Affordances and Emergence of Concepts

İlkay Atıl  Nilgün Dağ  Sinan Kalkan  Erol Şahin
KOVAN Research Lab, Dept. of Computer Engineering,
Middle East Technical University, Ankara, TURKEY
{ilkayatil,nilgundag,skalkan,erol}@ceng.metu.edu.tr

Abstract

In this paper, we used the affordance formalization framework (Sahin et al; Adaptive Behavior, 15(4), 447-472, 2007) to link the concepts represented by verbs and nouns in language to the affordance relations that a robot acquires through its interaction with the environment. First, we argued that a verb is linked not to a specific behavior of the robot, but to a specific effect that different behaviors may generate. We showed how demonstrations made by a human can allow the robot to learn an effect prototype in its own sensory-motor space, and use this to command the robot to achieve desired goals. Second, we investigated how the affordance relations of objects can be used to create noun concepts (to name objects) and how multi-task learning can utilize the learned object concepts to facilitate the learning of new affordances. The results are demonstrated on an iCub humanoid robot platform.

1. Introduction

The gap between the discrete concepts of language and the continuous and subjective sensory-motor experiences of agents is often referred to as the symbol grounding problem (Harnad, 1990) and has been acknowledged by many studies (Marocco et al., 2010, Steels and Kaplan, 2002, Roy and Pentland, 2002, Glenberg and Kaschak, 2002, Cangelosi et al., 2006, Dominey, 2005, Sugita and Tani, 2005).

Based on what ‘nouns’ and ‘adjectives’ represent in a language, one can take the the concepts represented by nouns (called ‘object concepts’ in the paper) as (1) perceptually different sets of features (i.e., appearance-based categorization) or (2) functionally different sets of features (i.e., function-based categorization, figure 1(b)). It is known that adults utilize both kinds of mechanisms for categorizing objects (Borghi et al., 2002); yet, for the current paper, we are interested in item (2) since (i) it is developmentally more relevant and (ii) item (1) is a well-established research area in Computer Vision and left out of the scope of the current paper.

In contrast to ‘noun concepts’, the issue of what ‘verb concepts’ (i.e., concepts represented by verbs) refer to is controversial. The literature has mostly taken each individual behavior as a ‘verb concept’ (see, for example, (Cangelosi and Riga, 2006)). With this view, for example, one should associate one verb for each push behavior in figure 1(a). The alternative view is to associate verbs to each effect that can be achieved on the entities; in a sense, this corresponds to generalizing over behaviors first to find the behaviors leading to similar effects and then linking verbs with these generalizations.

The concept of affordances, first introduced by (Gibson, 1986), is a suitable framework for investigating the co-development of behavior and perception and language although Gibson coined the term affordance (as the action possibilities offered by the environment to the actor) for studying perceptual processes. As E. J. Gibson stated, learning affordances also leads to discovering features and invariant properties of objects (Gibson, 2000), which can be said to correspond to the ‘object concepts’ that we have talked above. In this paper, we extend what E. J. Gibson has proposed and claim that through affordances, an organism also discovers ‘verb concepts’, as defined above.

As proposed by (Sahin et al., 2007), we formalize an affordance as a relation between an entity (e), a

Figure 1: (a) Different behaviors one can push an object. (b) A set of perceptually different but functionally equivalent objects².

behavior \((b)\) and an effect \((f)\) (see figure 2(a)):

\[
a = (e, b, f),
\]

where we take as an entity \(e\) the raw perceptual appearance of what is available in the scene, and as the effect \(f\) the difference between the post-behavior and pre-behavior appearances of the entity (i.e., \(f = e' - e\), where \(e'\) is the post-behavior appearance of the entity).

In the light of the above discussions about what ‘object concepts’ and ‘verb concepts’ should be, we can formulate the ‘object concepts’ as \((e, < b, f >)\), where \(< >\) denotes the equivalence relation, and ‘verb concepts’ as \((< e, b >, f)\) (as shown in figure 2(b) and (c)).

Derivation of ‘object concepts’ or ‘verb concepts’ through interactions has already attracted substantial interest in the literature. For the case of ‘object concepts’, for example, (Nolfi and Marocco, 2002) used the tactile information retrieved from touch sensors for categorizing objects. In (Da˘ ĭg et al., 2010), a robot explores the affordances of the objects and then, objects were grouped based on what they afforded. For a different setup for a mobile robot that needs to traverse an environment, (Ugur and Sahin, 2010, Sun et al., 2010) proposed learning through interactions the categories of entities that allow traversability.

