Grounded Object Individuation by a Humanoid Robot

Jivko Sinapov and Alexander Stoytchev
Developmental Robotics Laboratory
Iowa State University
{jsinapov, alexs}@iastate.edu

***Abstract*—This paper proposes a theoretical model that enables a robot to partition its unlabeled sensorimotor experience with different objects into discrete clusters, each corresponding to a specific object. To solve this object individuation problem, the robot was trained to detect whether two perceptual stimuli were produced by the same object or by two different objects. The model was tested using a large-scale experiment in which a humanoid robot explored 100 different objects by performing a variety of exploratory behaviors on them and detecting the resulting sensory feedback from several sensory modalities. The results show that with a small amount of prior training, the robot's model was able to successfully individuate the objects with a high degree of accuracy.**

I. Introduction

Humans learn to individuate objects by first learning to detect whether two perceptual stimuli were produced by the same object or by two different objects [1]. This ability allows humans to infer how many unique objects they have observed and to establish an object representation that can be used to map individual experiences with an object to a unique object identifier [2]. Studies in developmental psychology have shown that this skill is fundamental to establishing an internal object representation that can handle the large number of objects that humans encounter in their daily lives [1], [3].

In contrast, most methods used by robots to recognize objects start with a fixed object representation in which the robot's training data is labeled with one of a finite number of object identities (see [4], [5], [6], [7], [8], [9], [10], [11] for a representative sample of such approaches). These methods implicitly make the assumption that the object individuation task has already been solved. In other words, training the robot's object recognition models requires that the training observations are labeled with the correct object identity. Providing labeled data, however, becomes increasingly more difficult as the number of objects increases. Furthermore, an autonomous robot operating in human environments is bound to encounter new objects that were not in its training dataset. Therefore, in addition to recognizing objects, robots must also be able to individuate novel objects.

To address these challenges, this paper proposes a behavior-grounded approach to object individuation that enables a robot to estimate how many objects it has interacted with, and group its sensorimotor experience with objects according to the estimated object identities. The method was tested using a large-scale experiment in which the robot interacted with 100 different objects using 10 different exploratory behaviors. The results demonstrate that by using a small amount of prior training, the model can successfully individuate novel objects that were not present in the robot's training set.

Fig. 1. The humanoid robot used in our experiments, along with the 100 objects that it explored.

II. Related Work

A. Psychology

When psychologists study how humans individuate and identify objects they typically use an experimental design in which the participant is presented with a sequence of objects and at the end is asked to infer how many unique objects were encountered [2]. In this setting, the subject cannot observe multiple objects at the same time, and thus must rely on the objects' perceptual features when solving the task. The results of the experiment conducted by Kemp *et al.* [2] show that prior experience with objects with known object identities is necessary in order to solve the object individuation task on a novel set of objects.

Therefore, it is not surprising that humans use a variety of cues, other than object features, when individuating objects [2], [1]. For example, spatial cues can be used to individuate objects since observing two objects next to each other indicates that the two objects are not the same [12]. Humans also use temporal cues, e.g., they assume that an object would remain the same object over the course of contiguous manipulation or observation [13]. Most importantly, such spatial and temporal cues can inform the observer that the featural differences

between the objects are not due to noisy observations, but due to the two objects being different [2], [12].

Inspired by these results from psychology, this paper proposes a learning approach to object individuation in which the robot was initially trained to detect whether two sensorimotor experiences are produced by the same object or by two different objects. Subsequently, the trained model was used to partition the robot's sensorimotor experience with novel objects in order to individuate them. The results of our experiments suggest that, just as for humans, prior information, in the form of a training set with known object identities, is necessary for solving this problem.

B. Robotics

Object individuation has received relatively little attention in robotics. In contrast, a wide variety of methods have been developed that allow robots to recognize previously observed objects. The majority of these methods use 2D and 3D visual features (see [14], [15], [7], [9], [11]). Other vision-based approaches have also been proposed for finding image regions from multiple views that contain the same object [16]. In addition, experiments have demonstrated that robots can also recognize objects and their categories using proprioceptive [6], [8], auditory [4], [5], tactile [17], [18] and multi-modal [19], [10], [20] sensory feedback. The main limitation of these systems is that the object recognition models can only be trained on fixed datasets containing labeled data for all objects that the robot may encounter. In other words, while such systems can recognize previously observed objects, they cannot individuate novel objects that they encounter after training time.

It is worth noting that this limitation does not only plague object recognition methods, but also affects a variety of other robotic systems. For example, to learn the affordances of a tool, the methods described in [21] and [22] assume that the robot's sensorimotor data is cleanly partitioned according to the identity of each tool. Similarly, when categorizing objects as either containers or non-containers, the robot in [23] started with the implicit assumption that it already knows the identities of all objects that it has to interact with. These and many other examples show that today's robots typically start with fixed object representations, and thus lack the ability to individuate objects that they may encounter in the future.

III. Experimental Methodology

A. Robot

The upper-torso humanoid robot used in our experiments (shown in Fig. 1) has two 7-DOF Barrett Whole Arm Manipulators (WAMs), each equipped with the 3-finger Barrett Hand. The robot's head was equipped with an Audio-Technica U853AW cardioid microphone that was used to capture auditory feedback. Proprioceptive feedback was captured by the built-in sensors in each WAM, which measure joint-torques at 500 Hz. Finally, visual feedback was detected using the robot's right eye, a 640 by 480 resolution Logitech webcam.

