

Toward Interactive Learning of Object Categories by a Robot: A Case Study with Container and Non-Container Objects

Shane Griffith, Jivko Sinapov, Matthew Miller and Alexander Stoytchev
Developmental Robotics Laboratory
Iowa State University
{shaneg, jsinapov, mamille, alexs}@iastate.edu

Abstract—This paper proposes an interactive approach to object categorization that is consistent with the principle that a robot’s object representations should be grounded in its sensorimotor experience. The proposed approach allows a robot to: 1) form object categories based on the movement patterns observed during its interaction with objects, and 2) learn a perceptual model to generalize object category knowledge to novel objects. The framework was tested on a container/non-container categorization task. The robot successfully separated the two object classes after performing a sequence of interactive trials. The robot used the separation to learn a perceptual model of containers, which, in turn, was used to categorize novel objects as containers or non-containers.

I. INTRODUCTION

Object categorization is one of the most fundamental processes in human infant development [1]. Yet, there has been little work in the field of robotics that addresses object categorization from a developmental point of view [2]. Traditionally, object categorization methods have been vision based [3]. However, these disembodied approaches are missing a vital link, as they leave no way for a robot to verify the correctness of a category that is assigned to an object. Instead, the robot’s representation of object categories should be grounded in its behavioral and perceptual repertoire [4] [5].

This paper proposes an embodied approach to object categorization that allows a robot to *ground* object category learning in its sensorimotor experience. More specifically, the robot’s task is to detect two classes of objects: containers and non-containers. In the proposed framework, interaction and movement detection are used to ground the robot’s perception of these two object categories. First, the robot forms a set of outcome classes from the detected movement patterns during its interactions with different objects (both containers and non-containers). Second, objects are grouped into object categories by the frequency that each outcome class occurs with each object. Third, a perceptual model is learned and used to generalize the discovered object categories.

The framework was tested on a container/non-container categorization task, in which the robot dropped a block above the object and then pushed the object. First, the robot identified three outcomes after interacting with the objects: co-movement outcomes, separate movement outcomes, and noisy outcomes. Second, the robot identified that co-movement outcomes occurred more often with containers than with non-containers and thus separated containers from non-containers using unsupervised clustering. Third, a perceptual model was

learned and was shown to generalize well to novel objects. Our results indicate that the robot can use interaction as a way to detect the functional categories of objects in its environment.

II. RELATED WORK

A. Developmental Psychology

The theories postulated by developmental psychologists often lay the groundwork for the approaches taken in developmental robotics. This is most certainly the case with this paper. We believe that robots could be better equipped to categorize objects by investigating how infants acquire the same ability. According to Cohen [1], infants form object categories by processing the relationships between certain events (e.g., movement patterns).

Infants have an innate ability to perceive objects as connected, bounded wholes (cohesion principle), which allows them to predict *when* an object will move and *where* it will stop moving [6]. The cohesion principle could be violated in two ways: 1) objects that are perceived as separate entities are observed to move together; and 2) objects that are perceived as a single entity are observed to move separately. Therefore, it is reasonable to assume that infants learn *move together* and *move separately* events from experiences that violate the cohesion property. It follows that if a robot can sense the duration of movement and the co-movement patterns of objects, it could learn from these events.

An infant’s perception of objects affects whether the cohesion property is violated or not. Needham *et al.* [7] showed that at 7.5 months infants expect a key ring and keys to move separately, while at 8.5 months infants expect them to move together. This shows that with experience, infants are able to associate the *move together* outcome with some object categories. Thus, it is reasonable to assume that a robot could discover object categories by interacting with multiple objects.

This paper tests two assumptions: 1) a robot can learn from the co-movement patterns of two different objects; and 2) a robot can discover object categories from these movement patterns. It does so by testing whether a robot can discover what humans naturally call containers as an object category. A container has the property that objects in the container move with it, whereas objects beside it do not. We suggest that this property is one embodied definition of containers that a robot can easily learn. In fact, several studies in psychology have relied on this phenomenon to determine infants’ knowledge of containers [8] [9] [10].



