

# Learning to See before Learning to Act: Visual Pre-training for Manipulation

Lin Yen-Chen<sup>12</sup> Andy Zeng<sup>1</sup> Shuran Song<sup>13</sup> Phillip Isola<sup>2</sup> Tsung-Yi Lin<sup>1</sup>

<sup>1</sup>Google, Brain team <sup>2</sup>Massachusetts Institute of Technology <sup>3</sup>Columbia University

**Abstract**—Does having visual priors (e.g. the ability to detect objects) facilitate learning to perform vision-based manipulation (e.g. picking up objects)? We study this problem under the framework of transfer learning, where the model is first trained on a passive vision task (i.e., the data distribution does not depend on the agent’s decisions), then adapted to perform an active manipulation task (i.e., the data distribution *does* depend on the agent’s decisions). We find that pre-training on vision tasks significantly improves generalization and sample efficiency for learning to manipulate objects. However, realizing these gains requires careful selection of which parts of the model to transfer. Our key insight is that outputs of standard vision models highly correlate with *affordance maps* commonly used in manipulation. Therefore, we explore directly transferring model parameters from vision networks to affordance prediction networks, and show that this can result in successful zero-shot adaptation, where a robot can pick up certain objects with zero robotic experience. With just a small amount of robotic experience, we can further fine-tune the affordance model to achieve better results. With just 10 minutes of suction experience or 1 hour of grasping experience, our method achieves  $\sim 80\%$  success rate at picking up novel objects.

## I. INTRODUCTION

The development of embodied cognition happens incrementally [1], with infants’ early experiences strongly ordered by the stages of development of their sensorimotor systems. After 3 or 4 months of looking at the world *passively*, infants begin to reach for objects *actively*. This staged nature of development organizes experiences in a natural curriculum that shifts supervisory signals from observation to interaction.

In contrast, recent breakthroughs on vision-based manipulation [2, 3, 4, 5] often ignore the staged nature of cognitive development and instead directly supervise control policies via interactions between robots and environments. This causes two major problems. First, the learning process is slow and expensive: learning a new manipulation skill usually demands hundreds to thousands of real-world interactions. Second, the learned policy does not generalize well to unseen objects, because it has only interacted with a limited number of training objects in a specific environment.

In this paper, we aim to answer the question: does having visual priors (e.g. the ability to detect objects) facilitate learning to perform vision-based manipulation (e.g. picking up objects)? We adopt the framework of transfer learning, where the model is first trained on a source task, and then reused to fit the target task. Specifically, we study the problem of transferring a vision model learned from visual observations

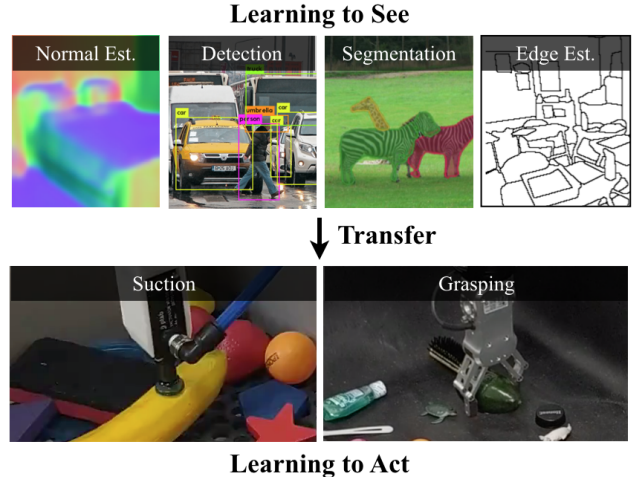


Fig. 1. **Overview.** Does first learning to see aid the speed at which a robot can learn to act? In this paper, we show how to transfer knowledge from pre-trained vision tasks to robotic manipulation policies, demonstrating large improvements in the sample efficiency and generalization ability of robot learning.

to an affordance model for a manipulation task. Although transfer learning has already shown many promising results on vision tasks, it remains unclear how to transfer knowledge learned from *passive* vision tasks (i.e., the data distribution is independent of the agent’s decisions) to *active* robotics tasks (i.e., the data distribution depends on the agent’s decisions), where good exploration strategies are vital.

One obvious solution to this problem is to transfer the parameters of a shared model architecture, usually termed a “backbone”, while randomly initializing task-specific convolutional or linear layers, which are commonly called “heads”, on top of it. Essentially, this approach transfers the pre-trained latent feature representations of vision models to affordance models. However, Zeng et al. [6] found that transferring latent feature representations from a model pre-trained on ImageNet does not accelerate the learning process or converge to better performance for pushing-grasping tasks.

In our work, we find that only transferring latent features for learning affordance models results in bad *exploration*. The key problem is that by randomly initializing the head of the affordance model, the resulting policy, even with pre-trained latent features, still *randomly* explores the environments and thus fails to collect useful supervisory signals. On the contrary, humans are able to take advantage of a set of perceptual biases (e.g. object-centric world structure) to aid the exploration of possible interactions in the environments.

