Learning grasping interaction with geometry-aware 3D representations

Xinchen Yan∗†, Mohi Khansari†‡, Yunfei Bai†, Jasmine Hsu†, Arkanath Pathak§
Abhinav Gupta∗, James Davidson‡, Honglak Lee‡
†Google Brain, ‡X Inc, ×Google Research, §Google, ∗University of Michigan
xcyan@umich.edu, {khansari, yunfeibai}@x.team
{hellojas, arkanath, abhinavgu, jcdavidson, honglak}@google.com

Abstract

Learning to interact with objects in the environment is a fundamental AI problem involving perception, motion planning, and control. However, learning representations of such interactions is very challenging due to a high dimensional state space, difficulty in collecting large-scale data, and many variations of an object’s visual appearance (i.e., geometry, material, texture, and illumination). We argue that knowledge of 3D geometry is at the heart of grasping interactions and propose the notion of a geometry-aware learning agent. Our key idea is constraining and regularizing interaction learning through 3D geometry prediction. Specifically, we formulate the learning process of a geometry-aware agent as a two-step procedure: First, the agent learns to construct its geometry-aware representation of the scene from 2D sensory input via generative 3D shape modeling. Finally, it learns to predict grasping outcome with its built-in geometry-aware representation. The geometry-aware representation plays a key role in relating geometry and interaction via a novel learning-free depth projection layer. Our contributions are threefold: (1) we build a grasping dataset from demonstrations in virtual reality (VR) with rich sensory and interaction annotations; (2) we demonstrate that the learned geometry-aware representation results in a more robust grasping outcome prediction compared to a baseline model; and (3) we demonstrate the benefits of the learned geometry-aware representation in grasping planning.

1 Introduction

Learning to interact with objects is a fundamental and challenging problem in artificial intelligence that involves perception, motion planning, and control. The problem is challenging because it not only requires understanding geometry (global shape of object, local surface around interaction) but it also requires estimating physical properties, such as weight of object, density and friction. Furthermore, it requires invariance to illumination, objects’ location and viewpoint. To handle this, current data-driven approaches use thousands of examples and learn end-to-end models. But collecting such large-scale data is extremely difficult and time-consuming. Is it possible to constrain the learning process somehow so that we can learn representation for interaction with less data?

We argue that geometry is at the heart of this type of interaction and propose the concept of geometry-aware learning agent with the following properties: (1) agent has a clear notion of the geometry: the location, orientation, and shape of object from visual input. Therefore, the agent is able to distinguish its ego-motion from the motions in the environment. (2) agent is able to relate the geometry of novel object with its previous experience and (3) agent is able to reason about the relationship between interaction and feedback (i.e., success or failure) by taking the geometry factors into consideration.

†The work was done while interning at Google Brain.
Whether this is a valid grasp? Yes or No. This is a valid grasp!

Geometry-aware representation
Global shape Local shape

Camera setting Encoder-decoder neural networks Learning-free OpenGL projection layer

Gripper setting

Visualization 1 Visualization 2 Visualization 3

Camera

Figure 1: Learning grasping interactions from demonstrations.

Compared to a learning agent that does not have explicit notion of geometry, we believe that the geometry-aware agent will learn a better understanding of interaction, feedback and geometry.

In this work, we propose a two-stage procedure for learning grasping interaction from demonstrations. First, the agent learns to build the geometry-aware representation from 2D visual input. Second, the agent learns to predict grasping interaction from demonstrations with the built-in geometry-aware representation. More specifically, we design an encoder-decoder deep neural network for learning this representation. Our geometry-aware encoder-decoder network has two components: a shape prediction network and a grasping outcome prediction network. The shape prediction network has an image encoder, a 3D shape decoder, and a learning-free OpenGL projection layer. The image encoder transforms the 2D visual data into the high-level geometry representation. The shape decoder network takes in the geometry representation and outputs the 3D volume of the object. To enable supervision with only 2D visual data, we propose a novel learning-free OpenGL projection layer similar to Yan et al. [2016], Rezende et al. [2016]. The grasping outcome prediction network has a state encoder and an outcome predictor. The state encoder network transforms the current visual state (e.g., object and gripper) to a high-level state representation. The outcome predictor network takes in an action, a state, and the geometry representations to produce an outcome (e.g., success or failure) of the grasping interaction.

We have built a large database consisting of 101 everyday objects with more than 150K grasping demonstrations in Virtual Reality (VR) with both human and artificial interactions. For each object, we collect 10-20 grasping attempts with a 1-DoF virtual gripper from right-handed users. For each attempt, we record a pre-grasp status which includes the location and orientation of the object and gripper, as well as the grasping outcome (e.g., success or failure). Additionally, we augment the data by perturbing the gripper location and orientation based on grasping demonstrations in the Bullet Physics simulator.