Fig. 1. The robot's vision system: a) the ZCam from 3DV Systems [11]; b) color image of the red bucket captured by the camera when mounted on the robot; c) the depth image corresponding to b).

B. Developmental Robotics

The work of Pfeifer and Scheier [12] is one of the earliest examples of object categorization by an autonomously exploring robot. They showed that the problem of categorizing three differently-sized objects was greatly simplified when the robot's own movements and interactions were utilized. In particular, a robot could grasp and lift small objects, push medium objects but not lift them, and do nothing with large objects. The robot ignored large objects that it could not manipulate, which allowed it to learn faster.

Additionally, Metta and Fitzpatrick [13] [14] found that object segmentation and recognition could be made easier through the use of a robotic arm. The arm scanned the scene and when it hit an object it detected a unified area of movement. The detected movement was used to delineate the object and construct a model for recognition. Furthermore, the robot poked the object to associate different movement outcomes (e.g., *rollable* and *non-rollable*) with the object model. Complex internal models were avoided because "the environment can be probed and re-probed as needed" [15].

Interaction-based methods can also work well for learning relations among objects, a problem closely related to object categorization. Sinapov and Stoytchev [16] showed that a simulated robot could infer the functional similarity between different stick-shaped tools using a hierarchical representation of outcomes. They also showed [17] that a robot could learn to categorize objects based on their acoustic properties. Similarly, in Montesano *et al.* [18], a robot that interacted with sphere- and cube-shaped objects discovered relationships between its actions, the objects' perceptual features (e.g., color, size, and shape descriptors), and the observed effects. The robot modeled the relationships with Bayesian networks.

Finally, in Ugur *et al.* [19], a simulated robot traversed environments that had random dispersions of sphere-, cylinder- and cube-shaped obstacles. It learned a perceptual model which identified the obstacles that could be traversed (spheres and lying cylinders in certain orientations) from the obstacles that could not be traversed (boxes and cylinders in upright positions). However, none of the robots in [16], [17], [18] or [19] learned explicit object categories.

This paper examines movement detection as a way to ground robot learning of object categories, specifically containers and non-containers. Edsinger and Kemp [20] have identified container manipulation as an important problem in robotics. In particular, they showed that two-armed robots have the precise control required to insert objects into containers. Following, this paper shows how robots can acquire the ability to identify containers from non-containers using interaction.



Fig. 2. The objects used in the experiments: a) the five containers: big red bucket, big green bucket, small purple bucket, small red bucket, small white bowl; b) these containers can easily become non-containers when flipped over.

III. EXPERIMENTAL SETUP

A. Robot

All experiments were performed with a 7-DOF Whole Arm Manipulator (WAM) by Barrett Technologies coupled with the three-finger Barrett Hand as its end effector. The WAM was mounted in a configuration similar to that of a human arm.

The robot was equipped with a 3-D camera (ZCam from 3DV Systems [11]). The camera captures 640x480 color images and 320x240 depth images. The depth resolution is accurate to ± 1 -2 cm. The camera captures depth by: 1) pulsing infrared light in two frequencies; 2) collecting reflected pulses of light; and 3) discretizing observed depth into pixel values. Figure 1 shows the 3-D camera and the camera's field of view when mounted on the robot.

B. Objects

The robot interacted with different container and non-container objects that were placed on a table in front of it (see Fig. 2). The containers were selected to have a variety of shapes and sizes. Flipping the containers upside-down provided a simple way for the robot to learn about non-containers. Therefore, the robot interacted with 10 "different" objects, even though there were only 5 real objects. During each trial the robot grasped a small block and dropped it in the vicinity of the object placed in front of it. The object was then pushed by the robot and the movement patterns between the block and the object were observed.

C. Robot Behaviors

Four behaviors were performed during each trial: 1) *grasp* the block; 2) *position* the hand in the area above the object; 3) *drop* the block; and 4) *push* the object. A person placed the block and the object at specific locations before the start of each trial. Figure 3 shows a sequence of interactions for two separate trials. The four behaviors are described below.