For example, babies do not randomly try all possible actions, grasping haphazardly at the air; they are biased toward interacting preferentially with the most obvious objects [7]. Inspired by this observation, we propose to transfer the entire vision model, including both features from the backbone and the visual predictions from the head, to initialize the affordance model. In this case, the resulting initial policy is simply the visual prediction model pre-trained on passive observations.

We show that such vision-guided exploration greatly reduces the number of interactions needed to acquire a new manipulation skill, then we explore which visual predictions can be used as a good initialization for the affordance model. We experiment with a wide variety of vision tasks and find that initializing the affordance model with most of them results in significantly faster convergence speed compared to a randomly initialized model, which suggests that many vision tasks can be helpful for learning the affordance model. Based on our results, we believe that learning to see from abundant visual observations before learning to act from expensive robot-environment interactions is a key step to improve the generalization ability and sample efficiency of vision-based manipulation.

Our contributions are three-fold: 1) we explore the problem of transferring vision models to perform robotics tasks and show that transferring both the backbone and the head of networks leads to better performance compared to only transferring the backbone, 2) we systematically benchmark the effectiveness of pre-training on different vision tasks for learning grasping and suction, 3) we offer a simple and powerful initialization strategy to turn an off-the-shelf vision model into the initialization of an affordance model without additional data collection. Specifically, this strategy enables a robot to learn to perform suction in 10 minutes and grasping in an hour with an approximately 80% success rate on unseen objects.

## II. RELATED WORK

**Offline Computer Vision.** Computer vision tasks focus on estimating environment properties, e.g. image classification [8], detection [9], segmentation [10, 11], and pose estimation [12]. Although they focus on estimating different latent properties, the learning process happens on prerecorded datasets. In this work, we study how to transfer the knowledge learned from prerecorded vision datasets to solve downstream manipulation tasks that require agents to collect their own datasets actively.

**Self-supervised Robot Learning.** Over the last few years, there has been growing interest in scaling up robot data collection through self-supervised learning. This approach has been applied to learn different robotic skills including grasping [2, 13, 14, 15, 16], pushing [17, 18, 19], tossing [3], and obstacle avoidance [20, 21]. Specifically for grasping, Pinto et al. [2] uses background subtraction to reduce the number of random grasping trials in empty space and shows that it improve sample efficiency. Building on this prior work, we instead explore how transferring model weights can

improve not only sample efficiency but also generalization to unseen objects.

**Transfer Learning.** Previous works on transfer learning for robotics focus on reusing skills learned from different environments [22] or bridging the gap between simulation and reality [23, 24, 25, 26]. In our work, we study the problem of how to transfer models learned from passive vision tasks to learn active manipulation tasks.

**Representation Learning.** The field of unsupervised learning has explored different ways to learn state representations [27, 28, 29] for policy learning. Recently, several works [30, 31, 32] have studied the benefits of combining various mid-level visual representations for reinforcement learning. Different from previous works, we 1) demonstrate real-world results on manipulation while previous works present simulation results on navigation, 2) highlight the importance of using a pre-trained model for exploration, and 3) propose an initialization strategy without the need of data collection at target domain.

## III. METHOD

In this section, we introduce the input representation, the vision models, and manipulation affordance models (Figure 3), followed by the details of transfer learning.

### A. Input Representation

We represent the visual input  $I$  as an RGB-D heightmap image of the workspace (i.e., a bin of objects). To obtain the heightmap, we follow the pre-processing steps in Zeng et al. [6] by first projecting the RGB-D image captured by the camera onto a 3D point cloud, and orthographically back-project upwards in the gravity direction. The RGB and D channels are both normalized (mean-subtracted and divided by standard deviation) before being fed into the model. In our real-world setup for suction, images captured by the camera cover a  $0.36 \times 0.26\text{m}$  area and have a pixel resolution of  $288 \times 208$ . For grasping, we have the camera cover a  $0.5 \times 0.5\text{m}$  area with pixel resolution of  $400 \times 400$ . In both cases, each pixel represents a  $1.25 \times 1.25\text{mm}$  vertical column in the robot’s workspace.

### B. Learning Vision Models

We follow the common practice in [10] to design the vision model using ConvNet. The model has a heavy backbone model for extracting visual features and a lightweight vision head, which usually consists of one or a few convolution layers for making visual prediction from features of a backbone model. The visual prediction is a 2D heatmap that can represent a wide variety of vision tasks.