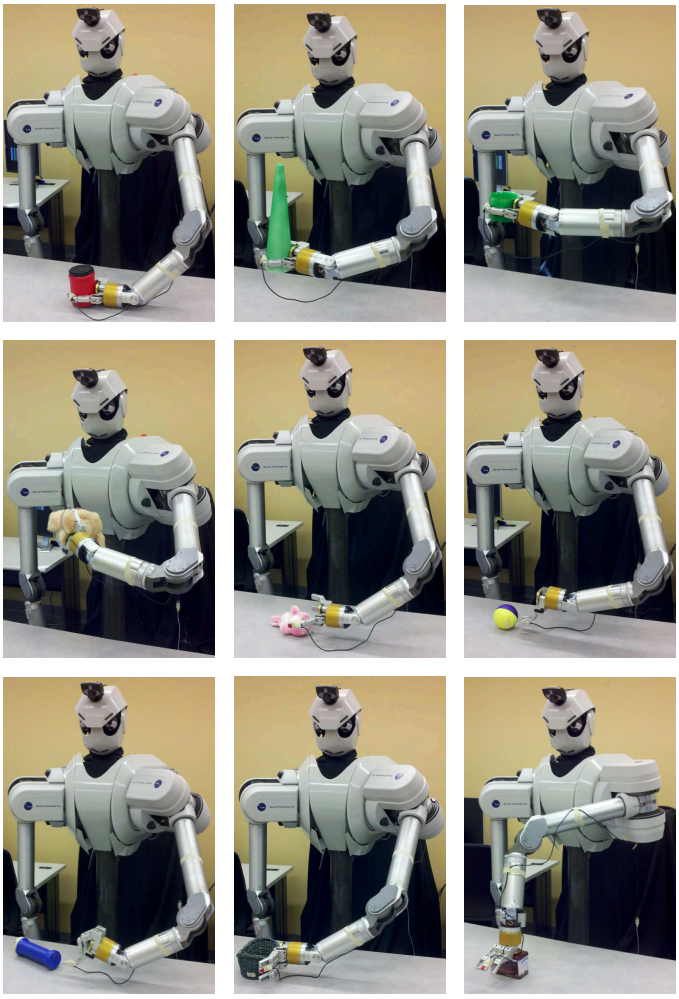


Fig. 2. The exploratory behaviors that the robot performed on all objects. From top to bottom and from left to right: *grasp, lift, hold, shake, drop, tap, poke, push,* and *press*. In addition to the 9 behaviors pictured above, the robot also performed the *look* behavior, which consisted of taking an RGB snapshot of the object on the table.

B. Objects

To test the proposed model, the robot explored 100 different household objects, which are shown in front of the robot in Fig. 1. Some of the objects are visually identical, but they differ in other properties – for example, the five red containers were filled with different contents that produced different sounds when the objects were shaken. The five blue containers, on the other hand, contained varying amounts of rice, and thus they differed only in weight. To the best of our knowledge, this dataset contains the largest number of objects ever explored by a robot over the course of a single experiment.

C. Exploratory Behaviors

The robot was equipped with 10 different behaviors that it applied on all objects: *look*, *grasp*, *lift*, *hold*, *shake*, *drop*, *tap*, *poke*, *push*, and *press*. The *look* behavior consisted of taking an RGB snapshot of the object while the other nine behaviors (see Fig. 2) were encoded as joint-space trajectories

that were executed using Barrett's default PID controller. The robot performed its set of 10 exploratory behaviors on each of the 100 objects 5 different times. This resulted in a total of 5000 behavioral interactions, which were organized into 500 exploratory trials, where each trial corresponds to the 10 different behaviors performed in a sequence on a single object. During the execution of each behavior, the robot recorded auditory, proprioceptive, and visual feedback, which were used to extract different features as described below.

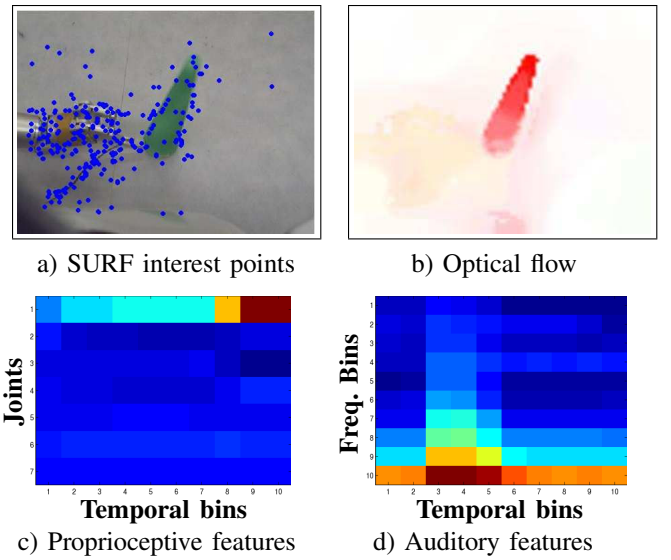
D. Sensorimotor Feature Extraction

1) Color: for each exploratory trial, the robot extracted an $8 \times 8 \times 8$ color histogram in RGB space with uniformly spaced bins from the RGB image of the object recorded during the *look* behavior. For each image, background subtraction was used to segment the object from the background.

2) SURF: the Speeded-Up Robust Features (SURF) described in [24] were computed for all images captured by the robot's camera. Fig. 3.a shows an example image captured by the robot's camera along with the detected SURF interest points. The X-means [25] algorithm was used to quantize the detected SURF feature descriptors using 0.5% of all detected feature descriptors. This resulted in a dictionary containing 200 visual "words." Using the learned quantization, for each of the 5000 behavioral interactions, a 200-dimensional feature vector was computed encoding a histogram of the SURF descriptors detected over the course of executing the behavior.