1) *Grasp Behavior*: The robot grasped the block at the start of each trial. The grasp behavior required the robot to open its hand, move next to the block, and close its hand.

2) *Position Behavior*: The robot positioned its hand in the area above the object after grasping the block. Drop positions were uniformly selected from a 40cm \times 40cm area relative to the center of the object. The object was consistently placed in the same location.

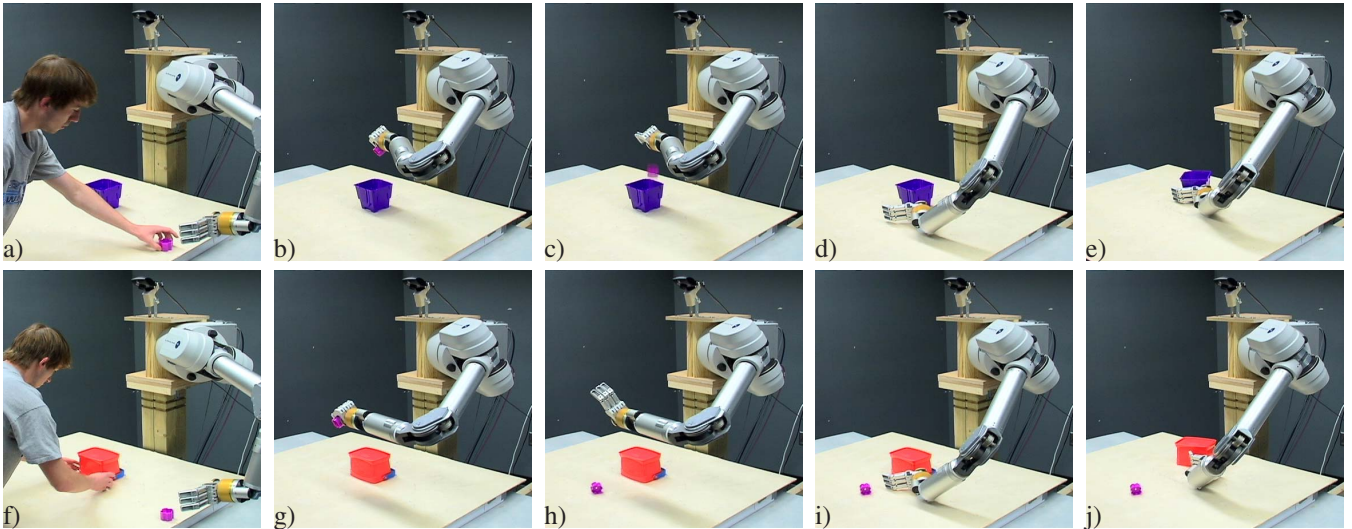


Fig. 3. The sequence of robot behaviors for two separate trials: a) before each trial a human experimenter placed the block and the container at a marked location; b) the robot carried out each trial by grasping the block and positioning the hand in the area above the container; c) dropping the block; d) starting the *push* behavior; e) and ending the *push* behavior. f-j) The same as a)-e) but for a non-container object.

3) *Drop Behavior*: The robot dropped the block once its hand was positioned in the area above the object. The block either fell into the object (except when the trial involved non-container objects), or fell beside it. In some cases the block rolled off the table (approximately 5% of 1000 trials). In these situations, a human experimenter placed the block at the location on the table where it rolled off.

4) *Push Behavior*: The robot pushed the object after dropping the block. The pushing direction was uniformly selected between two choices: *push-toward-self* or *push-toward-right-of-self*. The robot pushed the object for 10 cm with an open hand (see Fig. 3.d and 3.e).

IV. METHODOLOGY

A. Data Collection

Experimental data was collected during the *push* behavior. This interaction was captured from the robot's 3-D camera as a sequence of 640x480 color images and 320x240 depth images recorded at roughly 20 fps. The *push* behavior lasted approximately 3.5 seconds for a single trial. A total of roughly $3.5 \times 20 = 70$ images were recorded per trial.

For each of the 10 objects shown in Fig. 2 the robot performed 100 interaction trials for a total of 1000 trials.