In this work, we consider the following vision tasks: 1) edge detection, 2) corner detection, 3) object center detection, 4) foreground segmentation, and 5) surface normal estimation. Specifically, we transform surface normal estimation as identifying pixels whose normal are similar to the anti-gravity direction. We collect the dataset by letting robots randomly interact with the environment. The annotations for each task are obtained through simple computer

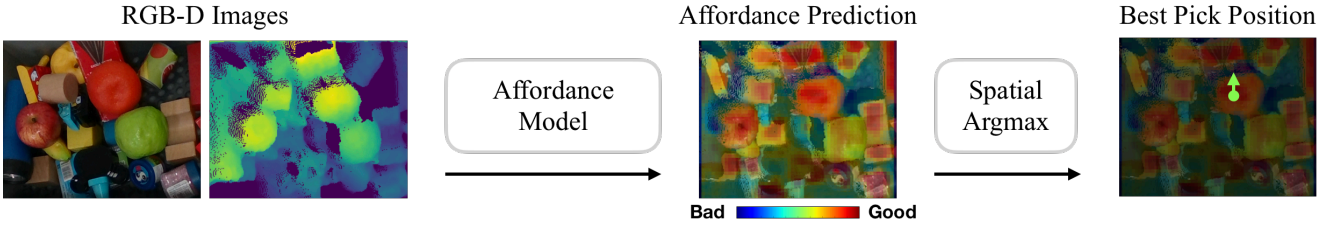


Fig. 2. **Illustration of affordance-based manipulation.** Given an RGB-D image, we predict pixel-wise affordance with an affordance model. At each pixel, the predicted value of affordance represents the success rate of performing the corresponding motion primitive at that pixel’s projected 3D position. The motion primitive is executed by the robot at the position with the highest affordance value.

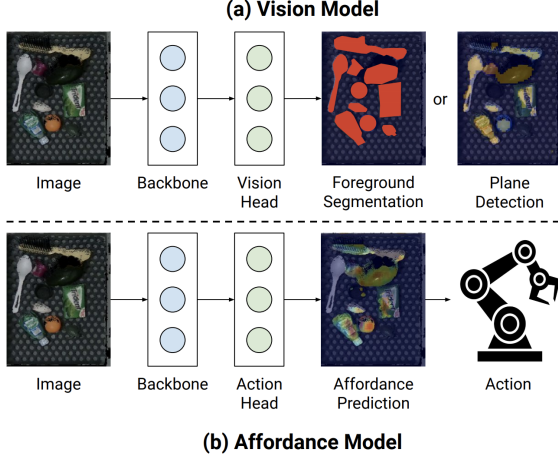


Fig. 3. **Comparison of vision and affordance models.** Both vision and affordance models take RGB-D image as input and process it through backbone and head to output predictions. The major difference is that affordance models’ predictions will then be executed by the robot with corresponding motion primitive to affect the future data distribution.

vision techniques such as Canny edge detection [33], Harris corner detection [34], and background subtraction. In our real-world setup, we also experiment with additional vision models trained with ImageNet classification and COCO object detection, which do not contain any domain and task-specific training data from the environment where the robot is deployed.

To train the vision system, we apply binary cross-entropy loss:  $L = -y_i \log f(I_i) - (1 - y_i) \log (1 - f(I_i))$ , where  $I_i$  is the input RGB-D image,  $y_i$  is the binary ground truth label, and  $f$  is the vision model. Gradients are computed over all the pixels in the input RGB-D image.

### C. Learning Manipulation Affordances

The affordance model, which consists of a ConvNet and a motion primitive, is illustrated in Figure 2. The ConvNet learns to predict a dense 2D heatmap that encodes the probability of picking success at each pixel location. The motion primitive is a pre-defined function which controls the robot arm to perform manipulation task from a fixed initial arm position. Different manipulation tasks need their own motion primitive. The motion primitive is open-loop, with motion planning executed using a stable, collision-free IK solver.

**Suction primitive.** The suction primitive takes as input parameters  $\phi_s = (p)$  and executes a top-down suction centered at a 3D position  $p = (p_x, p_y, p_z)$ . During execution, the suction cup approaches  $p$  along the gravity direction until the 3D position of the suction cup meets  $p$ , at which point the suction activates, and lifts upwards 10cm. Picking success is then determined by thresholding on the difference between digital vacuum pressure readings before and after the primitive, which serves as a sufficient signal to determine whether the suction contact resulted in a good seal.

**Grasping primitive.** The grasping primitive takes as input parameters  $\phi_g = (p, \theta)$  and executes a top-down parallel-jaw grasp centered at a 3D position  $p = (p_x, p_y, p_z)$  oriented  $\theta$  around the gravity direction. In order to get  $\theta$ , we rotate the input heightmap by 16 different angles and feed all of them into our model to estimate the affordance of horizontal grasps for each heightmap. We then perform a max-pooling across all heightmaps to identify the best grasping position and orientation. During execution, the open gripper approaches  $p$  along the gravity direction until the 3D position of the middle point between the gripper fingertips meets  $p$ , then the gripper closes and lifts upwards 10cm. Picking success is determined by thresholding on the measured width between gripper fingertips.

