are grouped in the same effect equivalence classes. (d) The (entity,behavior) pairs with all dashed lines correspond to same affordance equivalence class with the common effect of ‘moved-away’. Specifically, (upright-cylinder, kick), (lying-cylinder,kick), (lying cylinder, tap) pairs generate the same move-away effect by rolling the objects away. However, since tap behavior is not strong enough to overturn the upright-cylinder, (upright cylinder, tap) pair does not afford rollability. (The figure is adapted from [2])

*Discovery of effect categories:* After interaction is completed, by using effect instances ( $\{\xi_o^{b(\nu)}\}$ ) in  $\Gamma$ , similar effects are grouped together to get a more general description of the effects that the behavior repertoire of the robot can create. This grouping can be achieved with unsupervised clustering techniques like straight-forward k-means or Self-Organized Maps if number of desired effect categories are known, or with X-means if category number is unknown. If complete course of the behavior is important, temporal effect trajectories are clustered as in [5]. Since many clustering techniques are sensitive to the number and shape of the clusters, more advanced methods such as kernel-based ones can be used [10]. After clustering is completed, prototypes for each effect category ( $\bar{\xi}_c$ ) are computed by taking the mean of the effects in that category ( $c$ ). Hence, for a given effect ( $\xi$ ), the corresponding category can be found as:

$$c_\xi = \operatorname{argmin}_{1 \leq i \leq k} \|\xi - \bar{\xi}_i\| \quad (1)$$

where  $1 \leq i \leq k$  is the category index.

### III. EQUIVALENCE CLASSES

A robot can learn the mapping between the entities and the effect categories or between the behaviors and the effect categories in order to predict the effects it will create when it applies a given behavior on a given entity. This learning entails formation of entity, behavior or affordance equivalence classes. Figure 1 gives illustrative examples for different equivalence classes. When entity equivalence classes are formed, the robot gains the ability to select the object to acquire a certain effect when a certain behavior is applied. In [11], [12] entity equivalence classes are formed for traversability affordance

using pre-defined and discovered effect categories, respectively. Generation of behavior equivalence classes has not been explicitly studied yet, since it requires either a very rich discrete behavioral repertoire as in Figure 1 or the behaviors that are parametric. Finally, affordance equivalence classes that are more generic and complex than these two classes, can be formed by learning the mapping from (entity-behavior) pairs to effect categories. Using affordance equivalence the robot gains the flexibility to predict the effect based on object feature values and behavior parameters. In other words, the robot is able to choose a behavior-object pair for obtaining a desired effect. The realization details of these equivalence classes are as follows:

*Entity equivalence:* The class of entities that generates same effect when a certain behavior is applied forms entity equivalence class. To acquire entity equivalence class, the mapping between the entities and the anticipated effect categories are learned for the particular behavior. This is achieved by training classifiers with the collected affordance relation instances. Formally, the mapping between the object features and the effects created by a particular behavior  $b$  is learned by a classifier ( $\chi^b$ ) using the data set:

$$\mathbf{T}^b = \{(\mathbf{f}_o, c_{\xi_o^b})\}$$

where  $\mathbf{f}_o$  is given as the input feature vector to the classifier  $\chi^b$ , and  $c$  is the corresponding target category. After training, predicted effect category can be found without applying behavior  $b$  to an object with perceptual features  $\mathbf{f}_o$  by:

$$c^{\text{predicted}}(b, o) = \chi^b(\mathbf{f}_o)$$

Formally, objects  $o_i$  and  $o_j$  are in the same entity equivalence class for behavior  $b$  if and only if the same effect category is predicted for the same behavior execution:

$$o_i \sim o_j \text{ iff } \chi^b(\mathbf{f}_{o_i}) = \chi^b(\mathbf{f}_{o_j})$$

*Behavior equivalence:* The class of behaviors which support the generation of the same effect category when applied to a certain entity is called behavior equivalence class. Similar to entity equivalence classes, the mapping between behaviors and the effect categories can also be found by training classifiers with the collected set of behavior parameters. Formally, the relationship between the behaviors and the effects created for a given object is learned by a classifier using the data set:

$$\mathbf{T}^o = \{(\boldsymbol{\nu}, c_{\xi_o^{b(\boldsymbol{\nu})}})\}$$

where  $\boldsymbol{\nu}$  is given as the input feature vector to classifier, and  $c$  is the corresponding target category. After training, predicted effect can be found without applying behavior  $b(\boldsymbol{\nu})$  to the object  $o$  by:

$$c^{\text{predicted}}(b, o) = \chi^o(\boldsymbol{\nu})$$

Formally, behaviors  $b(\boldsymbol{\nu}_1)$  and  $b(\boldsymbol{\nu}_2)$  are in the same behavior equivalence class for object  $o$  if and only if same effect category is predicted when they are applied to the object  $o$ :

$$b(\boldsymbol{\nu}_1) \sim b(\boldsymbol{\nu}_2) \text{ iff } \chi^o(\boldsymbol{\nu}_1) = \chi^o(\boldsymbol{\nu}_2)$$

