Affordance Discovery using Simulated Exploration

Robotics Track

Adam Allevato
The University of Texas at Austin
Austin, TX
allevato@utexas.edu

Andrea Thomaz
The University of Texas at Austin
Austin, TX
athomaz@ece.utexas.edu

Mitch Pryor
The University of Texas at Austin
Austin, TX
mpryor@utexas.edu

ACM Reference Format:
Adam Allevato, Andrea Thomaz, and Mitch Pryor. 2018. Affordance Discovery using Simulated Exploration. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 3 pages.

1 INTRODUCTION

Allowing robots to understand their world in terms of *affordances* [7] allows for generalization, learning, and complex planning, while also being intuitive for humans to understand. In recent work, affordances are often learned with hand-coded robot actions, which can limit or bias the model. Real-world training has also been used to learn affordances [5] and manipulation models [1], but is time-consuming and unsafe for the robot and its environment.

In this work, we present a method for learning affordance models by leveraging randomized self-exploration entirely in simulation. Our approach learns dynamic behavior via simulated manipulation actions sampled from a continuous feature space, building a model of action-effect correspondences. Clustering provides a subset of actions which is labeled by humans to provide context to the model. These labels, actions, and effects together make up an affordance model that can plan actions to invoke a desired label. We show that our method results in robust controllers, even for subtle affordances such as *leaning* and *touching*. We evaluate our results by performing experiments both in simulation and on a real robot with a set of 6 reference objects.

2 RELATED WORK

Robotic agents can learn how to interact with their environment through self-exploration [1, 13], articulated motion models [9], and grasping [10]. Exploration and model estimation can also be done in simulation [2, 3, 12], with neural networks recently becoming the norm for both real and simulated efforts. By using an object-grounded affordance representation and applying human-readable labels, we learn a model for both robots and humans, rather than an opaque end-to-end network.

Chu et al. coupled human-guided exploration and self-exploration to learn a fixed set of affordances [5]. On the other hand, simulations and crowdsourcing have been used for large-scale affordance learning [6, 8, 16, 17, 19], but only for a fixed set of manipulation actions. Our work combines continuously-sampled actions and open-ended effects, as well as labels, into a single framework.

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

3 AFFORDANCE LEARNING

We formally define affordances as relations between the spaces of objects ($\mathcal{O}$), actions ($\mathcal{A}$), and effects ($\mathcal{E}$) [11]. Our aim in particular is to find an action $a \in \mathcal{A}$ to induce a particular effect $e \in \mathcal{E}$ on an object $o \in \mathcal{O}$, the so called "action planning" step.

We define an *affordance label*, $l \in \mathcal{L}$, as a short, descriptive natural-language phrase that maps to one or more effects. A *labeling function* f maps every possible effect to a label, $f : \mathcal{E} \to \mathcal{L}$. Affordance labels augment the affordance model and achieve an appropriate shared representation between human and robot. We use a single label to describe a cluster of related effects to account for the ambiguity inherent in natural language labels.

Our goal is to use simulated data to learn models for an affordance-based action selection system that can be deployed on a real robot. The procedure has four main steps: data collection, data labeling, model construction, and action selection/playback.

To collect data, we simulate actions in the MuJoCo simulator [18]. The 6-element input includes the starting and ending (x, y, z) position of the end effector in a reference frame centered on the target object. This is a richer representation than prior work, which generally constrains actions to a fixed-length motion parameterized by an (x, y) point and an angle. We search the action space using random sampling, rather than the predefined actions or grid search used in prior work [5, 19]. Each manipulation action includes one object, one gripper, and sampled action. After each 8-second episode, we store the 6D action and the effect: a normalized $\mathbf{SE}(3)$ transformation, $(\Delta x/s_x, \Delta y/s_y, \Delta z/s_z, \Delta rx, \Delta ry, \Delta rz)$, where $s_{\{x,y,z\}}$ are the object's bounding box dimensions.

After collecting action-effect data, we cluster in the normalized $\mathbf{SE}(3)$ effect space of each individual object by fitting a Gaussian Mixture Model (GMM) using the Expectation-Maximization (EM) algorithm and minimizing the Bayesian Information Criteria (BIC) [15], producing a generative model of object behavior. To account for possible over-segmentation, we replay simulations and collect one- and two-word labels from humans for each of the GMM clusters. (After labeling the exemplars for all clusters, some clusters map to the same label.)