As for the ‘verb concepts’, the literature has attributed verbs to individual behaviors (Marocco et al., 2010, Krunic et al., 2009, Cohen et al., 2005). Similar to our framework, (Rudolph et al., 2010) proposes relating behaviors to the representations of their effects yet generalization over behaviors or effects is not pursued for deriving ‘verb concepts’. (Kozima et al., 2002) has also proposed generalization over behaviors based on their effects yet only for imitation purposes.

In this paper, in an embodied environment, we investigate how a robot can develop such object and verb concepts through interactions, namely, while deriving the affordances of the objects in the environment. We show that our affordance formalism allows several ways for yielding ‘object concepts’, which are derived from what the entities afford. An important contribution of the current paper is demonstrating that the derived object categories helps learning new tasks\(^3\) faster. Moreover, using the same framework, we show that marginalization over the entities and the behaviors leads to ‘verb concepts’ and we demonstrate the ‘verb concepts’ in a simple imitation task (although our aim in this paper is not proposing another imitation framework).

2. Methods

2.1 Data

We acquired the perceptual data from a SwissRanger 4000 range camera which can capture depth scenes with a resolution of \(176 \times 144\) at 30 fps. The range camera provides three images: the range data, the amplitude of the signal and the confidence of the signal.

![Image of objects used in the experiments](image-url)

Figure 3: Objects used in the experiments. The cylinders and the boxes have handles.

--ACTION, behavior or task learning, in the current paper, means acquiring the ability to predict the outcome of a behavior on different objects.
In the experiments, objects of different sizes (big, medium, small) and of different shapes (ball, cylinder, cube) are used (shown in figure 3). We perform simple behaviors, namely push-right, push-left, push-forward, rotate 45°, rotate 90° and lift on these objects. The behaviors are performed by the user as though they were applied by a robot arm. We used the robot for testing the learned ‘verb concepts’.

2.2 Features

We segmented the objects through background segmentation. From the segmented range data, we extract the following features:

- The 3D position of the object,
- The shape of the object as a 10-bin histogram of the shape indexes (Koenderink and van Doorn, 1992),
- The 3D orientation of the object using an SVM classifier trained to estimate the orientation of the entities,
- The size of the object along X, Y and Z axes.

The features extracted from the object before the behavior are called the initial features, i.e., the entity $e$ in equation 1, whereas the features extracted after the behavior are called the final features. The difference between the final and the initial features defines the effect features, i.e., the effect $f$ in equation 1.

2.3 Affordance Learning Model

In order to learn an affordance relation, we assume that clusters in the effect space are labeled by a human supervisor. Figure 4 shows the effect clusters derived as a result of such a labeling for our data. Note that the effects arisen from different behaviors can be assigned to the same clusters.

Using these labels, we train a Support Vector Machine classifiers for each behavior to map the initial features to the effect clusters. Then, these SVMs are used to predict which effect cluster a new object can yield to for a given behavior to be applied on the object (see figure 5).

3. Verb Concepts

As pointed out in section 1., we argue that verbs should correspond to a generalization of behaviors (called ‘verb concepts’) that achieve the same effect.

For deriving verb concepts, we find the effect categories in the effect space and represent them in a compact way, which we call the ‘effect prototypes’. We claim that, for a robot, an ‘effect prototype’ can be the label for a verb concept which can be linked to a verb.

3.1 Deriving Effect Prototypes

The effect prototypes can be obtained from labeled effect clusters in the effect space (figure 4). One sees from the change of feature elements in each effect cluster (figure 6) that some feature elements increase consistently while others decrease, stay constant or change in an unpredictable way. Therefore, we can represent an effect prototype as a string consisting of ‘+’, ‘−’, ‘0’, ‘*’, corresponding respectively to increase, decrease, no change and unpredictable change in the corresponding feature element. Each element of the effect prototype has the mean and variance of the observed change, along with the description of the change.

We assign the labels ‘+’, ‘−’, ‘0’ and ‘*’ to feature elements using unsupervised clus-
tering (namely, Robust Growing Neural Gas (Qin and Suganthan, 2004)) in the space of mean and variance (i.e., we (i) collect a set of effects, (ii) compute the mean and variance of each element in this set of effects and (iii) group feature elements based on their mean and variance into four clusters using unsupervised clustering and (iv) give the labels ‘+’, ‘−’, ‘0’ and ‘∗’ to the clusters based on what they represent), which yields to the effect prototypes (only strings are shown) displayed in table 1.

