In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI),
New York City, NY USA July 2016

Learning to Order Objects Using Haptic and Proprioceptive Exploratory Behaviors

Jivko Sinapov, Priyanka Khante, Maxwell Svetlik and Peter Stone
Department of Computer Science
University of Texas at Austin
{jsinapov,pkhante,maxwell,pstone}@cs.utexas.edu

Abstract

This paper proposes a novel framework that enables a robot to learn ordinal object relations. While most related work focuses on classifying objects into discrete categories, such approaches cannot learn object properties (e.g., weight, height, size, etc.) that are context-specific and relative to other objects. To address this problem, we propose that a robot should learn to order objects based on ordinal object relations. In our experiments, the robot explored a set of 32 objects that can be ordered by three properties: height, weight, and width. Next, the robot used unsupervised learning to discover multiple ways that the objects can be ordered based on the haptic and proprioceptive perceptions detected while exploring the objects. Following, the robot's model was presented with labeled object series, allowing it to ground the three ordinal relations in terms of how similar they are to the orders discovered during the unsupervised stage. Finally, the grounded models were used to recognize whether new object series were ordered by any of the three properties as well as to correctly insert additional objects into an existing series.

1 INTRODUCTION

The ability to order physical objects by various properties emerges early in childhood development and is thought to be fundamental for understanding the property of numbers [Kingma and Reuvekamp, 1984]. In the Montessori method of teaching [Montessori, 1917], children solve ordering tasks using specialized sets of toys that allow the child to learn the target dimension along which the objects should be ordered [Pitamic, 2004]. Ordering tasks also appear in intelligence tests [Hagmann-von Arx *et al.*, 2008] which suggests that ability to order objects is an important aspect of human intelligence.

In cognitive robotics, there have been relatively few works exploring how a robot can learn to order a set of objects. Instead, most related work has focused on the problem of object categorization, i.e., assigning an object to 1 or more discrete categories (see [Sanchez-Fibla *et al.*, 2013; Orhan *et al.*, 2013; Yürüten *et al.*, 2013; Sinapov *et al.*, 2014a; Chu *et al.*, 2015] for a representative sample). Such methods have shown that using behaviors in conjunction with visual features allows a robot to ground the meaning of nouns and adjectives. However, many object properties cannot be represented well using discrete categories. For example, "height" is better represented by an ordering rather than a discrete clustering as the ordering captures the continuous nature of the property while the clustering does not.

Figure 1: The robot used in our experiments, shown here with the 32 objects it explored in order to learn three ordinal properties: *weight*, *width*, and *height*.

To address this gap, we propose a framework for learning object ordering skills that are grounded in a robot's physical interactions with objects. In our method, the robot uses unsupervised learning to discover how objects can be ordered using multiple and different types of sensorimotor features that the robot detects during object exploration. Next, the robot undergoes a supervised learning stage during which it is trained that specific example object orders are associated with specific object properties, namely "height", "width", and "weight". More specifically, the robot learns how the three ordinal concepts relate to the object orders discovered in the previous stage. The learned model is subsequently used by the robot to recognize the property according to which a new series of objects is ordered by. Finally, the robot uses this recognition ability to correctly insert additional objects into existing object series.

2 Related Work

2.1 Psychology and Cognitive Science

Ordering objects emerges early in childhood – by age 2, children can compare and sort objects according to size [Graham *et al.*, 1964]. Other studies have shown that learning to classify objects as "big" or "small" is easier when the object is directly comparable to other objects in its surroundings as opposed to using only its absolute size (see [Ebeling and Gelman, 1994; 1988]). Similarly, comparing objects to each other helps children learn to order objects according to their height [Smith *et al.*, 1986].

In addition to supervised learning of relative concepts, some studies have also looked at whether children form order representations in an unsupervised manner. The experiments described in [Sugarman, 1981] found that the order of objects that children explore freely is influenced by how perceptually similar the objects are. For example, if the child is exploring a small toy, it is more likely to switch to a slightly bigger one as opposed to one that is much bigger. This result suggests that children can order objects in an unsupervised manner and can extract certain natural orders from interaction with objects.

Based on these findings, the framework proposed in this paper uses both unsupervised and supervised learning. First, the robot uses an unsupervised approach to detect a range of possible orders for a given set of objects. This is followed by a supervised learning stage in which the robot grounds the concepts "weight", "height", and "width" in terms of the orders discovered in the unsupervised stage.