The affordance model associates $\mathbf{SE}(3)$ object transformations with labels independent of class and size. Therefore, we expect labels to transfer from one object to another with no transformations. To transfer the model, we predict the likelihood of new objects' effects using the labeled GMM developed for the first object. Low-likelihood effects and unused labels are discarded, and the remainder makes up the transferred model.

At execution time, the robot needs to select an appropriate action to achieve a desired affordance label on a particular object. If the provided label maps to multiple effect clusters, we must choose a

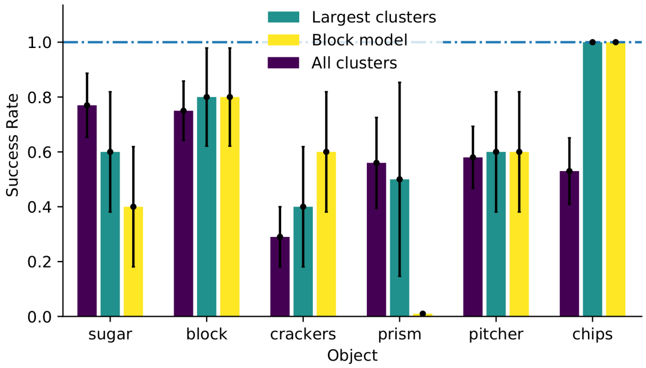

Figure 1: Various action selection methods, scored by how often they resulted in the correct effect in *real-world* execution. Bars denote standard error.

specific action to perform. First, for each cluster, we find the effect closest to the cluster center, and define it as the *exemplar*. We then evaluate exemplars for all clusters that match the label ("all clusters"), as well as the exemplar of only the highest-population cluster for each label ("largest clusters"). The exemplar action(s) can be replayed directly on the robot using a Cartesian-space motion planner, converting the 6D action representation into a joint trajectory[1].

4 EXPERIMENTAL VALIDATION

We developed affordance models for a set of 6 objects (see Figure 2. The objects were modeled with low-fidelity mesh models and primitives. Action samples were taken from $\alpha x \alpha x z_o$ regions, with the start region centered on $(\frac{\alpha}{2}\text{m}, 0, z_o/2)$ and the end region centered on $(-\frac{\alpha}{2}\text{m}, 0, z_o/2)$, where z_o is the height of the manipulated object. We also let $\alpha = 0.2$m—dictated by the overall working space of our robot arm. 2000 actions were sampled for each object.

For each component of the GMM, we found the exemplar action as described in Section 3, and collected labels from a member of the research team. During this ground-truth labeling, we used 6 affordance labels and select between them by applying simple qualitative rules.

To evaluate how well a labeling function generalizes, we want to measure its accuracy when its applied to objects other than the one used to build the model. Therefore, we perform a cross-validation experiment, applying models from each object to all the others. The labeling function $f_a()$ learned for one object is applied to all other objects (see Section 3). The prediction accuracy is then given by the equation $\textbf{accuracy}(a, b) = \sum_{i=1}^{n} \mathbf{1}(f_a(e_i) = l_i)/n$, where $\mathbf{1}(\cdot)$ is the indicator function. We expect the cross-validation accuracy, $accuracy(a, a)$, to equal 1 when self-applied, since in that case, the labeling function f_a is performing prediction on the data used to originally generate it.

5 RESULTS

In terms of action planning performance, Figure 1 shows that the "largest clusters" approach outperformed "all clusters" in a majority of cases. This suggests that the largest cluster for any given label does in fact represent the primary example of the affordance effect.

The labeling model learned for the block had the highest average prediction score of all the cross-applied models. We attribute to the fact that the block's action-effect mapping contains several different behaviors, allowing it to capture behaviors of other objects as well as its own.

The models learned in simulation are influenced heavily by the object dimensions and (in the case of the mesh-modeled objects) unstable simulation behavior. Despite this, the affordances discovered via our method, shown in Figure 2, are subtle enough to include a *leaning* behavior, where the object rotates about its x- or y-axis, but does not tip. A modified exploration strategy and more varied actions could potentially uncover even more behaviors on each object, resulting in better cross-object model transfer.

6 DISCUSSION AND CONCLUSIONS

We have developed a method for generating object affordance models purely using open-ended simulated exploration, and shown results that point to the ability to develop affordance models that are object-invariant. A key area for future study is the label collection process. Ideally, labels are supplied by non-expert users, rather than supplying labels from a single expert user. Collecting labels for a large number of users and different object types would provide insight into how well labels generalize across effects, actions, and human users.