3.2 Effect Prototypes as Verb Concepts

In this section, we demonstrate that the robot can understand others’ behaviors in terms of the verb concepts that he has developed4 shown in figure 75. In other words, the robot can observe an behavior and find the verb concept (i.e, the concept that a corresponding verb would correspond to) that would describe the observed behavior.

Moreover, a goal given to the robot as a verb concept (as if it were given verbally) is interpreted by the robot and matched with a behavior that the robot can execute to achieve the instructed goal (shown in figure 8).

Figure 6: Distribution of effect features for different effect categories.

4Linking the effect prototypes that we have extracted in section 3.1 with verbs either through supervision or a more developmental approach like Talking-heads Experiment (Steels, 2003) is an interesting future work that we would like to pursue. Since it is not the focus of this paper, we demonstrate the developed verb concepts directly using effect prototypes.

5This understanding requires that a distance metric exists between the effect categories. We use the mahalanobis distance using the mean and variance of the effect categories.
Table 1: Effect prototype strings that are extracted using unsupervised clustering. Note that each feature element has an associated mean and variance of the change. $\theta$ stands for orientation.

<table>
<thead>
<tr>
<th>Position (X-Y-Z)</th>
<th>Shape</th>
<th>Orientation</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pushed Right</td>
<td>+00</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rolled Right</td>
<td>+00</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Pushed Left</td>
<td>-00</td>
<td>*</td>
<td>0</td>
</tr>
<tr>
<td>Rolled Left</td>
<td>-00</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Pushed Forward</td>
<td>0--</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rolled Forward</td>
<td>0--</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>No effect</td>
<td>000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rotated 45$^\circ$</td>
<td>000</td>
<td>*</td>
<td>+</td>
</tr>
<tr>
<td>Rotated 90$^\circ$</td>
<td>000</td>
<td>***</td>
<td>*</td>
</tr>
<tr>
<td>Lifted</td>
<td>0+-</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 7: After being demonstrated a behavior and introduced a new object, iCub understands the ‘verb concept’ that corresponds to executed behavior and imitates it. The robot finds out that the observed behaviors in (a), (b) and (c) respectively corresponds to the verb concepts ‘0–0000000000000000’, ‘-00000000*00000000’ and ‘+0000000000000000’.

Figure 8: iCub is given two ‘verb concepts’ and decides to execute push forward and push right behaviors on the objects successfully achieving the goal.

4. Object Concepts

As pointed out in section 1., objects can be categorized based on appearance and function. In this section, based on their function, we propose two ways for the emergence of object concepts, or categories, based on function: (1) Categorization using the effects of behaviors on the objects and (2) Grouping entities based on the outcome of each behavior. The reason that we have two different proposals for the same purpose is because there may be more than one way to categorize objects based on their function.

4.1 Categorization Using Effects

In (Da˘g et al., 2010), we studied how we can represent objects using their affordances in a simplified setting. We demonstrated that one can use the affordances of objects to categorize them using the fact that different objects afford different behaviors (or similar behaviors with different effects). For this, we found clusters in the $<e_1, \ldots, e_i, \ldots, e_n>$ space, where $e_i$ is the predicted effect of the $i_{th}$ behavior on the object.

Fig. 9 shows categories that emerge from our current setup. Note that cluster 2 is mainly composed of balls for the reason that push behaviors make it easy to distinguish them from boxes and cylinders. Instances in the other two categories (or clusters 1 and 3) are separated in accordance with their sizes as push behaviors executed on these objects does not suffice
to differentiate cylinders from boxes rather effect of lift behavior decides on the form of these categories.

4.2 Categorization Per Behavior

In this section, we present an alternative system to the emergence of concepts presented in section 4.1; although they are different, they might lead to similar object concepts since they both rely on the functions of objects. Moreover, we show that the object concepts that have been acquired mediates fast learning of new tasks, i.e., behaviors.

The proposed system incrementally works as follows for each behavior (see figure 10):

- We cluster the effects of the behaviors (the same set of clusters as in figure 4) using RGNG.

- We inspect the entities in each cluster and find the feature element $f_i$ that best distinguishes that the entities in that effect cluster from the entities in other effect clusters.

The distinguishing feature element is found using a simple feature selection algorithm (shown in algorithm 1). This algorithm rates the features according to their separation ability of different groups. After we rate the features, we select the most relevant feature for clustering.