B. Movement Detection

The robot processed the frames from the 3-D camera to detect movement and to track the positions of the block and the object. To locate each object, the color images were segmented based on the object's color and the coordinates of the largest blobs were calculated. The value for z was found at the corresponding $[x, y]$ position in the depth image. The last known position was used if the block or the object was occluded.

Movement was detected when the $[x, y, z]$ position of the block or the $[x, y, z]$ position of the object changed by more

than a threshold, δ , over a short temporal window $[t', t'']$. The threshold, δ , was empirically set to 10 pixels per two consecutive frames. A box filter with a width of 5 was used to filter out noise in the movement detection data.

C. Acquiring Interaction Histories

Once a trial, i , was executed, the robot constructed the triple (B_i, O_i, F_i) , indicating that the behavior $B_i \in \mathcal{B}$ was used to interact with object $O_i \in \mathcal{O}$ and outcome vector F_i was observed. The behavior represented with B_i was either *push-toward-self* or *push-toward-right-of-self*. Also, $\mathcal{O} = \{O_1, \dots, O_{10}\}$ denoted the set of objects (containers and non-containers) used in the experiments. Finally, each outcome was represented with the numerical feature vector $F_i \in \mathbb{R}^2$.

The outcome $F_i = [f_1^i, f_2^i]$ captured two observations: 1) whether the object O_i and the block moved at the same time, and 2) whether the object O_i and the block moved in the same direction. Hence, f_1^i equaled the number of time steps in which both the object and the block moved together divided by the number of time steps in which the object moved. In other words, the value of f_1^i will approach 1.0 if the object and the block move at the same time, but it will approach 0.0 if the object and the block do not move at the same time.

Additionally, the second outcome feature, f_2^i , was defined as $f_2^i = \|\Delta pos_i(object) - \Delta pos_i(block)\|$, where $\Delta pos_i(object) \in \mathbb{R}^3$ and $\Delta pos_i(block) \in \mathbb{R}^3$ are equal to the detected change in position of the object and the block, respectively, while they are pushed during trial i . In other words, the value of f_2^i will approach 0.0 if the object and the block move in the same direction, but it will become arbitrarily large if the object and the block move in different directions. Both f_1^i and f_2^i are required in order to represent whether the block and the object move together or move separately (see Fig. 4).

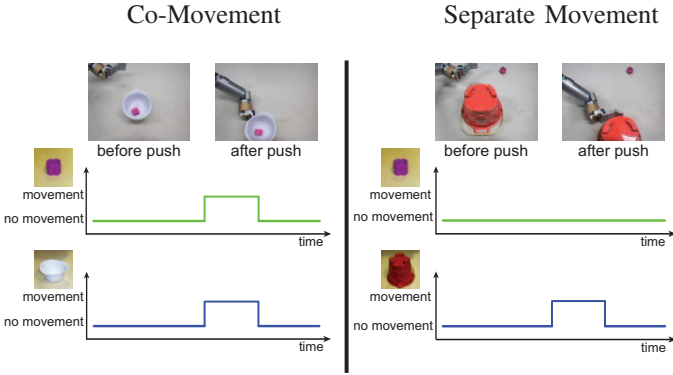


Fig. 4. An example of co-movement (left) and separate movement (right). Co-movement outcomes occur when the block falls into a container. In this case, the block moves when the container moves. Separate movement outcomes occur when the block falls to the side of the container or during trials with non-containers. In these instances the movements of the two objects are not synchronized.

D. Discovering Outcome Classes

Various co-movement patterns can be observed by acting on different objects in the environment. Outcome classes can be learned to represent these patterns. The robot’s interaction history would change over time, gradually growing more robust to outliers. A variety of factors affect the number of possible outcome classes (e.g., number of perceptual observations). Let $\{F_i\}_{i=1}^{1000}$ be the set of observed outcomes after performing 100 interaction trials with each of the 10 objects. We used unsupervised clustering with X-means to categorize the outcomes, $\{F_i\}_{i=1}^{1000}$, into k classes, $C = \{c_1, \dots, c_k\}$. X-means extends the standard K-means algorithm to estimate the correct number of clusters of the dataset [21]. Section V.A describes the results.