ACKNOWLEDGMENTS

This material is based upon work supported by a Department of Energy Nuclear Energy University Program Graduate Fellowship and ONR Grant N00014-16-1-2785.

[1]We deployed our model on a robot with a Kinova Jaco arm and a Weiss WSG50 parallel gripper. The platform includes two onboard computers and runs ROS [14] Kinetic.

Figure 2: Affordances exhibited on our robot. The object/label pairs shown are: (a) sugar/tip (b) block/no-effect (c) crackers/push (d) prism/touch (e) pitcher/turn (f) chips/lean. The chips, crackers, and sugar objects are from the YCB [4] object set.

REFERENCES

[1] Pulkit Agrawal, Ashvin V Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. 2016. Learning to poke by poking: Experiential learning of intuitive physics. In *Advances in Neural Information Processing Systems*. 5074–5082.

[2] Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. 2017. Hindsight Experience Replay. *arXiv* (jul 2017). arXiv:1707.01495 http://arxiv.org/abs/1707.01495

[3] Arunkumar Byravan and Dieter Fox. 2017. SE3-nets: Learning rigid body motion using deep neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 173–180. https://doi.org/10.1109/ICRA.2017.7989023

[4] Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. 2015. The YCB object and model set: Towards common benchmarks for manipulation research. In *2015 International Conference on Advanced Robotics (ICAR)*. IEEE, 510–517.

[5] Vivian Chu and Andrea L Thomaz. 2017. Analyzing differences between teachers when learning object affordances via guided exploration. *The International Journal of Robotics Research* (Mar 2017), 0278364917693691. https://doi.org/10.1177/0278364917693691

[6] Paulo Abelha Ferreira and Frank Guerin. 2017. Learning How a Tool Affords by Simulating 3D Models from the Web. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*. IEEE Press. http://aura.abdn.ac.uk/handle/2164/9430

[7] James G Greeno. 1994. Gibson's affordances. (1994).

[8] Simon Hangl, Emre Ugur, Sandor Szedmak, and Justus Piater. 2016. Robotic playing for hierarchical complex skill learning. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2799–2804. https://doi.org/10.1109/IROS.2016.7759434 arXiv:1603.00794

[9] Karol Hausman, Scott Niekum, Sarah Osentoski, and Gaurav S. Sukhatme. 2015. Active articulation model estimation through interactive perception. In *Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 3305–3312. https://doi.org/10.1109/ICRA.2015.7139655

[10] Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. 2017. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research* (jun 2017), 027836491771031. https://doi.org/10.1177/0278364917710318

[11] Manuel Lopes, Francisco S Melo, and Luis Montesano. 2007. Affordance-based imitation learning in robots. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1015–1021. https://doi.org/10.1109/IROS.2007.4399517

[12] Chaitanya Mitash, Kostas E. Bekris, and Abdeslam Boularias. 2017. A Self-supervised Learning System for Object Detection using Physics Simulation and Multi-view Pose Estimation. In *2017 IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE.

[13] L. Pinto and A. Gupta. 2017. Learning to push by grasping: Using multiple tasks for effective learning. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2161–2168. https://doi.org/10.1109/ICRA.2017.7989249

[14] Morgan Quigley, Ken Conley, Brian P Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. 2009. ROS: an open-source Robot Operating System. In *2009 IEEE International Conference on Robotics and Automation (ICRA), Workshop on Open Source Software*.

[15] Gideon Schwarz. 1978. Estimating the Dimension of a Model. *The Annals of Statistics* 6, 2 (mar 1978), 461–464. https://doi.org/10.1214/aos/1176344136

[16] Jaeyong Sung, Seok Hyun Jin, and Ashutosh Saxena. 2018. *Robobarista: Object Part Based Transfer of Manipulation Trajectories from Crowd-Sourcing in 3D Pointclouds*. Springer International Publishing, Cham, 701–720. https://doi.org/10.1007/978-3-319-60916-4_40

[17] Andrea L. Thomaz and Maya Cakmak. 2009. Learning about objects with human teachers. In *Proceedings of the 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI '09)*. ACM Press, New York, New York, USA, 15. https://doi.org/10.1145/1514095.1514101

[18] E. Todorov, T. Erez, and Y. Tassa. 2012. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 5026–5033. https://doi.org/10.1109/IROS.2012.6386109

[19] Emre Ugur and Justus Piater. 2015. Bottom-up learning of object categories, action effects and logical rules: From continuous manipulative exploration to symbolic planning. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2627–2633. https://doi.org/10.1109/ICRA.2015.7139553