As the data for learning affordances comes from robot-environment interactions, it is not independent and identically distributed. Thus, we curate incoming data in an experience replay buffer and perform prioritized experience replay to construct inputs for each iteration of training. To train the affordance model, we again adopt binary cross-entropy loss. In this case,  $I_i$  is the input RGB-D image,  $y_i$  is the binary ground truth picking success label, and  $f$  is the affordance model. Since we only have the label of the single pixel on which the picking primitive was executed, gradients are only passed through that pixel. All other pixels have zero loss.

### D. Transferring Vision Model to Affordance Model

To facilitate the transfer, we design vision and affordance models to share the same backbone and head architecture. The learned parameters of the vision model are directly taken to initialize parameters of affordance model. We study two types of vision to affordance transfer: 1) transfer of backbone model; 2) transfer of backbone and head models. It is common to only transfer backbone model, especially

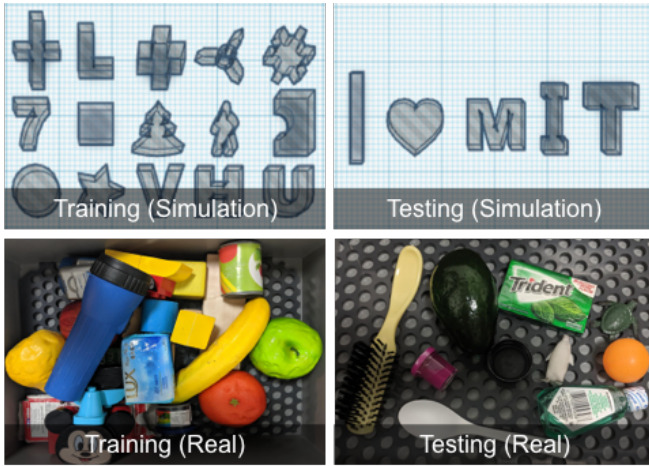


Fig. 4. **Objects used in experiments.** We show objects in simulated (top) and real (bottom) experiments, split by training objects (left) and testing objects (right) which are unseen during training.

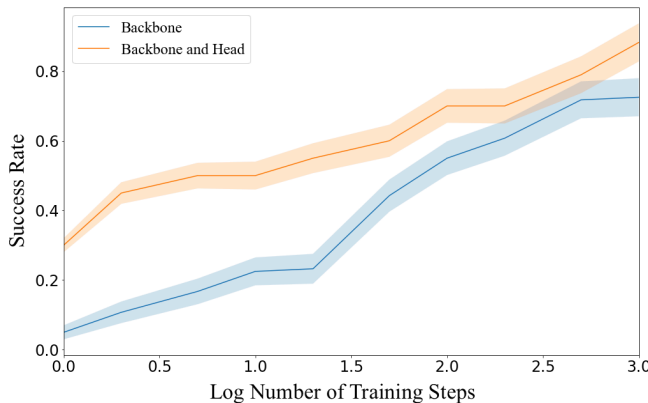


Fig. 5. **Performance of transfer learning.** Transferring entire model (backbone + head) achieved a success rate of 86% while partial model (backbone only) merely achieved 71%. We report the mean and 95% confidence interval of the performance over five runs.

when the head architectures are not compatible, e.g., image classification does not have the same architecture as affordance prediction head so one can only transfer the backbone of ImageNet pre-trained model. However, we show that transferring the entire vision model, including both backbone and head, are critical for improving sample efficiency for learning the affordance model. The intuition of transferring the head is that visual predictions are closely related to affordance of manipulation (see Figure 8). For example, picking the object center can be a good manipulation policy for apple. In our experiments, we find that exploring and fine-tuning with respect to visual predictions greatly speed up the learning.

#### IV. EXPERIMENTS

We executed a series of experiments in both simulated and real settings (conducted in isolation) to evaluate how well trained vision models can contribute to the performance of manipulation affordance models – in particular, for suction and grasping. The goal of our experiments is three-fold: 1) to investigate whether transferring only a part of the network (i.e., backbone) versus the entire network (i.e., including both

backbone and head) leads to improved manipulation performance, 2) to benchmark the value of various visual task(s) by how much they contribute to manipulation performance, and 3) to determine which large-scale real-world datasets result in the most improvement for manipulation performance.

**Evaluation protocol.** We evaluate the performance of our manipulation affordance models for suction and grasping by measuring picking success rate  $\frac{\# \text{ successful picks}}{\# \text{ picking attempts}}$  i.e., the rate which an object remains in the end-effector after executing the respective primitive. In all experiments, we report the total picking success rate per test run after executing a maximum of 25 picking attempts with seen objects used for training and 15 picking attempts with unseen objects. If the robot successfully picks up all available objects before reaching the maximum count of picking attempts, the test run ends. Prior to each test run, we reset the environment to a pre-defined canonical cluttered arrangement of objects, unseen during training. During evaluation, model gradients are disabled.