E. Discovering Object Categories

Certain outcome classes are observed more often with some objects than with others. This difference can be used to form object categories. For example, compared to non-containers, a container will more often exhibit the co-movement outcome when a small block is dropped above it. Therefore, the robot can use its interaction history with objects to discover different object categories, which might be how infants go about achieving this task [1].

Let us assume that the robot has observed a set of **outcome classes** $C = \{c_1, \dots, c_k\}$ from its interactions with several objects, $\mathcal{O} = \{O_1, \dots, O_{10}\}$. Let $H_i = [h_1^i, \dots, h_k^i]$ define the interaction history for object i , such that h_j^i is the number of outcomes from outcome class c_j that were observed when interacting with the i^{th} object.

The interaction histories were normalized using zero mean and unit standard deviation. Let the normalized interaction history, Z_i , for interaction history H_i be defined as $Z_i = [z_1^i, \dots, z_k^i]$, such that $z_j^i = (h_j^i - \mu_j) / (\sigma_j)$, where μ_j is the average number of observations of c_j , and σ_j is the standard deviation of observations of c_j . Through this formulation, the i^{th} object is described with the feature vector $Z_i = [z_1^i, \dots, z_k^i]$.

To discover **object classes**, the robot clustered the feature vectors Z_1, \dots, Z_{10} (one for each of the 10 objects shown in Fig. 2) using the X-means clustering algorithm. Clusters found by X-means were interpreted as object categories. X-means was chosen to learn both the individual *outcome classes* and *object classes* because: 1) it is an unsupervised clustering algorithm; and 2) it does not require the human programmer to know the number of clusters in advance. The results are described in section V.B.

F. Categorizing Novel Objects

It is impractical for a robot to categorize all novel objects by interacting with them for a long time. However, the robot can interact with a few objects to form a behavior-grounded object category and then learn a generalizable perceptual model from these objects. This method allows a robot to quickly determine the category of a novel object.

The predictive model could classify novel objects once it is trained with automatically labeled images. In this case, the robot interacted with 10 objects, so 10 depth images were used to train the predictive model, as shown in Figure 5 (only one image of each object was necessary since the robot viewed objects from a single perspective). The labels assigned to the 10 images were automatically generated by X-means during the object categorization step. For each depth image, let $s^i \in \mathbb{R}^n$ be a set of perceptual features extracted by the robot. The robot learns a predictive model $\mathcal{M}(s^i) \rightarrow k_i$, where $k_i \in \{0, 1, \dots, K\}$ is the predicted object category for the object described by features s_i , and K is the number of object categories detected by the X-means clustering algorithm.

The task, then, is to determine a set of visual features that can be used to discriminate between the learned clusters of objects. These objects have been grouped based on their functional features, *i.e.*, co-movement and non-co-movement. It is reasonable to assume that other features, like the shape of the objects, might be related to these functional properties, and therefore allow for the quick classification of novel objects into these categories. Presumably, as children manipulate objects and extract their functional features, they are also correlating visual features with their observations. Accordingly, the robot also attempted to build a perceptual model of containers by extracting relevant visual features and associating these features with the functional clusters.

To do this, the robot used the *sparse coding* feature extraction algorithm, which finds compact representations of unlabeled sensory stimuli. It has been shown that sparse coding extracts features similar to the receptive fields of biological neurons in the primary visual cortex [22], which is why it was chosen for this framework. The algorithm learns a set of basis vectors such that each input stimulus can be approximated as a linear combination of these basis vectors. More precisely, given input vectors $x_i \in \mathbb{R}^m$, each input x_i is compactly represented using basis vectors $b_1, \dots, b_n \in \mathbb{R}^m$ and a sparse vector of weights $s^i \in \mathbb{R}^n$ such that the original input $x_i \approx \sum_j b_j s_j^i$. The weights $s^i \in \mathbb{R}^n$ represent the compact features for the high-dimensional input image x_i . We used the algorithm and MATLAB implementation of Lee *et al.* [23] for learning the sparse coding representation.