*Affordance equivalence:* This complex equivalence can be achieved by finding the mapping from entity features and behavior parameters to effect categories. That is, a desired effect can be accomplished through different (entity, behavior) relations. One approach for this can deal with entities and behaviors independently, ie. by first linking entities to effect categories and then linking behaviors to effect categories and finally combining these two mappings in a third step. This corresponds to finding affordance equivalence based on computed entity and behavior equivalences. However, the entities and behaviors are coupled in terms of generated effects. For example, consider affordance equivalence classes are learned for push behavior, where a rollable cylinder is included in the environment. For simplicity, the entity has only one feature, namely object orientation, and the behavior has only one parameter, namely the direction of push. It is impossible to learn the mapping between push direction and effect category without considering the object orientation, or learn the mapping between the object orientation and the effect category without considering the direction of push. Instead, the entity features and the behavior parameters can be transferred to a joint input space using transformation function  $\Psi(\cdot)$ . Then, the mapping between this space and the effect categories can be learned. Formally, the relationship between the (entity, behavior) pair and the effect categories is learned by a classifier using the data set:

$$\mathbf{T} = \{(\Psi(\boldsymbol{\nu}, \mathbf{p}_o), c_{\xi_o^{b(\boldsymbol{\nu})}})\}$$

After training, predicted effect can be found without applying behavior  $b(\boldsymbol{\nu})$  to the object  $o$  with features  $\mathbf{f}_o$  by:

$$c^{\text{predicted}}(b, o) = \chi(\Psi(\boldsymbol{\nu}, \mathbf{p}_o))$$

Formally, (object,behavior) pairs  $(o_1, b(\boldsymbol{\nu}_1))$  and  $(o_2, b(\boldsymbol{\nu}_2))$  are in the same affordance equivalence class if and only if the same effect category is predicted when behaviors are applied to the objects respectively:

$$(o_1, b(\boldsymbol{\nu}_1)) \sim (o_2, b(\boldsymbol{\nu}_2)) \text{ iff } \chi(\Psi(\boldsymbol{\nu}_1, \mathbf{p}_{o_1})) = \chi(\Psi(\boldsymbol{\nu}_2, \mathbf{p}_{o_2}))$$

*Input space transformation:* The aim of  $\Psi(\cdot)$  function is two-fold. First, since the representation of behavior parameters and object features are intrinsically in different levels, the two corresponding vectors may not be combined in a straightforward way. Second, generally only a small fraction of the features/parameters are used in affordance prediction. For example although object color might be a very informative feature in many applications, it is not relevant for physical affordances such as pushing or rolling. In a similar vein, object coordinates are used to guide manipulation behaviors like grasping, but these coordinates do not determine whether the object is graspable or not (if the object is in a reachable distance). As a result, in order to provide perceptual economy and minimize the input space for training the classifiers, the irrelevant features and parameters are automatically excluded using this space transformation function.

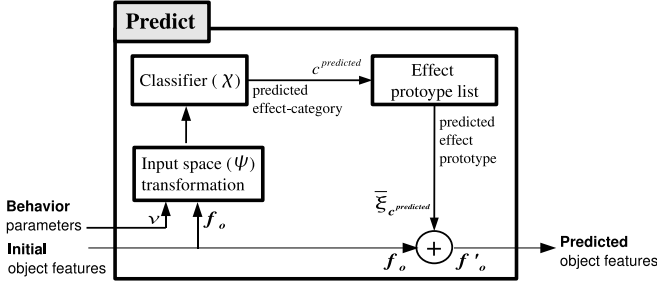


Fig. 2. The *Predict*-operator that is trained to predict the anticipated perceptual features of an object when the behavior is applied to the object

#### IV. PREDICTION BASED ON LEARNED AFFORDANCES

The learned affordance equivalence classes can be used to estimate the future perceptual states of the objects that the robot will perceive after the execution of its behaviors. This prediction is performed by adding the prototype of predicted effect features to the currently perceived object features. The basic prediction operator is demonstrated in Figure 2. Formally the predicted features of an object upon application of a behavior can be computed as:

$$f'_o(\{b(\nu)\}) = f_o + \bar{\xi}_{c^b}^{b_{(b,o)}}$$

Based on the prediction operator, it is possible to predict the effects of behaviors over the estimated future object features, again using the learned relations. The robot can estimate the total effect that a sequence of behaviors will create and it can predict the entity that it will perceive after the execution of the sequence. As a simple example, assume that the robot learned entity equivalence classes for different discrete behaviors (turn-left and lift). It is then asked to predict the anticipated feature vector of an object when these two behaviors are applied in a sequence. Figure 3 shows a hypothetical 2-step prediction scheme. Note that, although the prediction operator in Figure 2 supports parametric behaviors, since entity (not affordance) equivalence classes are learned in this example, a separate classifier exists for each different discrete behavior.