3) Optical Flow: during the execution of each behavior (except *look*), the stream of images captured by the camera was used to compute dense optical flow using the algorithm and MATLAB implementation proposed by Sun *et al.* [26]. For each pixel in a given image in the sequence, the algorithm computed a real-valued vector (u, v) encoding the direction of motion (i.e., the vector's angle) as well as the magnitude of the motion (i.e., the vector's norm). Fig. 3.b shows the detected optical flow for a single frame captured during the execution of the *poke* behavior on one of the green cones (the hue encodes the angle of the optical flow vector, while the intensity corresponds to the vector's norm). To reduce the dimensionality of the optical flow feedback, *weighted angular histogram* features were extracted from the sequence of optical flow images by binning the angles into 10 equally spaced bins. In other words, the norms of the optical flow vectors with angles ranging from 0 to $2\pi/10$ were added to bin number 1, while those in the range of $2\pi/10$ to $2 \times 2\pi/10$ were added to bin number 2, and so forth.

4) Proprioception: proprioceptive features were extracted from the recorded joint torques for all 7 joints of the robot's left arm for all behaviors except *look*. The torques were recorded at 500Hz. To reduce the dimensionality of the signal, the series of torque values for each joint were discretized into 10 temporal bins (i.e., each bin encoded the average torque that was measured over its corresponding time window). This resulted in lower-dimensional data points $\mathbf{x} \in \mathbb{R}^{10 \times 7}$, which were subsequently used to represent the robot's proprioceptive experience with the objects. Fig. 3.c shows an example 10×7 feature vector, visualized as a matrix in which the rows correspond to the 7 joints and the columns correspond to the 10 temporal bins. In addition to the joint-torque proprioceptive features, at the end of the *grasp* behavior, the final joint position for each of the three fingers was recorded and used as an additional source of proprioceptive feedback.



a) SURF interest points b) Optical flow

c) Proprioceptive features d) Auditory features

Fig. 3. Visualization of some of the sensorimotor features used by the robot. a) Sample SURF interest points computed from a single image; b) Sample dense optical flow computed while executing the *poke* behavior; c) Sample proprioceptive features detected while executing the *press* behavior; d) Sample audio features computed from the DFT for the *drop* behavior.

5) Audio: after the execution of each of the 9 interactive behaviors, the log-normalized Discrete Fourier Transform (DFT) was computed for the recorded waveform. The DFT was computed with the SPHINX4 natural language processing library package [27] using $2^7 + 1 = 129$ frequency bins. To reduce dimensionality, the DFT was further discretized using 10 temporal bins and 10 frequency bins, where the value for each bin was set to the average of the values in the DFT matrix that fell into it. Fig. 3.d shows one discretized DFT that was calculated after performing the *drop* behavior.

In summary, during each exploratory trial, the robot performed 10 exploratory behaviors on one of the 100 objects. Five of these trials were recorded for each object. During the execution of each behavior, the robot extracted features from several sensory modalities, where each viable combination of behavior and sensory modality (e.g., *drop-audio* or *look-color*) determined a unique sensorimotor context. The auditory, proprioceptive, and optical flow features were extracted while performing all 9 interactive behaviors. SURF features were extracted for all 10 behaviors. Color features were extracted from the static images captured during the *look* behavior while hand-proprioceptive features were extracted during the execution of the *grasp* behavior. Thus, the total number of sensorimotor contexts available to the robot was $9 \times 3 + 10 + 1 + 1 = 39$.

IV. Theoretical Model

A. Notation and Problem Formulation

Let $\mathcal{S}$ be the set of sensorimotor contexts available to the robot, where each context refers to a specific combination of a behavior and a sensory modality. Also, let $\mathcal{T}$ be the full set of 500 exploratory trials with all objects. During each trial, the robot applies its set of exploratory behaviors on some object $o \in \mathcal{O}$. The i^{th} exploration trial can be represented with the collection of observed sensory feedback signals, $T_i = \{x_i^s\}_{s \in \mathcal{S}}$, where each feature $x_i^s \in \mathbb{R}^{d_s}$.

The object individuation task can be formulated as follows. Let $\mathcal{T}_{test} = \{T_i\}_{i=1}^{n}$ be a test set containing n interaction trials in which the robot explored a test set of objects, $\mathcal{O}_{test} \subset \mathcal{O}$. The individuation task is to separate the set of trials $\mathcal{T}_{test}$ into groups, such that each group contains only the trials with one of the objects in $\mathcal{O}_{test}$.

In other words, the object individuation task is a special case of clustering in which each data point corresponds to a sensorimotor observation with a physical object. In contrast to fully unsupervised clustering methods, the approach described here uses prior information in the form of a set of training trials for which the object identities are known. Let $\mathcal{O}_{train} \subset \mathcal{O}$ be the objects in the robot's training set such that $\mathcal{O}_{train} \cap \mathcal{O}_{test} = \emptyset$. The set $\mathcal{T}_{train} = \{T_i, o_i\}_{i=1}^{n_{train}}$ contains the exploratory trials with the training objects, where each trial T_i is labeled with the corresponding object identity $o_i \in \mathcal{O}_{train}$.

The method for object individuation described in this paper consists of the following three stages:

1) *Distance Estimation Stage:* During this step, the robot estimates pair-wise distances for each pair of trials in $\mathcal{T}$, and for each sensorimotor context s.
2) *Learning Stage:* The data in $\mathcal{T}_{train}$ is used to learn a model that can classify a pair of trials as either "same", i.e., belonging to the same object, or "different", i.e., belonging to two different objects.
3) *Individuation Stage:* The learned model is applied on each pair of trials in the set $\mathcal{T}_{test}$, and in conjunction with a graph-based clustering algorithm is used to produce the labels of the final object individuation.

The next three subsections provide a detailed description for each of these three stages.

B. Distance Estimation Stage

In the first stage, the task is to estimate the perceptual dis-similarity for each pair of trials in the set $\mathcal{T}$. Given a sensorimotor context $s \in \mathcal{S}$, let $x_i^s \in \mathbb{R}^{d_s}$ and $x_j^s \in \mathbb{R}^{d_s}$ be the feature vectors detected in that context for trials T_i and T_j. In this work, the dis-similarity between trials T_i and T_j in context s was estimated by computing the Euclidean distance between the feature vectors x_i^s and x_j^s. Thus, for each context $s \in \mathcal{S}$, the robot estimated a pair-wise trial distance matrix, $\mathbf{W}^s \in \mathbb{R}^{|\mathcal{T}| \times |\mathcal{T}|}$, such that each entry $W_{ij}^s \in \mathbb{R}$ encoded the perceptual dis-similarity between trials T_i and T_j in that context. Finally, for each matrix, the values of all elements were linearly rescaled to lie in the range from 0.0 to 1.0.