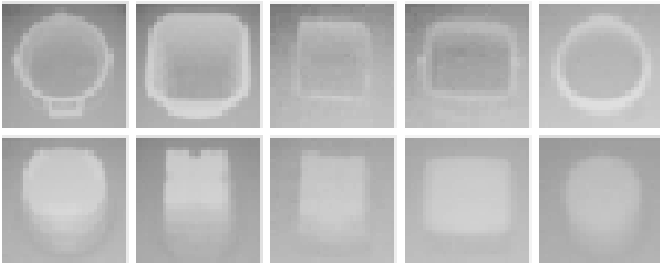


Fig. 5. The 10 depth images of the objects used as input to the sparse coding algorithm. The 320x240 ZCam depth images were scaled down to 30x30 pixels before the algorithms generated sparse coding features from them.

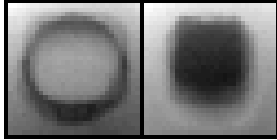


Fig. 6. The two basis vectors that were computed as a result of the *sparse coding* algorithm. These visual features were later used to classify novel objects as ‘containers’ or ‘non-containers.’

The robot extracted 2 features (i.e., $n = 2$ in the formulation above) from the 10 objects used during the trials, as shown in Figure 6. The figure shows that the algorithm extracted a feature characteristic of container objects and a feature characteristic of non-container objects. Each input x_i consisted of a 30 x 30 depth image of the object, as shown in Figure 5.

Given a novel object, O^{test} , the robot extracted a 30 x 30 depth image of it, x_{test} , and found the feature weight vector $s^{test} \in \mathbb{R}^2$ such that $x_{test} \approx \sum_j b_j s_j^{test}$. The robot then used the Nearest Neighbor algorithm to find the training input x_i (a 30 x 30 depth image of one of the 10 training objects) such that the Euclidean distance between its sparse feature weight s^i and s^{test} is minimized. The robot subsequently categorizes the novel object (as either ‘container’ or ‘non-container’) with the same class label as the nearest neighbor training data point.

V. RESULTS

A. Discovering Outcome Classes

Figure 7 shows the results of unsupervised clustering using X-means to group trials with similar outcome classes. The figure also shows the frequency with which each outcome class occurred for each container and non-container. X-means found three outcome classes among all of the trials: one cluster of co-movement events, one cluster of separate movement events, and a third cluster corresponding to noisy observations.

The first two outcome classes were expected. We found that the third outcome class had several causes. Sometimes the human experimenter was placing the block on the table after it fell off, sometimes the block was slowly rolling away from the container, and sometimes the movement detection noise was not completely filtered out. However, the fact that the robot formed a co-movement outcome class meant that it could find meaningful relationships among its observations. This result suggests that the robot could possibly categorize objects in a meaningful way.