2.2 Robotics and AI

Grounding concepts related to object ordering has received very little attention in robotics so far. In machine learning, ranking [Liu, 2009] is a related problem in which items (e.g., search results) need to be ranked according to some criteria (e.g., user's preferences and a search query). While such methods could be adapted to object ordering problems, they typically assume that there is a lot of training data and only learn one ranking relation at a time using a single flat feature vector representation. In addition, most methods focus on getting the top few results correct, i.e., the cost of mistakes depends on the position of the ranking.

In cognitive robotics, most related work focuses on the problem of grounding discrete object categories in visual features (e.g., [Gorbenko and Popov, 2012; Lai *et al.*, 2011]) as well as non-visual sensory modalities coupled with manipulative behaviors (e.g., [Hogman *et al.*, 2013; Orhan *et al.*, 2013; Yürüten *et al.*, 2013; Nakamura *et al.*, 2014; Sinapov *et al.*, 2014a; Celikkanat *et al.*, 2015; Chu *et al.*, 2015]). Using such methods a robot can ground the meaning of nouns (e.g., "pop can", "ball", etc.) as well as adjectives (e.g., "red", "round", etc.). These methods, however, cannot be used to ground ordering concepts as they only deal with the problem of classifying an individual object into a discrete set of categories.

In a closely related study, the method described in [Sinapov *et al.*, 2014b] relaxed the assumption that categories describe only individual objects and showed that through behavioral exploration a robot can learn pairwise object categories (e.g., A is heavier than B). However, the robot stopped short of recognizing that a given series of objects is ordered by a given property. A rare example of a robotics study directly dealing with object ordering is described in [Schenck *et al.*, 2012]. The robot in that study was able to solve the task of order completion (i.e., selecting the next object to complete a sequence) using both supervised and unsupervised approaches through the use of an objective function based on perceptual object similarity. In their work, the robot learned each ordering concept in isolation and did not explicitly ground those concepts (i.e., it could not explicitly recognize that a series of objects is ordered by weight, or height). In addition, the objects were specialized toys that varied only according to one dimension, as opposed to real-world objects.

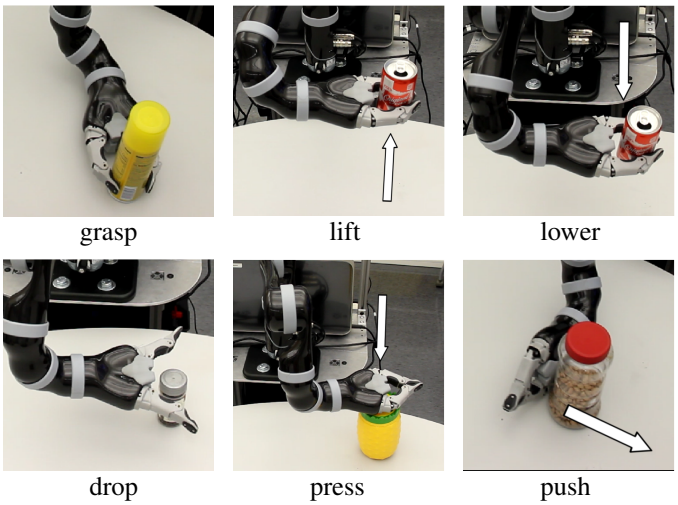


Figure 2: The behaviors the robot used to explored the objects. From left to right and top to bottom: *grasp*, *lift*, *lower*, *drop*, *press*, and *push*. The arrows indicate the direction of motion of the end-effector for each behavior. In addition, the *hold* behavior (not shown) was performed after the *lift* behavior by simply holding the object in place for half a second.

3 Experimental Setup

3.1 Robot and Objects

The robot used in our experiment, shown in Figure 1, was a custom built mobile manipulator that uses the Segway Robotic Mobility Platform (RMP). The robot is equipped with a 6-DOF Kinova Mico Arm with a 2-fingered underactuated gripper as its end effector.

The set of objects $\mathcal{O}$ that the robot explored consisted of 32 common household items including cups, bottles, cans, and other containers, also shown in Figure 1. The objects varied according to their *weight*, *height*, and *width*. The objects' height and width was measured in millimeters while their weight was measured in grams. Some of the bottles were filled with water up to various levels. The objects were chosen such that the distributions of their weight, width, and height were roughly uniform.

3.2 Exploratory Behaviors and Sensory Modalities

The robot explored the objects using 7 different behaviors: *grasp*, *lift*, *hold*, *lower*, *drop*, *push*, and *press*, shown in Figure 2. The behaviors were designed with the purpose of enabling the robot to obtain a behavior-grounded multi-modal object representation independent of any particular task. While some of the behaviors are likely to be irrelevant for any particular ordering task, we assume that this information is not known to the robot in advance. The behaviors were performed assuming a fixed object location on the table and were encoded using joint-space trajectories as well as timed Cartesian velocity commands (for *lift*, *lower*, *push*, and *press*).