C. Learning Stage

A fundamental pre-requisite for object individuation is the ability to detect whether two perceptual stimuli were produced by the same object or by two different objects [1]. In the method proposed here, this is accomplished by learning a model that can classify a pair of trials as either "same" or "different", where the label depends on whether the same object was present in both trials or not. To learn such a model, two types of features were extracted for each pair of trials:

- *Perceptual dis-similarity features*: given a pair of trials T_i and T_j, a feature vector $\mathbf{f}^{ij} \in \mathbb{R}^{|\mathcal{S}|}$ was computed where each element $f_s^{ij} = W_{ij}^s$ for $s = 1$ to $|\mathcal{S}|$. In other words, $\mathbf{f}^{ij}$ encodes the perceptual distances between trials T_i and T_j in all available sensorimotor contexts.
- *Dis-similarity histogram features*: given a pair of trials T_i and T_j, and the computed feature vector $\mathbf{f}^{ij}$, the values in $\mathbf{f}^{ij}$ were used to construct a histogram that encodes the distribution of dis-similarities for the two trials. The histogram was constructed using 10 equally spaced bins, resulting in a 10-dimensional feature vector $\mathbf{h}^{ij}$.

During the learning stage, the two types of features were computed for all pairs of trials T_i and T_j from the set $\mathcal{T}_{train}$. This resulted in two datasets, $\mathcal{D}_{dist} = \{\mathbf{f}^{ij}, y_{ij}\}$ and $\mathcal{D}_{hist} = \{\mathbf{h}^{ij}, y_{ij}\}$, where each $y_{ij} = +1$ if trials T_i and T_j were performed with the same object and -1 otherwise. The first dataset, $\mathcal{D}_{dist}$, contained the raw perceptual distance features for each pair of trials, while the second, $\mathcal{D}_{hist}$, was based on features that encode the distribution of the raw perceptual distances.

The datasets were subsequently used to train two machine learning classifiers, $\mathcal{M}_{dist}$ and $\mathcal{M}_{hist}$ on the task of detecting whether two trials were performed on the same object. Thus, given a trial pair (T_i, T_j), the model $\mathcal{M}_{dist}$ produced an estimate for $\text{Pr}_{dist}(\text{"same"} \,|\, \mathbf{f}^{ij})$, i.e., the probability that the two trials contained the same object. Similarly, given the same trial pair, the model $\mathcal{M}_{hist}$ produced the same estimate based on the histogram features for the trial pair, i.e., $\text{Pr}_{hist}(\text{"same"} \,|\, \mathbf{h}^{ij})$. In the experiments described in this paper, each of the two models was implemented using the WEKA [28] implementation of the AdaBoost [29] algorithm with C4.5 decision tree [30] as a base classifier.

D. Individuation Stage

Given a test set of trials $\mathcal{T}_{test}$, the outputs of the classifiers $\mathcal{M}_{dist}$ and $\mathcal{M}_{hist}$, computed for each pair of trials in $\mathcal{T}_{test}$, were used to individuate the objects as described below. Let $\mathbf{A} \in \mathbb{R}^{|\mathcal{T}_{test}| \times |\mathcal{T}_{test}|}$ be the resulting individuation matrix where each entry was computed as:

$$A_{ij} = \frac{\text{Pr}_{dist}(\text{"same"} \,|\, \mathbf{f}^{ij}) + \text{Pr}_{hist}(\text{"same"} \,|\, \mathbf{h}^{ij})}{2}$$

In other words, each entry A_{ij} corresponds to the estimated probability that trials T_i and T_j were performed with the same object. This probability was computed using a uniform combination of the outputs of the two classifiers.

To construct an object individuation using the matrix $\mathbf{A}$, the robot used the *spectral clustering* algorithm, which is one of several *graph-based* or *similarity-based* clustering algorithms [31]. Given an affinity matrix, i.e., $\mathbf{A}$, the algorithm partitions the set of trials into disjoint clusters by exploiting the eigenstructure of the matrix $\mathbf{A}$. To solve the problem efficiently, Shi and Malik [32] proposed an algorithm that optimizes the *normalized cut* objective function. Given an input individuation matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, the algorithm can be summarized with the following steps:

1) Let $\mathbf{D} \in \mathbb{R}^{n \times n}$ be the degree matrix of $\mathbf{A}$, i.e., a diagonal matrix such that $\mathbf{D}_{ii} = \sum_j A_{ij}$.
2) Solve the eigenvalue system $(\mathbf{D} - \mathbf{A})x = \lambda \mathbf{D}x$ for the eigenvector corresponding to the second smallest eigenvalue and use it to bipartition the graph.
3) If necessary, recursively bipartition each subgraph that was obtained in Step 2.

This procedure recursively bipartitions the graph induced by the matrix $\mathbf{A}$ until the spectral clustering algorithm fails to find a bipartition with a high score according to the normalized cut objective function or until it fails to find a solution to the eigenvalue system. The code for the spectral clustering algorithm (Steps 1 and 2) used in our experiments is listed on the WEKA machine learning repository website (see http: //www.cs.waikato.ac.nz/ml/weka/index_related.html).

The output of this procedure is a partitioning of the n trials into k clusters, which can be represented as a set of k sets of trials, $\mathcal{C} = \{C_\ell | \ell = 1, \ldots, k\}$ or as a label vector $\omega \in \mathbb{N}^n$ where each entry $\omega_i \in \{1, \ldots, k\}$ encodes the partition label for trial T_i. The next section describes several measures that were used to evaluate the robot's object individuation model.

