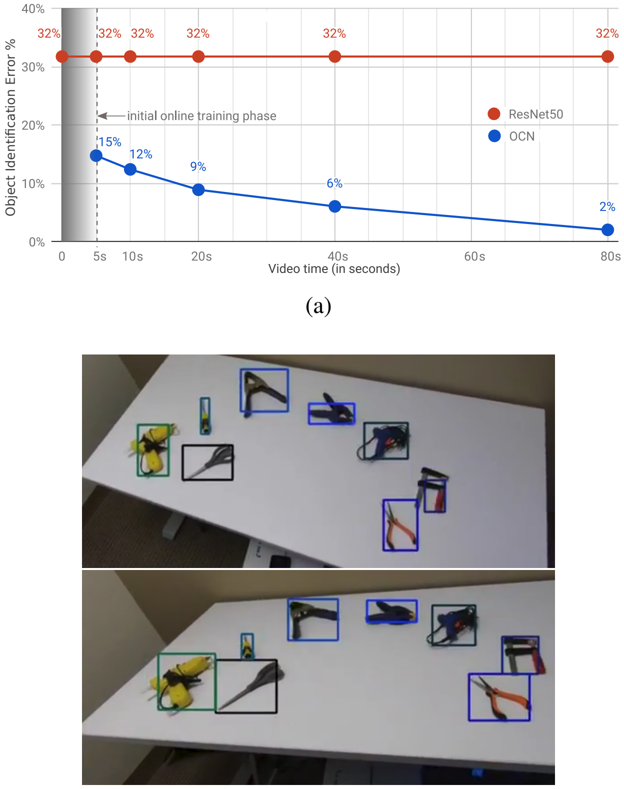
Online Object Representations with Contrastive Learning in Videos

Sören Pirk[1], Mohi Khansari[2], Yunfei Bai[2], Corey Lynch[1], Pierre Sermanet[1]
[1]Google Brain, [2]X

We propose a self-supervised approach for learning representations of objects from monocular videos and demonstrate it is particularly useful in situated settings such as robotics. The main contributions of this paper are: 1) a self-supervising objective that can discover and disentangle object attributes from video without using any labels; 2) leveraging object self-supervision for online adaptation: the longer our online model looks at objects in a video, the lower the object identification error, while the offline baseline remains with a large fixed error; 3) to explore the possibilities of a system entirely free of human supervision, we let a robot collect its own data, then self-supervise on it from scratch, and then show the robot can point to objects similar to the one presented in front of it. We encourage the reader to watch the video demonstrating online object adaptation as well as robotic pointing at this address: https://online-objects.github.io/

Motivation: One of the biggest challenge in real world robotics is robustness and adaptability to new situations. Indeed a robot deployed in the real world is likely to encounter a number of objects it has never seen before. Even if it can recognize the class of an object, it may be useful to recognize a particular instance of it. Relying on human supervision in this context is unrealistic. Instead if a robot can self-supervise its understanding of objects, it can adapt to new situations with online learning. Online self-supervision is key to robustness and adaptability and prerequisite to real-world deployment. Because we focus on situated settings (i.e. an agent is embedded in an environment), we can leverage signals such as temporal continuity in videos, as opposed to non-contiguous frames. We use temporal continuity as the basis for self-supervising correspondence between different views of objects.

Approach: We propose a model called Object-Contrastive Networks (OCN) trained with a metric learning loss (see Fig. 2). The approach is very simple: 1) extract object bounding boxes using a general off-the-shelf objectness detector ([1]), 2) train a deep object model on each bounding box extract from any random pair of frames from the video, using the following training objective: nearest neighbors in the embedding space are pulled together from different frames while being pushed away from the other objects from any frame (using n-pairs loss [2]). This does not rely on knowing the true correspondence between objects. The fact that this works at all despite not using any labels might be surprising. One of the main findings of this paper is that given a limited set of objects, object correspondences will naturally emerge when using metric learning. One advantage of self-supervising object representation is that these continuous representations are not biased by or limited to a discrete set of labels determined by human annotators. We



(a)

(b)

Figure 1: **The longer our model looks at objects in a video, the lower the object identification error**. Our model self-supervises object representations as the video progresses and converges to 2% error while the offline baseline remains at 32% error as shown in a. Two example frames of the video in b.