The robot perceived the objects using the *haptic* (i.e., joint torques) and *proprioceptive* (i.e., joint angular positions) sensory modalities. During the execution of each behavior, the robot recorded the torque and position values for all 6 joints at 15 Hz resulting in two $n \times 6$ matrices where n was the number of samples recorded. To reduce dimensionality, the temporal axis of each matrix was discretized into 10 equally spaced bins using the methodology of [Sinapov *et al.*, 2014a]. Thus, after executing an action on an object, the robot extracted two sensorimotor feature vectors, $\mathbf{x}_{haptic} \in \mathbb{R}^{10\times 6}$ and $\mathbf{x}_{proprioception} \in \mathbb{R}^{10\times 6}$. In addition, at the end of the grasp behavior, the robot also recorded the end finger positions, i.e., $\mathbf{x}_{fingers} \in \mathbb{R}^2$. Each viable combination of a behavior (one of the 7) and modality (either *haptic*, *proprioception*, or *fingers*) constituted a sensorimotor context. Thus, the set of sensorimotor contexts, $\mathcal{C}$, was of size $7\times 2+1 = 15$ (the *fingers* modality was only available for the *grasp* behavior).

The robot performed each behavior on each object 5 different times, which took 7.5 hours. Given context $c \in \mathcal{C}$ and object $i \in \mathcal{O}$, the set $\mathcal{X}_i^c$ contained all 5 feature vectors observed with object i in context c. Following, we describe the learning framework which uses these observations to ground the three ordinal relations, *weight*, *width*, and *height*.

4 Learning Framework

4.1 Notation and Problem Formulation

Let $\mathcal{O}$ be the full set of objects that the robot has explored. Let $S = (\mathcal{O}_S, G_S)$ be an object series defined for objects $\mathcal{O}_S \subset \mathcal{O}$ using a directed cluster graph $G_S = (V, E)$. Each object in $\mathcal{O}_S$ belongs to one of several clusters in V, which are connected using directed edges E. The set of edges E is constrained such that one cluster has just an outgoing edge (i.e., the start of the order), one cluster has only one incoming edge (i.e., the end of the order) while the rest have one of each. Figure 3.a) shows an example cluster graph with 5 clusters V and 10 objects $\mathcal{O}_S$.

Let $\mathcal{L}$ be the set of ordering concepts: in our case, {*weight*, *width*, *height*}. The robot is tasked with solving the following two problems:

Order Recognition

Given an example object series S, for each concept $l \in \mathcal{L}$, the task of the robot is to learn a model M_l such that $M_l(S) \rightarrow +1$ if S is ordered according to l and -1 otherwise.

Object Insertion

Given a series S and object i, the task is to construct a new series S' by inserting i at the correct position. In other words, S' should be ordered according to the same concept l that was used to construct the order S.

To solve these problems, the robot undergoes three distinct stages described in detail below.

4.2 Interaction Stage

During this stage, the robot explores the objects in $\mathcal{O}$ by performing a series of exploratory behaviors on them and recording sensorimotor features capturing multiple sensory modalities. Let $\mathcal{C}$ be the set of sensorimotor contexts (defined in the previous section) where each context corresponds to a combination of an exploratory behavior and a sensory modality. For each context $c \in \mathcal{C}$ and object $i \in \mathcal{O}$, let $\mathcal{X}_i^c$ be the set of sensorimotor features observed with object i in context c.

Let $R^c \in \mathbb{R}^{|\mathcal{O}|\times|\mathcal{O}|}$ be a matrix that specifies a pairwise object similarity relation in context c for each pair of objects, computed as follows. Given a sensorimotor feature vector $x_i^c \in \mathcal{X}_i^c$ detected with object i, let the function $knn\_count(x_i^c, j, k)$ return the count of k nearest neighbors to x_i^c detected with object j. Each entry r_{ij}^c in the matrix R^c was computed as:

$$r_{ij}^c = \sum_{x_i^c \in \mathcal{X}_i^c} knn\_count(x_i^c, j, k)$$

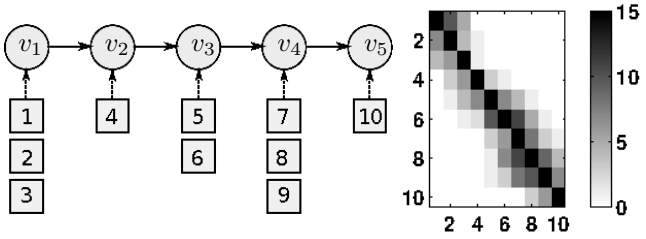
In other words, each entry r_{ij}^c was set to the number of times a sensory signal $x_j^c \in \mathcal{X}_j^c$ was one of the k nearest neighbors of a sensory signal $x_i^c \in \mathcal{X}_i^c$. In our experiments, k was set to 25 but the results were similar for the range of 5 to 50 (k can be larger than the number of objects 32, as the matrix is computed using the raw observations, for which there are 5 per object). Following, we describe how this object representation is used by the robot to order objects in an unsupervised manner.