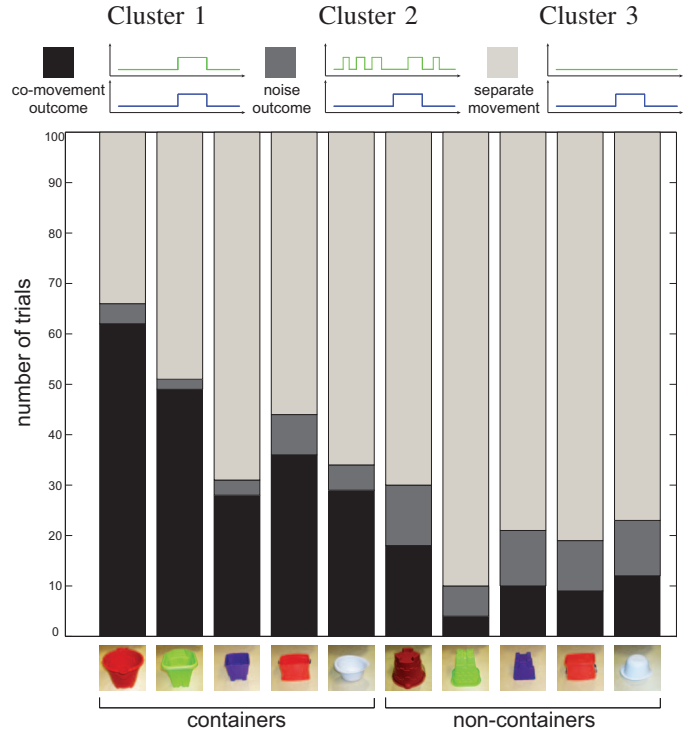


Fig. 7. The result of unsupervised clustering using X-means to categorize outcomes. X-means found three outcome classes: co-movement (black), separate movement (light gray), and cases of noise (dark gray). The co-movement outcome occurred more often with containers compared to non-containers. Movement duration and movement vector features were extracted from the robot’s detected movement data and used during the clustering procedure.

B. Discovering Object Categories

The result of unsupervised clustering using X-means to categorize objects resulted in two object categories: one cluster with the five containers (Fig. 2 a) and another cluster with the five non-containers (Fig. 2 b).

This result shows that a robot can successfully acquire an experience-grounded concept of containers. In other words, this grounded knowledge of containers could be verified at any time by re-probing the environment using the same sequence of interactions. But this also means that further experience with containers could enhance the robot’s container categorization ability.

The result also supports the claim that co-movement patterns can provide the robot with an “initial concept” [24] of containers when the interaction involved dropping a block from above and pushing the object. In this case, the functional properties of the objects were more salient than other variables that affected the outcome (e.g., size and shape).

C. Evaluation on Novel Objects

The robot was tested on how well it could detect the correct object category of 20 novel objects (see Fig. 8). The set of novel objects included 10 containers and 10 non-containers. Using the extracted visual features and the Nearest Neighbor classifier (see section IV.F), the robot was able to assign the correct object category to 19 out of 20 test objects. This

Novel Non-containers



Novel Containers



Fig. 8. The result of using a Nearest Neighbor classifier to label novel objects as ‘containers’ or ‘non-containers’. The flower pot (outlined in red) was the only misclassified object. Sparse coding features were extracted from the 10 training objects and used in the classification procedure.

implies that the robot not only has the ability to distinguish between the containers and non-containers that it interacts with, but it can also generalize its grounded representation of containers to novel objects that are only passively observed.

VI. CONCLUSION AND FUTURE WORK

This paper proposed a framework that a robot could use to successfully form simple object categories. The proposed approach is based on the principle that the robot should ground object categories in its own sensorimotor experience. The framework was tested on a container/non-container categorization task and performed well. First, the robot identified co-movement outcomes, separate movement outcomes, and noisy outcomes from the movement patterns of its interactions with objects. Second, the robot perfectly separated containers from non-containers using the pattern that co-movement outcomes occurred more often with containers than non-containers. Third, the robot used this separation to learn a perceptual model, which accurately detected the categories of 19 out of 20 novel objects.

These results demonstrate the feasibility of interaction-based approaches to object categorization. In other words, a robot can use interaction as a method to detect the functional categories of objects in its environment. Furthermore, a robot can also learn a perceptual model to detect the category of objects with which the robot has not interacted. Therefore, when the perceptual model is in question, the robot can interact with the object to determine the object category.

Numerous results in developmental psychology laid the groundwork for the framework presented in this paper. Future work should continue to build on this foundation by relaxing *several assumptions* at the center of this approach. An obvious extension would be to find methods of interaction-based object categorization that go beyond co-movement detection. Another interesting extension would be to modify the current framework so that the robot learns category-specific interactions

(e.g., dropping a block above an object and pushing the object) through imitation. We also plan to evaluate the approach presented in this paper in a richer environment with more objects, behaviors, and more categories of objects.