#### V. DISCUSSION

In this paper, we focus on object affordances and propose a developmental method that would enable the robot to ground symbolic object-based operators in its own continuous sensory-motor experiences. The method allows the robot to learn the object affordance relations. These relations are used to predict the change in the percept of the object when a certain behavior is applied. The effect prediction is achieved by forming entity, behavior and affordance equivalence classes.

Entity equivalence class formation has been studied in mobile robots [11], [12], [13]. Although mobility related affordances have been well-studied, the outlined method should be verified in more complex domains such as in manipulation robots with richer behavioral repertoire. For this purpose, we implemented several behaviors including grasping and pushing the objects using an anthropomorphic Gifu hand mounted on

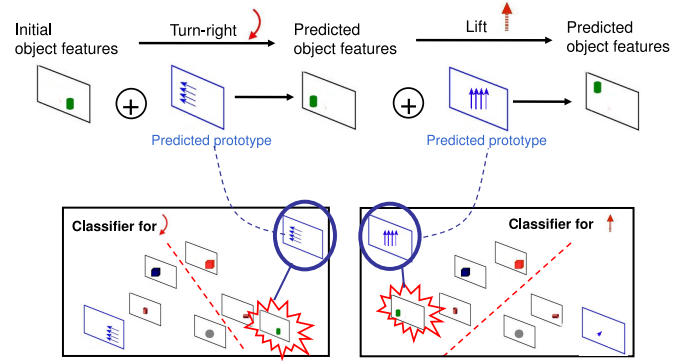


Fig. 3. A hypothetical situation that shows prediction process. The anticipated perceptual states of an object is shown when the two behaviors, namely turn-right and lift, are sequentially executed. Top figure shows the schematic explanation of summing of object feature vectors and predicted effect prototypes. Bottom figure shows how entity space is classified and how entity equivalence classes are formed based on discovered effect categories. Although effects are not represented in entity space, their prototypes are shown on different corners of the classifier to demonstrate the mapping.

PA10 arm system with 23 degree of freedoms. Training is being done in a physics based simulator where SR4000 range camera is used as the main sensor unit.

In the cited studies, behaviors are pre-coded as discrete, open-loop and unrelated controllers. When the behaviors are designed as more complex parametric closed-loop controllers, as it is the case in most of the robotic applications, learning only entity equivalence classes will not be sufficient to learn the affordances of the environment and to predict the effects created. In an on-going study, we are currently working on a specific but complex behavior, grasp behavior whose execution is based on the joint angles parameters and time. Using a fixed set of objects and human-guided correct grasp instances, behavior equivalence classes are expected to be formed in joint angle space for different objects.

#### VI. ACKNOWLEDGMENTS

This work was partially funded by the European Commission under the ROSSI project (FP7-004381). We would like to acknowledge that the ideas presented in this paper partially shaped through discussions with Mehmet Remzi Dogar and Maya Çakmak.

#### REFERENCES

- [1] J. Gibson, *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates, 1986.
- [2] 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, no. 4, pp. 447–472, 2007.
- [3] P. Fitzpatrick, G. Metta, L. Natale, A. Rao, and G. Sandini, "Learning about objects through action -initial steps towards artificial cognition," in *Proc. of ICRA 03*, 2003, pp. 3140–3145.
- [4] 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, no. 1, pp. 15–26, 2008.
- [5] J. Sinapov and A. Stoytchev, "Detecting the functional similarities between tools using a hierarchical representation of outcomes," in *7th IEEE International Conference on Development and Learning*, 2008.

- [6] J. Pisokas and U. Nehmzow, "Experiments in subsymbolic action planning with mobile robots," in *International Symposium on Robotics and Automation*, 2002.
- [7] C. Geib, K. Mourão, R. Petrick, N. Pugeault, M. Steedman, N. Krueger, and F. Wörgötter, "Object action complexes as an interface for planning and robot control," in *Workshop: Towards Cognitive Humanoid Robots at IEEE RAS Int Conf. Humanoid Robots*, 2006.
- [8] S. Hart, R. A. Grupen, and D. Jensen, "A relational representation for procedural task knowledge," in *AAAI*, 2005, pp. 1280–1285.
- [9] J. Modayil and B. Kuipers, "Autonomous development of a grounded object ontology by a learning robot," in *Proc. of AAAI*, 2007, pp. 1095–1101.
- [10] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik, "Support vector clustering," *Journal of Machine Learning Research*, vol. 2, pp. 125–137, 2001.
- [11] E. Uğur, M. Doğar, M. Çakmak, and E. Şahin, "The learning and use of traversability affordance using range images on a mobile robot," in *Proc. of ICRA'07*, 2007.
- [12] M. R. Doğar, M. Çakmak, E. Uğur, and E. Şahin, "From primitive behaviors to goal-directed behavior using affordances," in *Proc. of IROS'07*, November 2007, (To appear).
- [13] M. Çakmak, M. Doğar, E. Uğur, and E. Şahin, "Affordances as a framework for robot control," in *Proceedings of The 7th International Conference on Epigenetic Robotics, EpiRob'07*, 2007, (To appear).