4.3 Unsupervised Order Discovery

In the second stage, the robot uses unsupervised learning to discover multiple possible ways to order the objects in $\mathcal{O}$. To do so, for each sensory motor context $c \in \mathcal{C}$, the matrix R^c is used to fit an order S_c using the methodology described in [Kemp and Tenenbaum, 2008]. More specifically, given an input matrix R^c, the method in [Kemp and Tenenbaum, 2008] searches for the cluster graph that maximizes the posterior probability of the data given the cluster graph. This probability will be high if features in the data vary smoothly over the graph and low otherwise.

Figure 3 shows an example order fitting using a synthetically generated 10×10 matrix. Due to space limitations, we refer the reader to [Kemp and Tenenbaum, 2008] for further details. In our experiments, we used the source code provided with that publication with default prior likelihood parameters.

At the end of this stage, for each context $c \in \mathcal{C}$, let S_c be the order that maximizes $Pr(R^c|S_c)$. In other words, at this stage the robot has discovered a set of object orders, one per sensorimotor context, which will subsequently be used to ground ordinal object relations.



a) b)

Figure 3: Example unsupervised object ordering using synthetic data. a) The object series $S = (\mathcal{O}_S, G_S)$. The circles denote the vertices in G_S while the boxes denote the set of objects $\mathcal{O}_S$ for which the series is defined. The dotted arrows denote cluster memberships while the bold arrows represent the set of edges E in the cluster graph G_S. b) The input matrix $R \in \mathbb{R}^{10\times 10}$ used to compute the ordering in a).

4.4 Order Grounding Stage

In the third and final stage, the robot learns an order recognition model M_l for each ordering concept $l \in \mathcal{L}$ in a supervised manner. Let $D_l = (S_n, y_n)_{n=1}^N$ be a set of examples such that each S_n is an object order and each $y_n \in \{-1, +1\}$ is a class label indicating whether the objects are ordered according to l. To learn a classifier from such data, the robot represents each example according to its similarity to the orders discovered in the previous stage.

More specifically, given an example order S_n, let S_n^c be the order constructed by arranging the objects $\mathcal{O}_{S_n}$ according to the order S_c associated with context c. The series S_n^c is not necessarily a total order as two or more objects may belong to the same cluster node in the graph G_c. Let $h_S(i,j) \in \mathbb{I}$ be the relative difference in position between objects i and j in order S. For example, if i occupies one slot above j, then $h_S(i,j) = 1$; on the other hand, if i and j occupy the same cluster node, then the value is 0. Given two orders, S_n and S_n^c containing the same set of objects $\mathcal{O}_S$, we can then define a distance function between them as:

$$d(S_n, S_n^c) = \sum_{i\in\mathcal{O}_S}\sum_{j\in\mathcal{O}_S} \left|\frac{h_{S_n}(i,j)}{length(S_n)} - \frac{h_{S_n^c}(i,j)}{length(S_n^c)}\right|$$

where $length(S)$ returns the number of clusters in the cluster graph G_S of series S. Thus, a training example S_n can be represented by a feature vector $\mathbf{x}_n \in \mathbb{R}^{|\mathcal{C}|}$ such that each element $x_{n,c} = d(S_n, S_n^c)$. This feature vector encodes how similar S_n is to each of the orders that the robot discovered using unsupervised learning. Given this feature representation, for each dataset $D_l = (S_n, y_n)_{n=1}^N$, the robot learns a classifier that can output an estimate for $Pr(y_n = +1|\mathbf{x}_n)$.