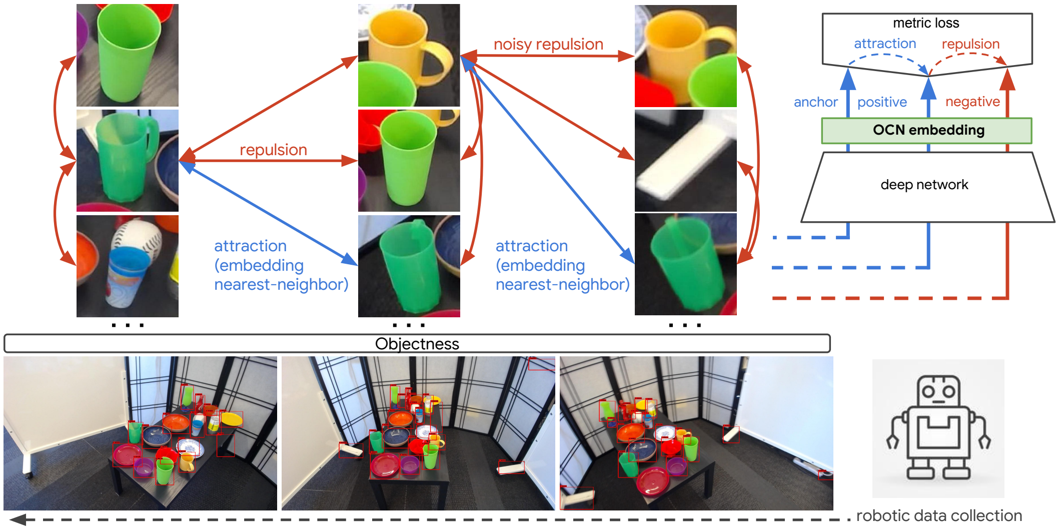


Figure 2: **Object-Contrastive Networks (OCN)**: by attracting embedding nearest neighbors and repulsing others using metric learning, continuous object representations naturally emerge. In a video collected by a robot looking at a table from different viewpoints, objects are extracted from random pairs of frames. Given two lists of objects, each object is attracted to its closest neighbor while being pushed against all other objects. Noisy repulsion may occur when the same object across viewpoint is not matched against itself. However the learning still converges towards disentangled and semantically meaningful object representations which can be useful in autonomous robotics applications.

show these embeddings discover and disentangle object attributes and generalize to previously unseen environments.

Online Results: We quantitatively evaluate the online adaptation capabilities of our model through the object identification error of entirely novel objects. In Fig. 1 we show that a model observing objects for a few minutes from different angles can self-teach to identify them almost perfectly while the offline supervised approach cannot. OCN is trained on the first 5s, 10s, 20s, 40s and 80s of the 200s video, then evaluated on the identification error in the last 120s of the video for each phase. The supervised offline baseline stays at 32% error rate, while the OCN improves down to 2% error after 80s, a 15x error reduction.

Offline Analysis: To analyze what our model is able to disentangle, we quantitatively evaluate performance on a large-scale synthetic dataset with 12k object models (e.g. Fig. 3a), as well as on a real dataset collected by a robot and show that our unsupervised object understanding generalizes to previously unseen objects. In Tab. 1 we find that our self-supervised model closely follows its supervised equivalent baseline when trained with metric learning. As expected the cross-entropy/softmax supervised baseline approach performs best and establishes the error lower bound while the ResNet50 baseline are upper-bound results.