Our main contributions are summarized below:

- We build a database with rich visual sensory data and grasping annotations.
- We demonstrate that the proposed geometry-aware encoder-decoder network is able to learn the shape as well as grasping outcome better than models without notion of geometry.
- We demonstrate that the proposed model has advantages in guiding grasping exploration and achieves better generalization to novel viewpoints and novel object instances.

2 Related Work

A common approach for robotic grasping is to detect the optimal grasping location from 2D visual input (RGB or RGBD images) Saxena et al. [2008], Montesano and Lopes [2012], Lenz et al. [2015].
3 Multi-objective framework with geometry-aware representation

In this section, we develop a two-stage learning framework that performs 3D shape prediction and grasping outcome prediction with geometry-aware representation. Being able to generate 3D object shapes (e.g., volumetric representation) from any scene given 2D sensory input is a very important feature of our geometry-aware agent. More specifically, in our formulation, the geometry-aware representation is (1) an occupancy grid representation of the scene centered at camera target in the world frame and (2) invariant to camera viewpoint and distance.

3.1 Learning generative 3D geometry-aware representation from 2D sensory input

For simplicity, we begin with a single-view formulation. If the ground-truth 3D volumetric representation $V$ is given, we can fit a functional mapping $f^V : I \rightarrow V$ that approximates the 3D object shape from 2D sensory input $I$. However, the ground-truth 3D object shape (e.g., explicit supervision of 3D volume or occupancy grid) is not always directly available. Inspired by Rezende et al. [2016], Yan et al. [2016], we tackle the 3D shape learning in a weakly supervised manner without explicit 3D shape supervision. In Yan et al. [2016], an in-network projection layer is introduced for 3D shape learning from 2D masks (e.g., 2D silhouette of object). However, a 2D silhouette is an insufficient supervision signal (e.g., consider the concave shape) in robotic grasping. Therefore, we also consider a 2D depth map $D$ as the supervision signal for learning the object geometry. To find the correspondence between a 3D shape and 2D depth map, we introduce a learning-free projective operator similar to Yan et al. [2016] that implements the exact rendering procedure for 2D depth estimation from the 3D world.

We formulate the projective operation by $f^D : V \times P \rightarrow D$ that transforms a 3D shape into a 2D depth map with the camera transformation matrix $P$. Here, the camera transformation matrix decomposes as $P = K[R; t]$, where $K$ is the camera intrinsic matrix, $R$ is the camera rotation matrix, and $t$ is the camera translation vector. In our implementation, we also use a 2D silhouette as an object mask $M$ for learning. Empirically, we found that this additional objective makes the learning easier during training. Finally, given a 2D observation $I$ from a single-view, the loss function is defined as follows:

$$L^{single}_\theta = \lambda_D L^{depth}_\theta(D; D) + \lambda_M L^{mask}(\hat{M}; M)$$  \hspace{1cm} (1)

\footnote{Our design choice of using RGBD as an input signal is also motivated from the common availability of RGBD sensors in most robot platforms.}
Here, \( \lambda_D \) and \( \lambda_M \) are the constant coefficients for the depth and mask prediction terms, respectively.

**Learning-free projective operator for depth estimation.** Following the OpenGL camera transformation standard, for each point \( p_i^s = (x_i^s, y_i^s, z_i^s, 1) \) in 3D world frame, we compute the corresponding point \( p_i^m = (x_i^m, y_i^m, z_i^m, 1) \) in the normalized device coordinate system \((-1 \leq x_i^m, y_i^m, z_i^m \leq 1\)) using the transformation: \( p_i^m \sim Pp_i^s \). Here, the conversion from depth buffer \( z_i^s \) to real depth \( z_i^m \) is given by \( z_i^m = f^r(z_i^s) = -1/\alpha + z_i^m + \beta \) where \( \alpha = z_{near} - z_{far} / 2(z_{near} + z_{far}) \) and \( \beta = 2z_{near}z_{far} / (z_{near} + z_{far}) \). Here, \( z_{near} \) and \( z_{far} \) represents the far and near clipping planes of the camera.

Similar to the transformer network proposed in [Yan et al. 2016], Jaderberg et al. [2015], our learning-free projection can be considered as: (1) performing dense sampling from input volume (in the 3D world frame) to output volume (in normalized device coordinates); and (2) flattening the 3D spatial output across one dimension. Again, each 3D point \((x_i^n, y_i^n, z_i^n)\) in input volume \( V \in \mathbb{R}^{H \times W \times D} \) and corresponding point \((x_i^m, y_i^m, z_i^m)\) in output volume \( U \in \mathbb{R}^{H' \times W' \times D'} \) is related by the transformation matrix \( P \). Here, \((W, H, D)\) and \((W', H', D')\) are the width, height, and depth of the input and output volume, respectively. We summarize the dense sampling step and channel-wise flattening step in Eq. 2.