V. Evaluation

A. Performance Measures

The estimated partitioning $\hat{\mathcal{C}}$ and the corresponding label vector $\hat{\omega}$ were evaluated by comparing them to the ground truth individuation, represented by the partitioning $\mathcal{C}$ and the vector ω, using several different methods.

1) Normalized Mutual Information: Normalized Mutual Information (NMI) has been proposed as a measure to capture the similarity between two different clusterings over the same dataset [33]. Given two clusterings ω^a and ω^b defined over the same set of n trials, let k^a and k^b be the number of clusters in ω^a and ω^b respectively. Let n_h^a be the number of trials in cluster C_h according to ω^a, and let n_ℓ^b the number of trials in cluster C_ℓ according to ω^b. Also, let $n_{h,\ell}$ be the number of trials that are in cluster C_h according to ω^a, as well as in cluster C_ℓ according to ω^b. Using these definitions, the NMI estimate, ϕ^{NMI}, is defined as:

$$\phi^{NMI}(\omega^a, \omega^b) = \frac{\sum_{h=1}^{k^a} \sum_{\ell=1}^{k^b} n_{h,\ell} \, log\left(\frac{n * n_{h,\ell}}{n_h^a * n_\ell^b}\right)}{\sqrt{\left(\sum_{h=1}^{k^a} n_h^a \, log(\frac{n_h^a}{n})\right)\left(\sum_{\ell=1}^{k^b} n_\ell^b \, log(\frac{n_\ell^b}{n})\right)}}.$$

This pairwise measure of mutual information is always in the range of 0.0 to 1.0, where 1.0 indicates that the two partitionings are identical while 0.0 means that the two partitionings were computed over two disjoint datasets.

2) Mean Partition Entropy: The second performance measure was chosen to evaluate the purity of each resulting cluster in the individuation with respect to object identity. Given a partition $C_\ell \in \mathcal{C}$, let $\Pr_\ell(o)$ be the probability that a randomly sampled trial from C_ℓ was performed on object $o \in \mathcal{O}$. Given the distribution over all objects for a given partition C_ℓ, Shannon's entropy [34] can be computed by:

$$H_\ell = -\sum_{o \in \mathcal{O}} \Pr_\ell(o) \, log(\Pr_\ell(o)).$$

A value of 0.0 for a cluster C_ℓ would indicate that the cluster only contains trials with one object, while large values for H_ℓ would signify that the cluster contains trials with many different objects. Thus, given the full partitioning, $\mathcal{C}$, the Mean Partition Entropy (MPE) is defined as:

$$\text{MPE}(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{C_\ell \in \mathcal{C}} H_\ell.$$

3) α-Individuation Rate: The last measure estimates the percentage of objects in the test set that were individuated correctly. An object o is considered individuated if there exists a partition C_ℓ in the set $\mathcal{C}$ that contains at least α trials with object o and no trials with any other objects. In this study, the robot performed 5 trials with each object, and therefore, the α-Individuation Rate was computed for $\alpha = 3$, 4, and 5.

B. Baseline Comparison

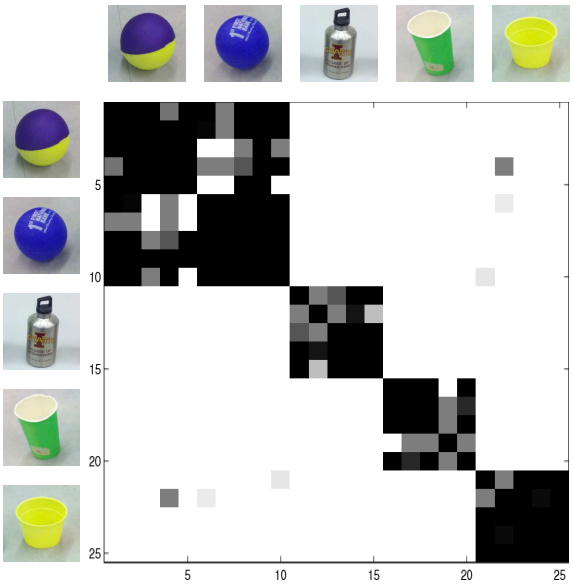
The method for object individuation was also compared against an unsupervised approach in which the test set of trials is partitioned using only the pairwise distance matrices $\mathbf{W}^s$. To do so, a trial affinity matrix $\mathbf{U}$ was constructed such that each entry $U_{ij} = (1/|\mathcal{S}|) \sum_{s \in \mathcal{S}} (1.0 - W_{ij}^s)$. In other words, each entry U_{ij} corresponds to the average perceptual similarity for the two trials computed across all sensorimotor contexts, with values close to 1.0 meaning highly similar and values close to 0.0 meaning highly dis-similar. The matrix $\mathbf{U}$ was then used as input to the partitioning algorithm described in Section IV.D to produce a final object individuation.

VI. Results

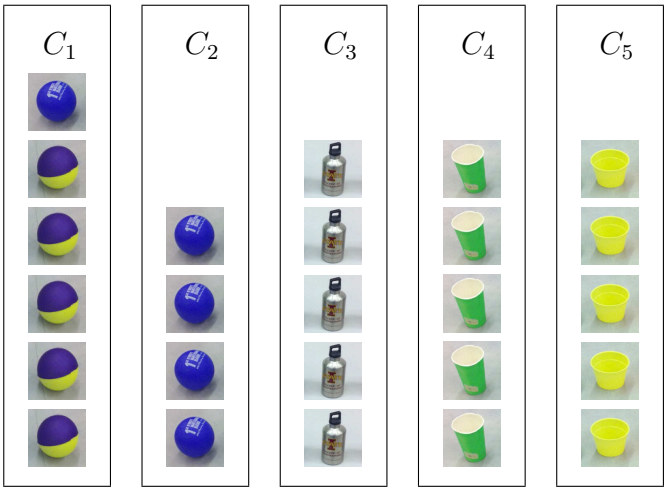
A. Example

Figure 4.a shows a sample trial individuation matrix, $\mathbf{A}$, which was computed using a test set of 25 trials with 5 different objects (5 trials per object). Each entry in the matrix encodes the estimated probability that a pair of trials was performed with the same object, where dark indicates high likelihood and white indicates low likelihood. The individuation model used to fill in the entries of the matrix was trained on a separate set of 25 trials with another set of 5 objects.