The robot used the learned classifiers for order recognition to solve the object order insertion task as follows. Given an existing order S and an additional object i, let $\mathcal{S}'$ denote the set of orders attained by inserting i in all possible slots. The robot then inserts object i into the series S resulting in the object series S^* such that:

$$S^* = \arg\max_{S'\in\mathcal{S}'} \max_l Pr_l(y' = +1|\mathbf{x}')$$

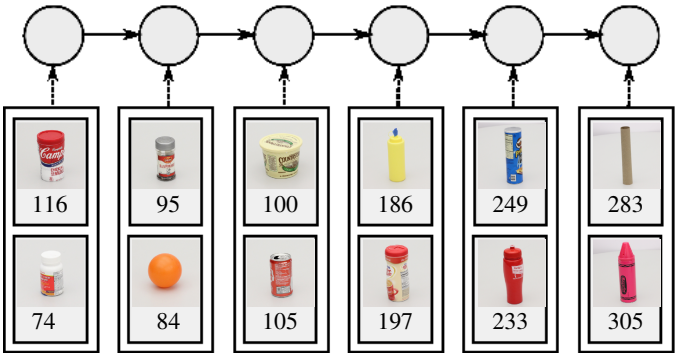


Figure 4: Example unsupervised object ordering estimated by the robot using sensorimotor observations in the *press-haptic* context. While the object ordering was estimated over all 32 objects, only 12 objects are shown here to prevent clutter. The number underneath each object denotes the object's height in millimeters. This specific combination of a behavior and sensory modality approximately orders the objects by height but due to perceptual noise, it is not perfect (e.g., the object with height of 116 mm is inserted into the first position rather than the third). The height averages for each of the six positions in the ordering were 126, 90, 102, 196, 245, and 248 mm.

where $\mathbf{x}'$ is the feature vector for the order $S' \in \mathcal{S}'$. In other words, the model picks the series that maximizes the output of one of the order recognition models M_l.

4.5 Evaluation

The framework is evaluated in a series of evaluation runs. During each run, for each object ordering concept $l \in \mathcal{L}$, a data set $D_l = (S_n, y_n)_{n=1}^{100}$ is constructed such that half the series are positive examples (i.e., ordered according to l) and the other half, negative. An object series was considered a positive example if and only if the change in the target attribute l from one position to the next exceeded a threshold θ_l (possibly set to 0.0). An object series was considered a negative example if the direction of the change along attribute l varied from position to position. The robot's order recognition models were then evaluated by performing 5-fold cross validation on D_l. The results were averaged over 100 evaluation runs with different random seeds used to generate the data sets D_l. The results are reported in terms of percentage recognition accuracy, i.e., $\frac{\text{\# of correct series classifications}}{\text{\# of total classifications}} \times 100$.

To evaluate the robot's ability to correctly insert an object into an existing order, for each ordering concept $l \in \mathcal{L}$, an additional set of object series D'_l was computed such that $D'_l \bigcap D_l = \emptyset$. Each series in D'_l consisted of 5 objects. For each series, one object was randomly removed and the robot's model was then tasked to insert that object into the order consisting of the remaining 4 objects. Let $\hat{p}$ be the index of position for the remainder object chosen by the robot and let p be the index of the correct position for that object in the original series. The error can therefore be computed by taking the absolute difference, i.e., $|\hat{p} - p|$.

Table 1: Order Recognition Rates (% Accuracy)

concept	k-NN	SVM	Decision Tree
weight	89.48	92.42	96.67
width	78.82	82.49	91.70
height	86.44	90.18	98.42

5 Results

5.1 Example Unsupervised Order Discovery

Figure 4 shows an example object order that was estimated by the robot during the unsupervised order discovery stage. In this case, the order was constructed using sensorimotor observations in the haptic sensory modality during the execution of the *press* behavior. Some of the objects are placed in the same position in the order, which could be due to perceptual noise (only 5 observations are available for each object) or because the differences in the height attribute for objects in the same cluster are too small to be detected by this particular behavior and sensory modality. The order reveals that this particular combination of a behavior and sensory modality induces an ordering based on the objects' *height* attribute. Similarly, other behaviors and modalities induced orderings that corresponded to the other two attributes considered in this study, *width*, and *weight*.

5.2 Object Order Recognition

Table 1 shows the accuracy of the robot's order recognition models using three different machine learning algorithms, k-Nearest Neighbor, Support Vector Machine with a quadratic kernel function, and C4.5 Decision Tree as implemented in the WEKA machine learning library [Witten and Frank, 2005]. For this test, the thresholds θ_l were set to 50 grams for *weight*, 15 mm for *width*, and 50 mm for *height*. In other words, an object series was considered to be ordered by height if each consecutive object was at least 50 mm taller than the previous one. All object series used for training and testing were of length 5 and were automatically generated by randomly sampling positive and negative examples from the set of all possible orders.