##### A. Simulation Experiments

Our simulation environment in PyBullet uses a UR5 robot arm with a Robotiq 2F-85 gripper. We simulate picking experiments with 20 different 3D object models: 15 seen during training and 5 unseen for testing (illustrated in Figure 4). These objects were chosen to maximize shape diversity. Two copies of each 5 object (10 in total) are randomly colored, scaled, and dropped into the bin during training and testing. For our simulation experiments, we follow Zeng et al. [3] to use a 7-layer fully convolutional residual network as backbone network architecture.

**Transfer partial or entire vision model?** Our first experiment compares transferring pre-trained weights from a partial backbone model versus pre-trained weights from both the backbone and vision head. This comparison is interesting because substantial prior work in computer vision uses the former pre-training paradigm for better performance – we test to see if this assumption still holds true for vision-to-manipulation transfer. Specifically, we train a foreground segmentation model in simulation (for which ground truth is available), use its weights to initialize a grasping affordance model, then fine-tune the affordance modes through trial and error. Figure 5 plots the picking success rates over training steps during fine-tuning using each of the aforementioned pre-training paradigms.

Interestingly, we observe that initializing our manipulation affordance models using pre-trained weights from both the backbone model and vision head yields the strongest overall picking performance. This is likely because the pre-trained weights from the vision head help to positively bias grasp samples predicted by the manipulation model towards foreground objects. This decreases the number of randomly predicted grasps in the background (as would be the case if the head was randomly initialized) early in training, and thus improves exploration and sample efficiency.

**Which vision tasks to transfer?** We next investigate the benefits of pre-training on various different vision tasks,



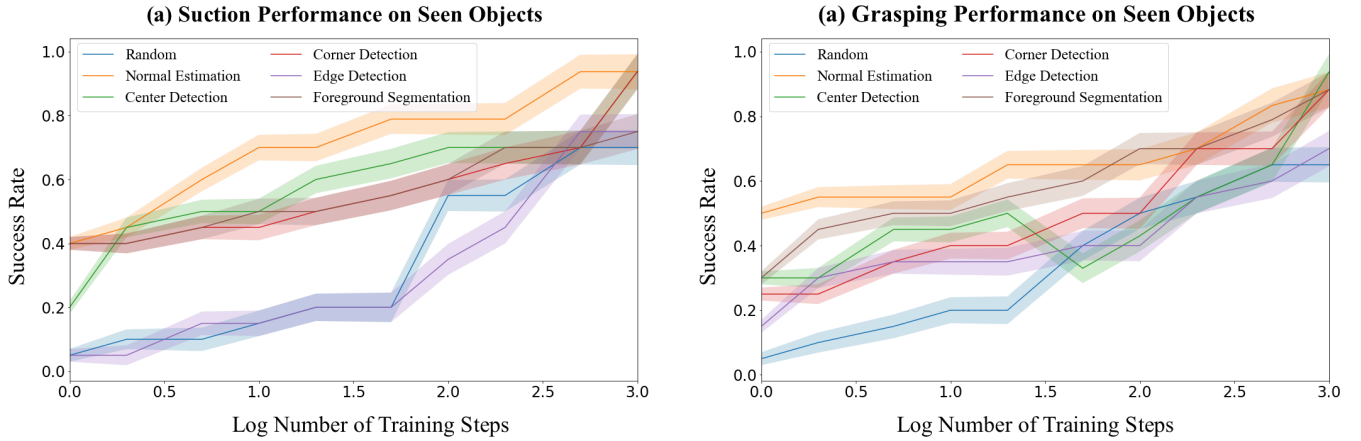


Fig. 6. **Simulation results of suction and grasping with models pre-trained on different vision tasks.** We report the mean and 95% confidence interval of the performance over five runs.

for which we can obtain ground truth training labels in simulation. These tasks include: 1) object center detection, 2) edge detection, 3) corner detection, 4) foreground segmentation, and 5) flat surface detection. We transfer pre-trained weights from these tasks to suction and grasping affordance models, then report their picking success rates while fine-tuning with trial and error on the training objects in simulation in Figure 6. We also execute test runs with novel unseen objects using the fine-tuned affordance models, and report their performance in Table I.

TABLE I  
SIMULATION PERFORMANCE ON UNSEEN OBJECTS.

Vision Tasks	Suction	Grasping
Random	0.71	0.71
Normal estimation	1.00	0.91
Center detection	0.83	0.91
Corner detection	0.83	0.91
Edge detection	0.90	0.83
Foreground segm.	0.83	0.91

From these results, we find that for the suction affordance model, pre-training on the vision task of flat surface detection drastically improves its sample efficiency. This is as expected since suction performs better (both in simulation and in real-world) on flat object surfaces than on other geometries. For the grasping affordance model, pre-training on foreground segmentation and flat surface detection outperforms other variants, while the former task enables the model to generalize better to unseen objects. This is surprising since we expected corner and edge detection to provide features that are more relevant for grasping. However, our results indicate that the grasping affordance model benefits most from understanding the distinction between foreground and background pixels.