Robotic Experiments: Here we let a robot collect its own data by looking at a table from multiple angles (Fig. 2 and Fig. 3b). It then trains itself with OCN on that data, and asked to point to objects similar to the one presented in front of it. Objects can be similar in terms of shape, color or class. If able to perform that task, the robot has learned to distinguish and recognize these attributes entirely on its own, from scratch and by collecting its own data. We find in Tab. 2 that the robot is able to perform the pointing task with 72% recognition accuracy of 5 classes, and 89% recognition accuracy of the binary is-container attribute.

Future Work: Current limitations include relying on all objects to be present in all frames of a video. Relaxing this constraint will allow for a more realistic use of the model. Additionally, the online training is currently not real-time as we first set out to demonstrate the usefulness of online-learning in non-real-time. Real-time training requires additional engineering that is beyond the scope of this research. Finally, the model currently relies on an off-the-self object detector which might be noisy, an avenue for future research is to backpropagate gradients through the objectness model as well to improve detection and reduce noise.

References

[1] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99. 2015. 1

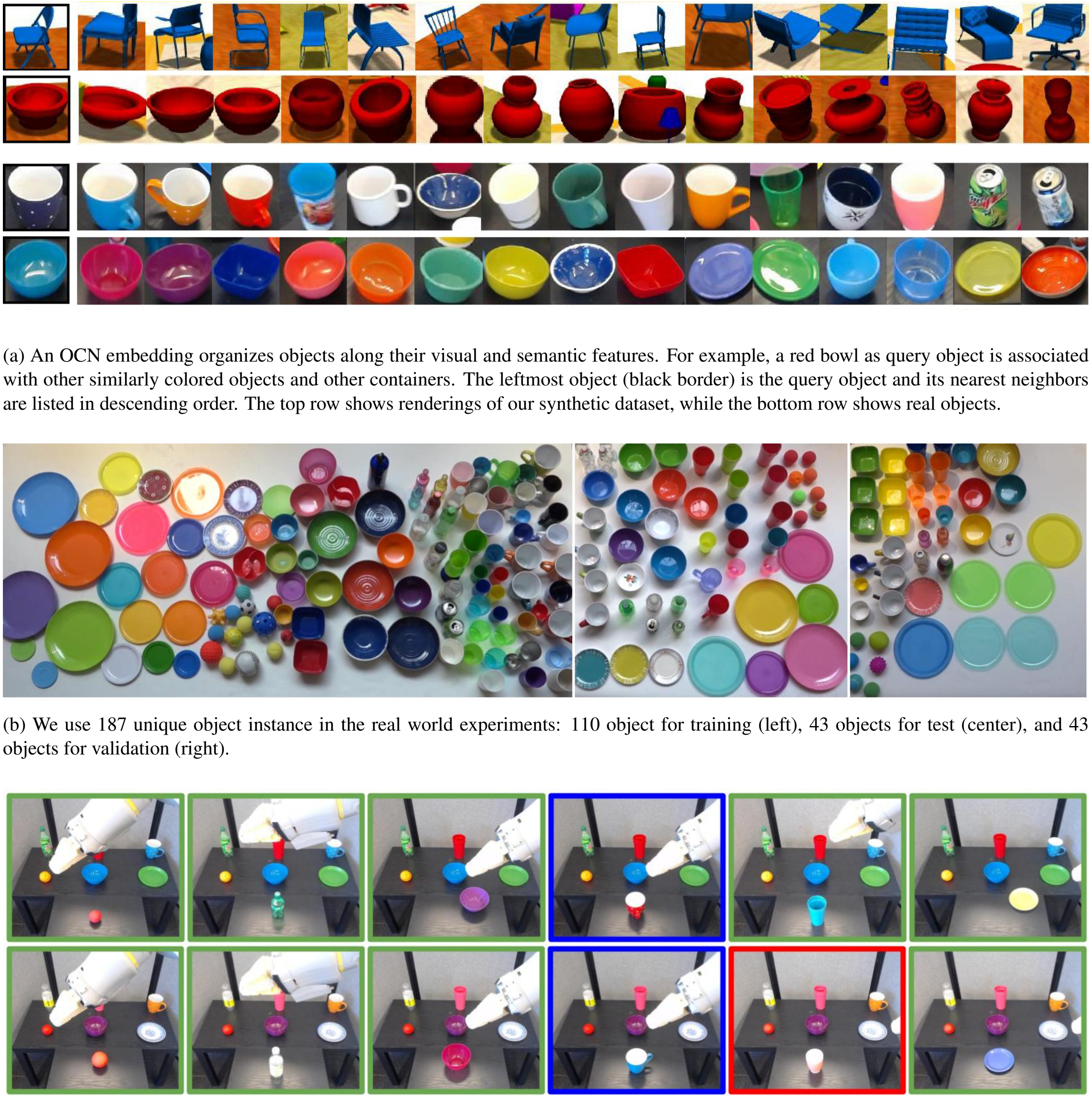
[2] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *NIPS*, pages 1857–1865. 2016. 1