The C4.5 Decision Tree classifier achieves the highest accuracy and naturally performs feature selection. Since each feature is associated with a sensorimotor context, a learned tree identifies the relevant behaviors and sensory modalities for a given ordering task. For example, the tree learned for ordering objects by weight relied on features corresponding to three different behaviors: *hold*, *lift*, and *lower*. For approximately 75% of the test data points, the decision was made using only the top level node, corresponding to the *hold* behavior and the *haptic* sensory modality. On the other hand, the tree learned for ordering objects by width relied exclusively on the feature associated with the *grasp* behavior and the *fingers* position sensory signal. Therefore, once learned, the models could in principle be used by performing only the discovered relevant behaviors for a given ordering task instead of exhaustively performing the entire set of actions.

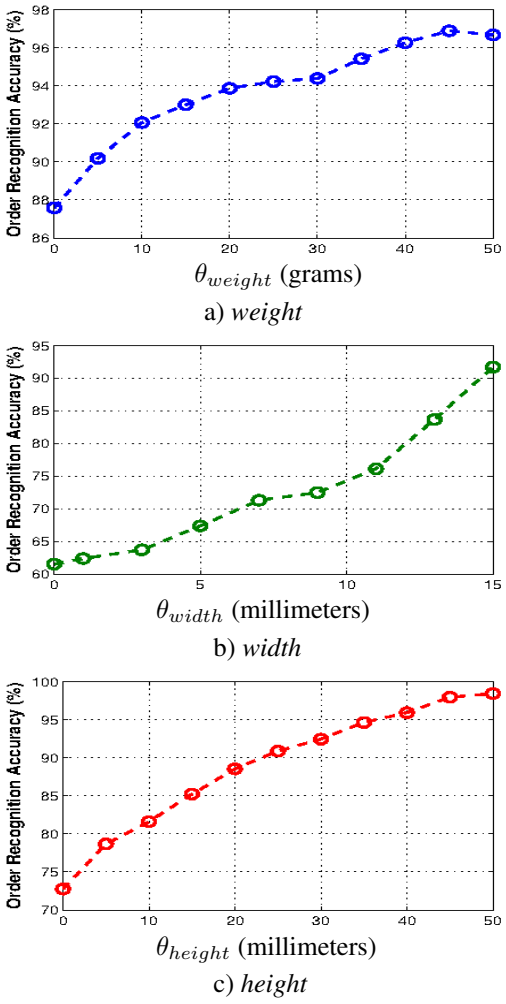
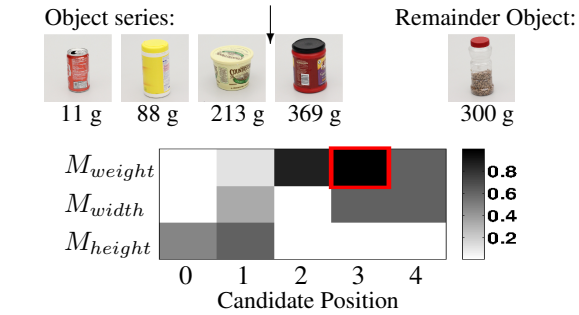


Figure 5: Order Recognition performance as a function of the thresholds θ_l. For instance, an object series was considered to be ordered by height if each consecutive object was at least θ_{height} mm taller than the previous one.

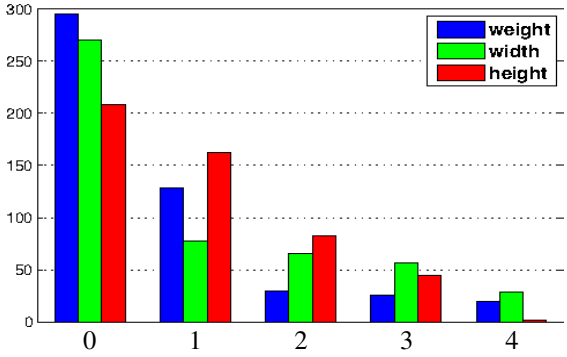
5.3 Sensitivity to Thresholds θ_l

In the next test, we evaluate the recognition performance with the Decision Tree classifier as the thresholds θ_l change from low to high. When these thresholds are high, series that constitute positive examples (i.e., ordered by the target property) tend to have larger differences in the target attribute from one position to the next in the series. Conversely, when these thresholds are low, the differences in the target attribute from one position to another in a given series are smaller and may even be undetectable by the specific behaviors and sensory modalities used in the experiment.