### B. Real-world Experiments

Our real-world setup consists of a UR5 robot arm, equipped with an RG2 gripper for grasping, a 3D-printed end effector with a B-Bellows DURAFLEX suction cup for suction, and a statically mounted Intel RealSense D415 overlooking a bin of objects from the side. The camera

captures  $1280 \times 720$  RGB-D images and is localized with respect to the robot base using an automatic calibration procedure from [6]. Our collection of objects includes 25 different toy blocks, fake fruit, decorative items, and office objects (see bottom left of Figure 4) during training. We use 10 additional objects (see bottom right of Figure 4) for evaluating our model on its generalization to unseen objects. For our real-world experiments, we use the region proposal network (RPN) model architecture in [35]. We adopt ResNet-50 FPN [36] as the backbone model and prediction layers on  $P2$  feature level as the head model. The multi-anchor predictions are collapsed by max-pooling on confidence scores to generate a single heatmap. To investigate the effects of vision tasks, we also include RPNs that are pre-trained on foreground segmentation and normal estimation as baselines. The data consists of 1000 cluttered observations with labels computed from depth images.

**Can off-the-shelf vision models help?** In our real-world experiments, we train vision models with the best-performing vision tasks from our simulation experiments (Section IV-A) and compare them with random initialization to verify our findings in simulation. We then explore whether off-the-shelf vision models can serve as good initialization for learning suction and grasping affordances. These off-the-shelf pre-trained models include: a) ImageNet, b) COCO-backbone, c) COCO, and d) COCO-fix, where both ImageNet and COCO-backbone only transfer backbone network weights. COCO and COCO-fix transfer both backbone and head network weights but COCO-fix is not trained. Note that off-the-shelf vision models are not fine-tuned on the target domain and thus don't require additional labels. We transfer pre-trained weights from these tasks to learn suction and grasping affordance models, then report picking success rates while fine-tuning with trial and error on the training objects in Figure 7. We also execute test runs with novel unseen objects (including transparent objects [37]) using the affordance models and report their performance in Figure 7. These results highlight three key observations: 1) we validate that our findings in simulation apply to our real-world setting, 2) off-the-shelf vision models generalize better than