For visualization purposes, the entries of the matrix are sorted by object identity. Because the matrix is sorted, the block pattern along the diagonal clearly shows that the learned



a) Estimated Object Individuation Matrix



b) Resulting Individuation

Fig. 4. a) An example object individuation matrix $\mathbf{A}$. The matrix encodes the estimated likelihood that a pair of trials in the test set were performed on the same object, where dark indicates high likelihood and white indicates low likelihood. In this example, the test set contained 25 trials with 5 different objects (5 trials per object). For better visualization, the entries of the matrix are sorted by object identity. b) The resulting object individuation. Each partition corresponds to a set of trials that, according to the trained model, were performed with the same object.

model was able to detect which pairs of trials were performed with the same object far better than chance. For comparison, Fig. 5 shows the perceptual similarity matrix, $\mathbf{U}$, for the same 25 exploratory trials, computed from the 39 raw context-specific distance matrices $\mathbf{W}^s$ using the unsupervised baseline approach. It is easy to see that the matrix $\mathbf{U}$ has more non-zero entries than the matrix $\mathbf{A}$ for pairs of trials that do not belong to the same object.

The estimated object individuation matrix $\mathbf{A}$ was used as an input to the partitioning algorithm to produce the final individuation shown in Fig. 4.b. Each of the 5 partitions in

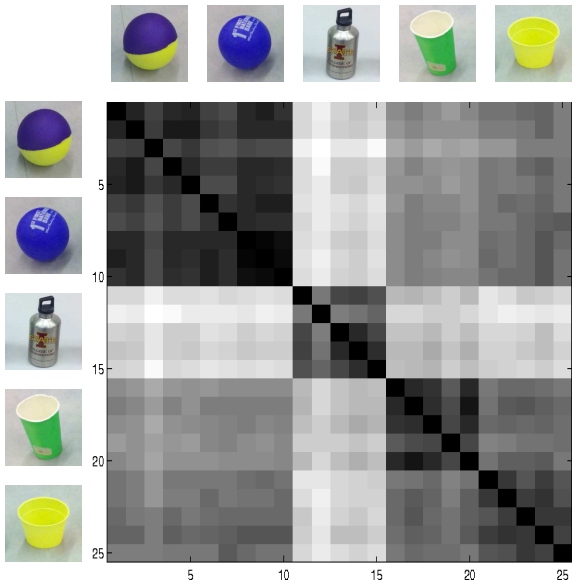


Fig. 5. An example perceptual similarity matrix, $\mathbf{U}$, for 25 exploratory trials computed using the 39 raw context-specific distance matrices $\mathbf{W}^s$.

the individuation corresponds to a set of trials that, according to the model, were performed with the same object. In this example, the model made one mistake as it incorrectly grouped one of the trials performed with the blue ball with the set of trials performed with the purple-yellow ball. The Normalized Mutual Information (NMI) between the output individuation and the ground truth individuation was 0.935. The α-Individuation Rate for $\alpha = 3$ and $\alpha = 4$ was 80.0% since in both cases there was one object (the first ball) that could not be individuated on its own. For $\alpha = 5$, the rate was 60.0% since only 3 of the objects were perfectly individuated (i.e., with all 5 trials in the same partition). To compare, when the perceptual similarity matrix $\mathbf{U}$ (see Fig. 5) was used to partition the test trials the results were noticeably worse. The individuation had a substantially lower NMI of 0.809 and the α-individuation rate was only 20.0% for $\alpha = 3, 4$ and 5 (i.e., one partition contained 5 trials with a single object, while all others were mixed).

B. Baseline Comparison

The proposed individuation model was compared against the baseline unsupervised approach for partitioning the trials in the test set. During each test, the two approaches were evaluated using a randomly sampled set of 20 training objects and another randomly sampled set of 20 test objects, such that the two sets were disjoint. To compare against a chance model, the same experiment was performed with the added step of randomly shuffling the entries in the individuation matrix $\mathbf{A}$ before clustering it (i.e., multiple randomly chosen pairs of values in the matrix were swapped before using the matrix to compute the partitioning). Table I shows the results of these evaluations, averaged over 100 tests. Both the learned and the unsupervised models performed much better than chance. Furthermore, the superior performance of the learned model

TABLE I
COMPARISON BETWEEN THE LEARNED INDIVIDUATION MODEL, THE BASELINE UNSUPERVISED MODEL, AND THE CHANCE MODEL

	Normalized Mutual Information	Mean Partition Entropy	α-Individuation Rate (%)		
			$\alpha = 3$	$\alpha = 4$	$\alpha = 5$
Learned	0.964	0.056	87.1	74.5	71.5
Unsupervised	0.878	0.416	32.2	32.2	31.9
Random	0.506	1.373	0.0	0.0	0.0

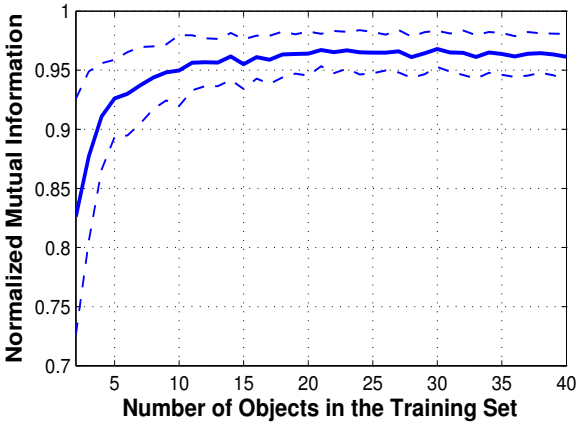


Fig. 6. Performance of the robot's object individuation model, measured by the Normalized Mutual Information criterion, as a function of the number of objects used to train it. The dashed lines show the standard deviation, which was computed over 100 tests.