The result of this test are shown in Figure 5. As we expect, recognition accuracy is higher when the attribute increases by a larger value from one position to the next in an object series. This is largely due to perceptual limitations of the specific robot that was used in our experiments. Nevertheless, even when the threshold is set to 0 (i.e., any strictly monotonically increasing series is considered a positive example), the robot



a) An Example Object Insertion Task



b) Histogram of Object Insertion Errors

Figure 6: a) An example order insertion task. In this task, the robot is presented with an object series of length 4 that is ordered by weight and then tasked with inserting an additional remainder object into one of 5 possible positions. The number beneath each object denotes its weight in grams. The matrix visualizes the output of the 3 order recognition models for each candidate position where the object can be placed in. The position and classifier which maximize the output is labeled with red. The object is correctly inserting in position 3, denoted by the arrow. b) A histogram of the errors made by the robot when solving the object insertion task. The error corresponds to the absolute difference between the position chosen by the robot and the correct position. An error of 0 indicates a correct solution.

is still able to achieve recognition rates substantially better than chance. Recognizing series ordered by *width* turns out to be the hardest of the tasks as the only sensorimotor features relevant to this property were the end finger positions after executing the *grasp* behavior. The arm's fingers are underactuated and the readings are quite noisy which made this property particularly hard for the robot to learn.

5.4 Object Insertion

Finally, we evaluate the robot's ability to correctly insert an object into an existing series. Figure 6.a) shows an example insertion task in which the robot's model has to insert an object into an existing object series which is ordered by weight. The matrix visualizes the output of the 3 order recognition models when given the series constructed by inserting the object into each possible position. The output is maximized at position 3 (i.e., left of the last object in the series) for the *weight* ordering concept. In this example, the object in inserted into the correct slot.

For each of the three ordinal properties, the robot was tested on 500 insertion tasks. Figure 6 shows a histogram of the errors made by the robot when using the Decision Tree classifier. The difference between the correct position and the one chosen by the robot was either 1 or 0 in 88% of the tests for weight, 70% for width, and 73% for height. The task of the robot was particularly difficult as some of the example series in this test could be ordered by more than just one property due to chance. Nevertheless, this result shows that the robot can use its order recognition models to not only recognize the property that an object set is ordered by, but also to add additional objects to the existing series.

6 Conclusion and Future Work

This paper proposed a novel framework that allows a robot to learn object ordering skills. The framework was evaluated in an experiment in which a robot learned to order common household objects using three different attributes: weight, width, and height. Based on findings in psychology and cognitive science, the framework described in this paper proposed that a robot can ground ordinal object relations in terms of object orderings discovered in an unsupervised manner. The results of our experiment showed that the framework allows a robot to learn to recognize whether a given series of objects is ordered according to a target property. In addition, the robot's recognition model was also used to successfully insert objects into an existing object series, a task that is common in human intelligence tests.

There are several direct lines for future work. A limitation of the current study is that once the objects were explored by the robot, the resulting tests were performed offline. In ongoing work, we are investigating methods that would enable the robot to efficiently explore novel objects and incrementally update the learned representations of ordinal object relations. More generally, as described in [Kemp and Tenenbaum, 2008], in addition to discrete categories (which are widely explored in robotics) and orders, object relations can induce a variety of other structural forms such as trees, rings, hierarchies, etc. Therefore, we are also pursuing a more general framework that would allow a robot to ground semantic concepts from language using multiple and heterogeneous structural forms. Finally, we are actively pursuing frameworks for grounding object-related knowledge through human-robot interaction scenarios such as the "I Spy" game described in [Thomason *et al.*, 2016].

Acknowledgments

This work has taken place in the Learning Agents Research Group (LARG) at the Artificial Intelligence Laboratory, The University of Texas at Austin. LARG research is supported in part by grants from the National Science Foundation (CNS-1330072, CNS-1305287), ONR (21C184-01), AFRL (FA8750-14-1-0070), and AFOSR (FA9550-14-1-0087).

References

[Celikkanat *et al.*, 2015] H. Celikkanat, G. Orhan, and S. Kalkan. A probabilistic concept web on a humanoid robot. *Autonomous Mental Development, IEEE Transactions on*, 7(2):92–106, June 2015.

[Chu *et al.*, 2015] Vivian Chu, Ian McMahon, Lorenzo Riano, Craig G McDonald, Qin He, Jorge Martinez Perez-Tejada, Michael Arrigo, Trevor Darrell, and Katherine J Kuchenbecker. Robotic learning of haptic adjectives through physical interaction. *Robotics and Autonomous Systems*, 63:279–292, 2015.

[Ebeling and Gelman, 1988] Karen S Ebeling and Susan A Gelman. Coordination of size standards by young children. *Child Development*, pages 888–896, 1988.

[Ebeling and Gelman, 1994] Karen S Ebeling and Susan A Gelman. Children's use of context in interpreting big and little. *Child Development*, 65(4):1178–1192, 1994.