Table 1: **Attributes classification errors:** using attribute labels, we train either a linear or nearest-neighbor classifier on top of existing fixed embeddings. The supervised OCN is trained using labeled positive matches, while the unsupervised one decides on positive matches on its own. All models here are initialized and frozen with ImageNet-pretrained weights for the ResNet50 part of the architecture, while the additional layers above are random and trainable.

	Class (12)	Color (8)	Binary	
	Attribute	Attribute	Attributes	Embedding
Method	Error	Error	Error	Size
[baseline] Softmax	2.98%	0.80%	7.18%	-
[baseline] OCN supervised (linear)	7.49%	3.01%	12.77%	32
[baseline] OCN supervised (NN)	9.59%	3.66%	12.75%	32
[ours] OCN unsupervised (linear)	10.70%	5.84%	13.76%	24
[ours] OCN unsupervised (NN)	12.35%	8.21%	13.75%	24
[baseline] ResNet50 embeddings (NN)	14.82%	64.01%	13.33%	2048
[baseline] Random Chance	91.68%	87.50%	50.00%	-

Table 2: Quantitative evaluation on the robot pointing experiment. We report on two attribute errors: ‘class’ and ‘container’. An error for ’class’ is reported when the robot points to an object of a different class among these 5 categories: Balls, plates, bottles, cups, bowls. An error for ’container’ is reported when the robot points to a non-container object when presented with a container object, and vice-versa.

Attributes	Balls	Bottles & Cans	Bowls	Cups & Mugs	Glasses	Plates	Total
Class error	11.1 ±7.9%	0.0 ±0.0%	22.2 ±15.7%	88.9 ±7.9%	38.9 ±7.9%	5.6 ±7.9%	27.8 ±3.9%
Container error	11.1 ±7.9%	0 ±0.0%	16.7 ±13.6%	16.7 ±0.0%	16.7 ±13.6%	5.6 ±7.9%	11.1 ±2.3%



(a) An OCN embedding organizes objects along their visual and semantic features. For example, a red bowl as query object is associated with other similarly colored objects and other containers. The leftmost object (black border) is the query object and its nearest neighbors are listed in descending order. The top row shows renderings of our synthetic dataset, while the bottom row shows real objects.

(b) We use 187 unique object instance in the real world experiments: 110 object for training (left), 43 objects for test (center), and 43 objects for validation (right).

(c) The robot experiment of pointing to the best match to a query object (placed in front of the robot on the small table). The closest match is selected from two sets of target object list, which are placed on the long table behind the query object. The first and the second row respectively correspond to the experiment for the first and second target object lists. Each column also illustrates the query objects for each object category. Image snapshots with green frame correspond to cases where both the ‘class’ and ‘container’ attributes are matched correctly. Image snapshots with blue frame refer to the cases where only ‘container’ attribute is matched correctly. Images with red frames indicates neither of attributes are matched.

Figure 3: Simulated and real robotics experiments.