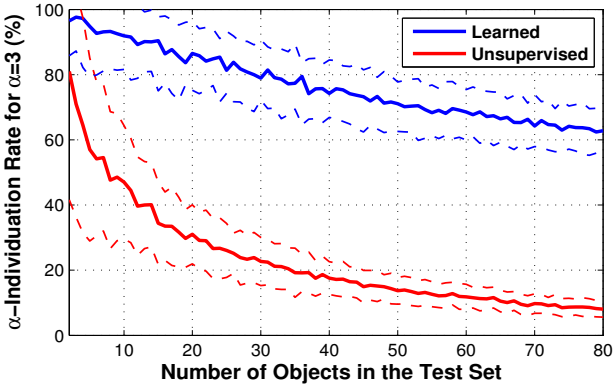


Fig. 7. Performance of the learned individuation model and the baseline unsupervised model as a function of the number of objects in the test set.

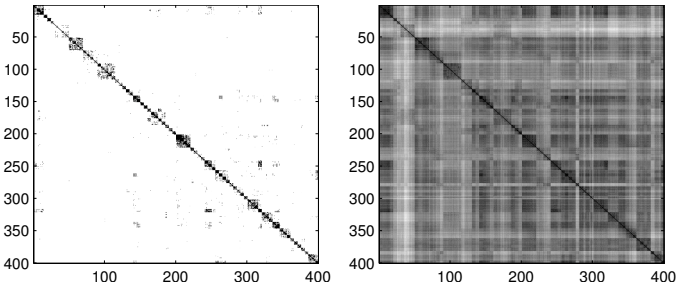


Fig. 8. Example object individuation matrix, $\mathbf{A}$ (left), and perceptual similarity matrix, $\mathbf{U}$ (right), for a set 400 exploratory trials with 80 different objects (5 trials per object).

clearly shows that prior information, in the form of exploratory trials with known object identities, can substantially improve the robot's performance when individuating novel objects.

C. Performance vs. Number of Training Objects

The performance of the object individuation model was also evaluated as a function of the number of training objects, m, which was varied from 2 to 40. For each value, 100 tests were performed, such that during each test the model was evaluated using a randomly sampled set of m training objects and another randomly sampled set of 20 test objects.

The results of these tests, shown in Fig. 6, indicate that the model's performance converges once there are at least 20 objects in the training set. Overall, even with a small number of training objects, the robot's model is able to successfully individuate novel objects substantially better than chance.

D. Performance vs. Number of Test Objects

The last experiment explored the relationship between the number of objects in the test set and the performance of the robot's object individuation model. Studies in psychology have shown that there are inherent limits on the number of objects that humans can individuate at a time [12], [35]. To find out if the same is true for our robot, the number of objects in the test set was varied from 2 to 80, while the number of training objects was kept constant at 20.

Figure 7 shows the results of this experiment, where performance was measured using the Normalized Mutual Information measure. The results show that, just as with humans, the individuation task becomes more difficult as the number of test objects is increased. A possible explanation for this is that as the test set becomes larger, there are more pairs of perceptually similar objects that complicate the task. Nevertheless, even with a test set of 80 objects, the learned model was still able to successfully individuate over 60.0% of the objects. The unsupervised model, on the other hand, was able to individuate only 10% of the novel objects.

Fig. 8 shows example object individuation and perceptual similarity matrices ($\mathbf{A}$ and $\mathbf{U}$) for a test set of 400 exploratory trials with 80 objects (5 trials per object). As before, the entries in the matrices are sorted by object identity. Unlike the perceptual similarity matrix, the individuation matrix is sparse and has very few large values for pairs of trials that were performed with two different objects. Furthermore, as shown in Fig. 7, the performance of the unsupervised model drops at a much faster rate as the number of objects is increased, which showcases the need for prior training before attempting to individuate novel objects.

VII. CONCLUSION AND FUTURE WORK

While the problem of object recognition is well studied in robotics, the task of individuating novel objects that were not part of the robot's training set has received very little attention. To address this gap, this paper proposed a method that allows a robot to successfully partition its sensorimotor experience with novel objects into clusters that correspond to the identities of the objects. The proposed method was tested with a large-scale

dataset in which the robot explored 100 objects using a variety of exploratory behaviors and sensory modalities. Using prior information from exploratory trials for which the identities of the objects are known, the robot was able to achieve high performance on the task of object individuation as measured by several different performance measures.

A key result from our experiments is that unsupervised methods for partitioning of the robot's sensorimotor experience may not be sufficient for solving the object individuation problem. Instead, prior information, in the form of exploratory trials with known object identities, is needed in order to learn whether the observed perceptual differences between two sensorimotor interactions are due to noise or due to the fact that the interactions were performed with two different objects. On average, the use of training data allowed the model to successfully individuate 87.1% of the objects in a test set of size 20, while, without it, the unsupervised model individuated only 32.2% of the 20 objects. Even with a larger test set of 80 objects, the learned model was able to individuate over 60% of the objects, while the model without prior training was able to individuate only 10% of the objects.

Another important result of this paper is that, similar to studies with humans, performance was sensitive to the number of objects to be individuated. Therefore, one viable direction for future work is to explore ways of individuating a large number of objects by incrementally individuating smaller object subsets. Another direction for future work is to consider the effect of category and object labels on the individuation, since it has been shown that the presence of labels (i.e., words that describe the object) can improve the object individuation performance of human infants [36].