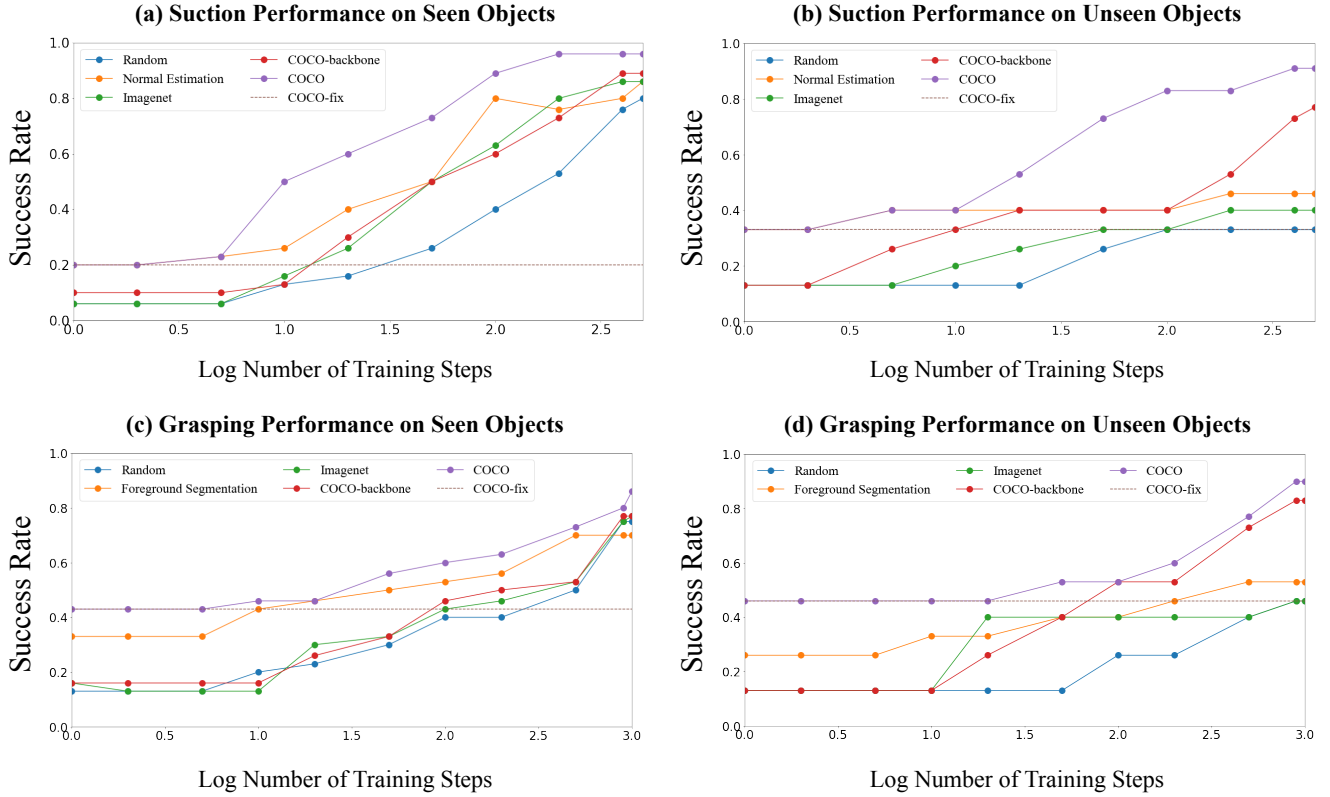


Fig. 7. Real-world results of suction and grasping with different pre-trained models.

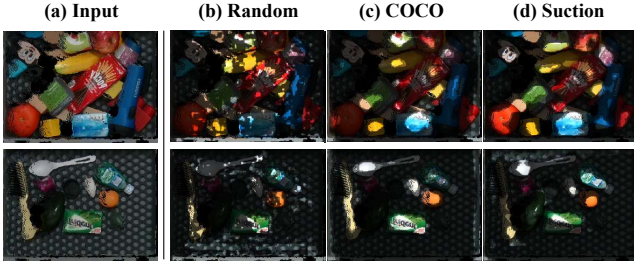


Fig. 8. Affordances predicted by different models. We show the affordances of suction predicted by different models given input from (a). (b) Random refers to a randomly initialized model. (c) COCO is the modified RPN model. (d) Suction is a converged model learned from robot-environment interactions.

vision models trained in the target environment (COCO-fix vs. Random on unseen objects, Figure 7b,d) 3) models initialized with COCO features perform better than models initialized with ImageNet features. Finding 2) is encouraging as it shows that learning from diverse offline observations (i.e., COCO dataset) can improve both sample efficiency and generalization when learning manipulation affordances. For finding 3), a possible explanation is that the features learned from object detection dataset are able to localize objects, which is crucial for manipulation; whereas models trained with image classification lose this spatial information.

**Visualization analysis.** To understand why vision tasks transfer well for learning affordances, we show the heatmap visualization of vision and affordance models (where hotter regions indicate higher predicted values). As shown in Figure 8, the Random and ImageNet models output random heatmaps since the prediction head is randomly initialized.

The heatmap of Normal and COCO models show strong object-centric activations, with an output distribution very similar to that of trained suction models. The heatmap visualization supports our intuition that vision and affordance tasks share strong similarities.

## V. CONCLUSION AND FUTURE WORK

In this work, we explore the extent to which pre-trained vision models (from passive observations) can benefit the process of learning affordance models (for active manipulation). We present a method that can significantly reduce the number of interactions required for learning manipulation policy by first learning visual predictions. A wide variety of visual tasks is studied in this work and we show that learning visual predictions not only provide better features but also provide good initial manipulation strategy. The real-world experiments show that off-the-shelf vision model provides generic vision features across domains, improves both training speed and final performance for learning manipulation in a new environment. We imagine the benefit of visual pre-training is not only limited for affordance based manipulation model. A future research direction is to apply vision-guided exploration to action model training with reinforcement learning, such as [38], where the data efficiency is the bottleneck of learning.

## ACKNOWLEDGEMENTS

We would like to thank the team members of Robotics at Google for insightful discussions and technical support.