REFERENCES

- [1] L. Cohen, “Unresolved issues in infant categorization,” in *Early category and concept development*, D. Rakison and L. M. Oakes, Eds. New York: Oxford University Press, 2003, pp. 193–209.
- [2] P. Fitzpatrick, A. Needham, L. Natale, and G. Metta, “Shared challenges in object perception for robots and infants,” *Journal of Infant and Child Development*, vol. 17, no. 1, pp. 7–24, 2008.
- [3] M. Sutton, L. Stark, and K. Bowyer, “Gruff-3: generalizing the domain of a functional-based recognition system,” *Pattern Recognition*, vol. 27, no. 12, pp. 1743–1766, 1994.
- [4] R. Sutton, “Verification, the key to AI,” on-line essay. [Online]. Available: <http://www.cs.ualberta.ca/~sutton/IncIdeas/KeytoAI.html>
- [5] A. Stoytchev, “Five basic principles of developmental robotics,” in *NIPS 2006 Workshop on Grounding Perception, Knowledge and Cognition in Sensor-Motor Experience*, 2006.
- [6] E. S. Spelke and K. D. Kinzler, “Core knowledge,” *Developmental Science*, vol. 10, no. 1, pp. 89–96, 2007.
- [7] A. Needham, J. Cantlon, and S. O. Holley, “Infants’ use of category knowledge and object attributes when segregating objects at 8.5 months of age,” *Cog. Psychology*, vol. 53, no. 4, pp. 345–360, 2006.
- [8] S. Hespos and E. Spelke, “Precursors to spatial language: The case of containment,” *The Categorization of Spatial Entities in Language and Cognition*, vol. 15, pp. 48–144, 2007.
- [9] S. Hespos and R. Baillargeon, “Reasoning about containment events in very young infants,” *Cognition*, vol. 78, no. 3, pp. 207–245, 2001.
- [10] A. M. Leslie and P. DasGupta, “Infants’ understanding of a hidden mechanism: Invisible displacement,” sRCD Biennial Conf. Symp. on Infants’ reasoning about spatial relationships. Seattle, WA. Apr. 1991.
- [11] 3DV Systems. <http://www.3dvsystems.com/technology/product.html>.
- [12] R. Pfeifer and C. Scheier, “Sensory-motor coordination: The metaphor and beyond,” *Robotics and Autonomous Systems*, vol. 20, no. 2, pp. 157–178, 1997.
- [13] G. Metta and P. Fitzpatrick, “Early integration of vision and manipulation,” *Adaptive Behavior*, vol. 11, no. 2, pp. 109–128, June 2003.
- [14] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini, “Learning about objects through action - initial steps towards artificial cognition,” in *in Proc. of the 2003 IEEE Intl. Conf. on Robotics and Automation*, 2003, pp. 3140–3145.
- [15] M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini, “Developmental robotics: a survey,” *Connection Science*, vol. 15, no. 4, pp. 151–190, 2003.
- [16] J. Sinapov and A. Stoytchev, “Detecting the functional similarities between tools using a hierarchical representation of outcomes,” in *Proc. of the 7th IEEE Intl. Conf. on Development and Learning*, 2008.
- [17] J. Sinapov, M. Wiemer, and A. Stoytchev, “Interactive learning of the acoustic properties of household objects,” in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2009.
- [18] 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.
- [19] E. Ugur, M. R. Dogar, M. Cakmak, and E. Sahin, “The learning and use of traversability affordance using range images on a mobile robot,” in *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, 2007.
- [20] A. Edsinger and C. C. Kemp, “Two arms are better than one: A behavior-based control system for assistive bimanual manipulation,” in *Proc. of the 13th Intl. Conf. on Advanced Robotics*, 2007.
- [21] D. Pelleg and A. Moore, “X-means: Extending k-means with efficient estimation of the number of clusters,” in *Proc. of the 17th Intl. Conf. on Machine Learning*, 2000, pp. 727–734.
- [22] B. A. Olshausen and D. J. Field, “Emergence of simple-cell receptive field properties by learning a sparse code of natural images,” *Nature*, vol. 381, pp. 607–609, 1996.
- [23] H. Lee, A. Battle, R. Raina, and A. Y. Ng., “Efficient sparse coding algorithms,” in *in Proc. of NIPS*, 2007, pp. 801–888.
- [24] R. Baillargeon, “How do infants learn about the physical world?” *Current Directions in Psychological Science*, vol. 3, no. 5, pp. 133–140, 1994.