REFERENCES

[1] P. Krøjgaard, "A review of object individuation in infancy," *British journal of developmental psychology*, vol. 22, no. 2, pp. 159–183, 2004.

[2] C. Kemp, A. Jern, and F. Xu, "Object discovery and identification," *Advances in Neural Information Processing Systems*, vol. 22, 2009.

[3] P. Tremoulet, A. Leslie, and D. Hall, "Infant individuation and identification of objects," *Cognitive Development*, vol. 15, no. 4, pp. 499–522, 2000.

[4] E. Torres-Jara, L. Natale, and P. Fitzpatrick, "Tapping into touch," in *Proc. 5-th Intl. Workshop on Epigenetic Robotics*, 2005, pp. 79–86.

[5] J. Sinapov, M. Weimer, and A. Stoytchev, "Interactive learning of the acoustic properties of household objects," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 2518–2524.

[6] L. Natale, G. Metta, and G. Sandini, "Learning haptic representation of objects," in *International Conference on Intelligent Manipulation and Grasping*, 2004.

[7] B. Rasolzadeh, M. Björkman, K. Huebner, and D. Kragic, "An active vision system for detecting, fixating and manipulating objects in the real world," *International Journal of Robotics Research*, vol. 29, no. 2-3, pp. 133–154, 2010.

[8] T. Bergquist, C. Schenck, U. Ohiri, J. Sinapov, S. Griffith, and A. Stoytchev, "Interactive object recognition using proprioceptive feedback," in *Proceedings of the 2009 IROS Workshop: Semantic Perception for Robot Manipulation, St. Louis, MO*, 2009.

[9] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D point cloud based object maps for household environments," *Robotics and Autonomous Systems*, vol. 56, no. 11, pp. 927 – 941, 2008.

[10] J. Sinapov, T. Bergquist, C. Schenck, U. Ohiri, S. Griffith, and A. Stoytchev, "Interactive object recognition using proprioceptive and auditory feedback," *The International Journal of Robotics Research*, vol. 30, no. 10, pp. 1250–1262, 2011.

[11] Z.-C. Marton, F. Seidel, F. Balint-Benczedi, and M. Beetz, "Ensembles of Strong Learners for Multi-cue Classification," *Pattern Recognition Letters (PRL), Special Issue on Scene Understandings and Behaviours Analysis*, 2012, in press.

[12] Y. Xu and M. Chun, "Selecting and perceiving multiple visual objects," *Trends in cognitive sciences*, vol. 13, no. 4, pp. 167–174, 2009.

[13] C. Becchio and C. Bertone, "Object temporal connotation," *Brain and cognition*, vol. 52, no. 2, pp. 192–196, 2003.

[14] M. Quigley, E. Berger, and A. Ng, "STAIR: Hardware and software architecture," *Presented at AAAI 2007 Robotics Workshop*, 2007.

[15] S. Srinivasa, C. Ferguson, D. Helfrich, D. Berenson, A. Collet, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and M. VandeWeghe, "HERB: A Home Exploring Robotic Butler," *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, 2009.

[16] H. Kang, M. Hebert, A. Efros, and T. Kanade, "Connecting missing links: Object discovery from sparse observations using 5 million product images," in *Proc. of the European Conference on Computer Vision*, 2012.

[17] T. Bhattacharjee, J. Rehg, and C. Kemp, "Haptic classification and recognition of objects using a tactile sensing forearm," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[18] J. Fishel and G. Loeb, "Bayesian exploration for intelligent identification of textures," *Frontiers in Neurorobotics*, vol. 6, 2012.

[19] J. Sinapov and A. Stoytchev, "Object category recognition by a humanoid robot using behavior-grounded relational learning," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 184–190.

[20] J. Sinapov, C. Schenck, K. Staley, V. Sukhoy, and A. Stoytchev, "Grounding semantic categories in behavioral interactions: Experiments with 100 objects," *Robotics and Autonomous Systems (in press)*, 2012.

[21] A. Stoytchev, "Behavior-grounded representation of tool affordances," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA)*, 2005, pp. 3060–3065.

[22] J. Sinapov and A. Stoytchev, "Detecting the functional similarities between tools using a hierarchical representation of outcomes," in *Proc. of the 7th IEEE International Conference on Development and Learning (ICDL)*, 2008, pp. 91–96.

[23] S. Griffith, J. Sinapov, V. Sukhoy, and A. Stoytchev, "A behavior–grounded approach to forming object categories: Separating containers from non-containers," *IEEE Transactions on Autonomous Mental Development*, vol. 4, no. 1, pp. 54–69, 2012.

[24] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[25] D. Pelleg and A. W. Moore, "X-means: Extending k-means with efficient estimation of the number of clusters," in *17th Int. Conf. on Machine Learning*, 2000, pp. 727–734.

[26] D. Sun, S. Roth, and M. Black, "Secrets of optical flow estimation and their principles," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 2432 –2439.

[27] K. Lee, H. Hon, and R. Reddy, "An overview of the SPHINX speech recognition system," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 1, pp. 35–45, 1990.

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

[29] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Thirteenth International Conference on Machine Learning*. San Francisco: Morgan Kaufmann, 1996, pp. 148–156.

[30] R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

[31] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[32] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[33] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *The Journal of Machine Learning Research*, vol. 3, pp. 583–617, 2003.

[34] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[35] L. Feigenson and S. Carey, "On the limits of infants' quantification of small object arrays," *Cognition*, vol. 97, no. 3, pp. 295–313, 2005.

[36] F. Xu, M. Cote, and A. Baker, "Labeling guides object individuation in 12-month-old infants," *Psychological Science*, vol. 16, no. 5, pp. 372–377, 2005.