# REFERENCES

- [1] L. Smith and M. Gasser, “The development of embodied cognition: Six lessons from babies,” *Artificial life*, vol. 11, no. 1-2, pp. 13–29, 2005.
- [2] L. Pinto and A. Gupta, “Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours,” in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 3406–3413.
- [3] A. Zeng, S. Song, J. Lee, A. Rodriguez, and T. Funkhouser, “Tossingbot: Learning to throw arbitrary objects with residual physics,” *arXiv preprint arXiv:1903.11239*, 2019.
- [4] S. Song, A. Zeng, J. Lee, and T. Funkhouser, “Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations,” *arXiv preprint arXiv:1912.04344*, 2019.
- [5] A. Zeng, “Learning visual affordances for robotic manipulation,” Ph.D. dissertation, Princeton University, 2019.
- [6] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, “Learning synergies between pushing and grasping with self-supervised deep reinforcement learning,” *arXiv preprint arXiv:1803.09956*, 2018.
- [7] S. P. Johnson, D. Amso, and J. A. Slemmer, “Development of object concepts in infancy: Evidence for early learning in an eye-tracking paradigm,” *Proceedings of the National Academy of Sciences*, vol. 100, no. 18, pp. 10 568–10 573, 2003.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [9] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.
- [10] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.
- [11] S. Song, A. Zeng, A. X. Chang, M. Savva, S. Savarese, and T. Funkhouser, “Im2pano3d: Extrapolating 360 structure and semantics beyond the field of view,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3847–3856.
- [12] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7291–7299.
- [13] L. Pinto, D. Gandhi, Y. Han, Y.-L. Park, and A. Gupta, “The curious robot: Learning visual representations via physical interactions,” in *European Conference on Computer Vision*. Springer, 2016, pp. 3–18.
- [14] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, “Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection,” *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018.
- [15] R. Calandra, A. Owens, M. Upadhyaya, W. Yuan, J. Lin, E. H. Adelson, and S. Levine, “The feeling of success: Does touch sensing help predict grasp outcomes?” *arXiv preprint arXiv:1710.05512*, 2017.
- [16] K. Zakka, A. Zeng, J. Lee, and S. Song, “Form2fit: Learning shape priors for generalizable assembly from disassembly,” *arXiv preprint arXiv:1910.13675*, 2019.
- [17] C. Finn and S. Levine, “Deep visual foresight for planning robot motion,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 2786–2793.
- [18] P. Agrawal, A. V. Nair, P. Abbeel, J. Malik, and S. Levine, “Learning to poke by poking: Experiential learning of intuitive physics,” in *Advances in Neural Information Processing Systems*, 2016, pp. 5074–5082.
- [19] F. Ebert, C. Finn, A. X. Lee, and S. Levine, “Self-supervised visual planning with temporal skip connections,” *arXiv preprint arXiv:1710.05268*, 2017.
- [20] G. Kahn, A. Villafior, V. Pong, P. Abbeel, and S. Levine, “Uncertainty-aware reinforcement learning for collision avoidance,” *arXiv preprint arXiv:1702.01182*, 2017.
- [21] D. Gandhi, L. Pinto, and A. Gupta, “Learning to fly by crashing,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3948–3955.
- [22] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, “Progressive neural networks,” *arXiv preprint arXiv:1606.04671*, 2016.
- [23] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, “Domain randomization for transferring deep neural networks from simulation to the real world,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 23–30.
- [24] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke, “Sim-to-real: Learning agile locomotion for quadruped robots,” *arXiv preprint arXiv:1804.10332*, 2018.
- [25] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, “Sim-to-real transfer of robotic control with dynamics randomization,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8.
- [26] A. A. Rusu, M. Vecerik, T. Rothörl, N. Heess, R. Pascanu, and R. Hadsell, “Sim-to-real robot learning from pixels with progressive nets,” *arXiv preprint arXiv:1610.04286*, 2016.
- [27] I. Higgins, A. Pal, A. Rusu, L. Matthey, C. Burgess, A. Pritzel, M. Botvinick, C. Blundell, and A. Lerchner, “Darla: Improving zero-shot transfer in reinforcement learning,” in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR.org, 2017, pp. 1480–1490.

- [28] D. Ghosh, A. Gupta, and S. Levine, “Learning actionable representations with goal-conditioned policies,” *arXiv preprint arXiv:1811.07819*, 2018.
- [29] Z. Xu, J. Wu, A. Zeng, J. B. Tenenbaum, and S. Song, “Densephysnet: Learning dense physical object representations via multi-step dynamic interactions,” *arXiv preprint arXiv:1906.03853*, 2019.
- [30] B. Zhou, P. Krähenbühl, and V. Koltun, “Does computer vision matter for action?” *arXiv preprint arXiv:1905.12887*, 2019.
- [31] A. Sax, B. Emi, A. R. Zamir, L. Guibas, S. Savarese, and J. Malik, “Mid-level visual representations improve generalization and sample efficiency for learning active tasks,” *arXiv preprint arXiv:1812.11971*, 2018.
- [32] W. B. Shen, D. Xu, Y. Zhu, L. J. Guibas, L. Fei-Fei, and S. Savarese, “Situational fusion of visual representation for visual navigation,” in *IEEE International Conference on Computer Vision (ICCV)*, 2019.
- [33] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Jun. 1986. [Online]. Available: <https://doi.org/10.1109/TPAMI.1986.4767851>
- [34] C. Harris and M. Stephens, “A combined corner and edge detector,” in *In Proc. of Fourth Alvey Vision Conference*, 1988, pp. 147–151.
- [35] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He, “Detectron,” <https://github.com/facebookresearch/detectron>, 2018.
- [36] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [37] S. S. Sajjan, M. Moore, M. Pan, G. Nagaraja, J. Lee, A. Zeng, and S. Song, “Cleargrasp: 3d shape estimation of transparent objects for manipulation,” *arXiv preprint arXiv:1910.02550*, 2019.
- [38] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine, “Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation,” *CoRR*, vol. abs/1806.10293, 2018. [Online]. Available: <http://arxiv.org/abs/1806.10293>