**Algorithm 1 A Simple Feature Selection Algorithm**

```
ent(i,f): $i^{th}$ feature of the $i^{th}$ entity
effect-group(i): effect group of the $i^{th}$ entity
feature(k): rating of the $k^{th}$ feature
feature(all) ← 0
for all i in entity set do
    for all k in feature set do
        Find the closest entity $j$ where effect-group(i) ≠ effect-group(j)
        feature(k) ← feature(k) + abs(ent(i,k) - ent(j,k))
    end for
end for
```

- We perform unsupervised clustering (RGNG) in the entity space on the feature element $f_i$ unless clustering in the entity space on the feature element $f_i$ has not been done before.

![Figure 9: Categories of the objects. Numbers represent the number of members in each category.](image)

![Figure 10: In (a), the system learns the behavior ‘push left’ and discovers that the entities that have led to the effect category ‘pushed left’ and ‘rolled left’ are distinguishable based on the feature element $f_1$ (found using the feature selection algorithm in algorithm 1). Clustering the entities in the entity space based on the feature element $f_1$ separates roll-able and unroll-able objects. In (b), the system is asked to learn the ‘push right’ behavior, which leads to the same distinguishing feature element $f_1$. Since the entities have already been grouped based on the feature element $f_1$, ‘push right’ behavior can utilize the same groups (‘noun concepts’) in the entity space. In (c), however, the system encounters the ‘lift’ behavior, which requires a different distinguishing feature, $f_2$; therefore, the system cannot utilize the already existing object concepts and needs to perform a separate grouping, which separates small vs big objects.]
4.2.1 Learning Behaviors Using Object Concepts

In this section, using neural networks, we show that the object concepts developed in section 4.2 helps learning new behaviors (which we might safely call ‘Multi-Task Learning’ (MTL)). We compare MTL with ‘Single-Task Learning’ (STL), where the behavior is learned separately.

We assume that MTL has already learned two behaviors (which are not related to each other based on the related features that affect the outcomes). Then, we try to learn a new behavior along with the already learned behaviors (MTL) or separately (STL) using the networks shown in figure 11. We compare the following three approaches:

- **MTL-1**: Learning the task (‘push right’) in the already learned MTL where the outcome of the network is the raw effect space (the network in figure 11(a)). MTL has already learned the ‘push left’ and ‘lift’ behaviors, and we add the task ‘push right’ to the MTL for learning.

- **MTL-2**: Learning the task (‘push right’) in the already learned MTL where the outcome of the network is the raw effect space (the network in figure 11(b)). MTL has already learned the ‘rotate’ and ‘lift’ behaviors, and we add the task ‘push right’ to the MTL for learning.

- **STL**: Learning the task (‘push right’) separately (STL) using the network in figure 11(c).

Figure 12 shows the number of epochs required for achieving the same accuracy (namely, a mean-squared error of 0.01) for learning MTL-1, MTL-2 and STL. We see that since ‘push right’ behavior relies on the same concepts that the ‘push left’ behavior has produced, MTL-1 can learn faster compared to STL. In contrast, as shown in figure 12, when learning ‘push right’ behavior where MTL-2 knows unrelated behaviors (‘rotate’ and ‘lift’), acquired object concepts in MTL-2 does not mediate learning, making the learning process slower then STL (due to concept generation process).

5. Discussion

In this paper, we presented a new approach for deriving ‘noun concepts’ (that mainly can be linked with nouns) and demonstrated that the learned ‘object concepts’ helps learning new tasks faster, in cases when the tasks are related to the already learned tasks. We have also proposed linking verbs with, what we call as, ‘verb concepts’, which are generalizations over behaviors defined in the effects of the behaviors. Although the main aim of the paper is not imitation, we have demonstrated that the robot could find the ‘verb concept’ corresponding to an observed behavior, or it can satisfy a goal given as a ‘verb concept’.

Incorporation of dynamic aspects of behaviors is an important future step since dynamic aspects are known to be important for verb concepts or semantics (Cohen et al., 2005, Cannon and Cohen, 2006).

Acknowledgments

This work is partially funded by the EU projects ROSSI (FP7-ICT-216125), and RobotCub (FP6-ICT-004370), and by TÜBİTAK (Turkish Scientific
Ilkay Atlı acknowledges support of the TÜBİTAK graduate student fellowship.

References