[Gorbenko and Popov, 2012] Anna Gorbenko and Vladimir Popov. Self-learning algorithm for visual recognition and object categorization for autonomous mobile robots. In *Computer, Informatics, Cybernetics and Applications*, pages 1289–1295. Springer, 2012.

[Graham *et al.*, 1964] Frances K Graham, Claire B Ernhart, Marguerite Craft, and Phyllis W Berman. Learning of relative and absolute size concepts in preschool children. *Journal of Experimental Child Psychology*, 1(1):26–36, 1964.

[Hagmann-von Arx *et al.*, 2008] Priska Hagmann-von Arx, Christine Sandra Meyer, and Alexander Grob. Assessing intellectual giftedness with the WISC-IV and the IDS. *Zeitschrift für Psychologie/Journal of Psychology*, 216(3):172–179, 2008.

[Hogman *et al.*, 2013] Virgile Hogman, Mats Bjorkman, and Danica Kragic. Interactive object classification using sensorimotor contingencies. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 2799–2805. IEEE, 2013.

[Kemp and Tenenbaum, 2008] Charles Kemp and Joshua B Tenenbaum. The discovery of structural form. *Proceedings of the National Academy of Sciences*, 105(31):10687–10692, 2008.

[Kingma and Reuvekamp, 1984] Johannes Kingma and Johan Reuvekamp. The construction of a developmental scale for seriation. *Educational and Psychological measurement*, 44(1):1–23, 1984.

[Lai *et al.*, 2011] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE, 2011.

[Liu, 2009] Tie-Yan Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331, 2009.

[Montessori, 1917] Maria Montessori. *The Advanced Montessori Method.* Frederick A. Stokes Company, 1917.

[Nakamura *et al.*, 2014] Tomoaki Nakamura, Takayuki Nagai, Kotaro Funakoshi, Shogo Nagasaka, Takafumi Taniguchi, and Naoto Iwahashi. Mutual learning of an object concept and language model based on MLDA and NPYLM. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 600–607. IEEE, 2014.

[Orhan *et al.*, 2013] Guner Orhan, Sertac Olgunsoylu, Erol Sahin, and Sinan Kalkan. Co-learning nouns and adjectives. In *Development and Learning and Epigenetic Robotics (ICDL), 2013 IEEE Third Joint International Conference on*, pages 1–6. IEEE, 2013.

[Pitamic, 2004] Maja Pitamic. *Teach Me to Do it Myself: Montessori Activities for You and Your Child*. Barron's Educational Series, 2004.

[Sanchez-Fibla *et al.*, 2013] Marti Sanchez-Fibla, Armin Duff, and Paul FMJ Verschure. A sensorimotor account of visual and tactile integration for object categorization and grasping. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 107–112. IEEE, 2013.

[Schenck *et al.*, 2012] C Schenck, J Sinapov, and A Stoytchev. Which object comes next? grounded order completion by a humanoid robot. *Cybernetics and Information Technologies*, 12(3):5–16, 2012.

[Sinapov *et al.*, 2014a] Jivko Sinapov, Connor Schenck, Kerrick Staley, Vladimir Sukhoy, and Alexander Stoytchev. Grounding semantic categories in behavioral interactions: Experiments with 100 objects. *Robotics and Autonomous Systems*, 62(5):632–645, 2014.

[Sinapov *et al.*, 2014b] Jivko Sinapov, Connor Schenck, and Alexander Stoytchev. Learning relational object categories using behavioral exploration and multimodal perception. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 5691–5698. IEEE, 2014.

[Smith *et al.*, 1986] L.B. Smith, N.J. Cooney, and C. McCord. What Is" High"? The Development of Reference Points for" High" and" Low". *Child Development*, pages 583–602, 1986.

[Sugarman, 1981] Susan Sugarman. The cognitive basis of classification in very young children: An analysis of object-ordering trends. *Child Development*, pages 1172–1178, 1981.

[Thomason *et al.*, 2016] Jesse Thomason, Jivko Sinapov, Maxwell Svetlik, Raymond Mooney, and Peter Stone. Learning multi-modal grounded linguistic semantics by playing i, spy. In *Proceedings of the Twenty-Fifth international joint conference on Artificial Intelligence (IJCAI)*, 2016.

[Witten and Frank, 2005] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufman, San Francisco, 2nd edition, 2005.

[Yürüten *et al.*, 2013] Onur Yürüten, Erol Şahin, and Sinan Kalkan. The learning of adjectives and nouns from affordance and appearance features. *Adaptive Behavior*, 21(6):437–451, 2013.