\[
U_j = \sum_{n} \sum_{m} \sum_{l} V_{nm} \max(0, 1 - |x_i^m - m|) \max(0, 1 - |y_i^m - n|) \max(0, 1 - |z_i^m - l|) \\
\mathcal{M}_{n'm'} = \max_{l'} U_{n'm'l'}
\]

\[
D_{n'm'} = \begin{cases} 
Z_{far}, & 
\text{if } \mathcal{M}_{n'm'} = 0 \\
\beta(n', l') - 1 \text{ where } l' = \arg \min_{l'}(U_{n'm'l'} > 0.5), & \text{otherwise}
\end{cases}
\]

Intuitively, the learning-free projective operator is performing ray-tracing along the projection axis.

**Learning from multi-view observations.** Learning to predict 3D shape from single-view 2D sensory input is a challenging task in computer vision due to shape ambiguity. To reduce ambiguity in shape prediction, we assume multiple observations of the scene are available during model training. From the interaction perspective, multi-view observations also provide useful additional input to the system. Given a series of \( n \) observations \( I_1, I_2, \ldots, I_n \) of the scene, the 3D reconstruction can be formulated as \( f^V : \{I_i\}_{i=1}^{n} \rightarrow V \). Similarly, the projective operation from \( i \)-th viewpoint is \( f^D : V \times P_i \rightarrow D_i \), where \( D_i \) and \( P_i \) are the depth and camera transformation matrix from corresponding viewpoint, respectively. We define the multi-view loss \( \mathcal{L}_{\text{multi}} \) in Eq. 3 with an emphasis on the shape prediction consistency across viewpoints.

\[
\mathcal{L}_{\theta}^{\text{multi}} = \lambda_D \sum_{i=1}^{n} \mathcal{L}_{\theta}^{\text{depth}}(\hat{D}_i, D_i) + \lambda_M \sum_{i=1}^{n} \mathcal{L}_{\theta}^{\text{mask}}(\hat{M}_i, M_i)
\]

**3.2 Learning predictive grasping interaction with geometry-aware representation.**

In general, motion planning and control for grasping is very challenging due to many factors involved. In this work, we focus on modeling the pre-grasp status as fine-grained motion planning becomes increasingly important when the gripper reaches close to target object. In our formulation, we assume grasping outcome is binary: either success or failure. The interaction is classified as success only if the action results in a valid grasp. Based on our formulation, the grasping success probability can be directly inferred from the visual observation \( I \) of current state and proposed action \( a = [p, o] \).

Inspired by previous work [Oh et al. 2015], [Finn et al. 2016], [Dosovitskiy and Koltun 2016], [Yang et al. 2015], [Pinto et al. 2016], where outcomes are high-order mappings from observations and actions, a straight-forward approach is to fit a functional mapping \( f_{\text{vanilla}} : I \times a \rightarrow l \). We refer to this model as a vanilla grasping interaction prediction model (see Figure 3 (a)).

Building upon the vanilla prediction model, we propose a novel geometry-aware prediction model. That is, the agent learns to predict the grasping interaction by taking the geometry-aware representation as an additional input. The benefits of such geometry-aware representation are two-folds:

- The geometry-aware representation provides a global shape prior for interaction predictions.
- The geometry-aware representation provides 3D information about the local surface centered around the interaction event.
Local shape inference via projection. In our implementation, we reuse the learning-free projection operator to obtain the local depth given the gripper position and orientation.

Finally, given a current observation $I$, proposed action $a$, and inferred 3D shape representation $V$, we fit a functional mapping $f_{\text{geometry-aware}} : I \times a \times V \rightarrow l$, where $l$ is the binary label of whether it is a valid grasp.

3.3 Deep geometry-aware encoder-decoder network

To implement the two components proposed in the previous sections, we introduce a deep geometry-aware encoder-decoder network (see Figure 2). Our model is composed of a shape prediction network and a grasping outcome prediction network. The shape prediction network has a 2D convolutional shape encoder and a 3D deconvolutional shape decoder followed by a global projection layer. Our shape encoder network takes RGBD images of resolution 128 $\times$ 128 and corresponding 4-by-4 camera view matrices as input; the network and outputs identity units as an intermediate representation. Our shape decoder is a 3D deconvolutional neural network that outputs voxels at a resolution of 32 $\times$ 32 $\times$ 32. We implemented the projection layer (with camera view and projection matrix) that transforms the voxels back into foreground object silhouettes and depth maps at an input resolution (128 $\times$ 128). Here, the purpose of generative pre-training is to learn viewpoint invariant units (e.g., object identity units) through object segmentation and depth prediction. The outcome prediction network has a 2D convolutional state encoder and a fully connected outcome predictor with an additional local shape projection layer. Our state encoder takes RGBD images (pre-grasp scene) of resolution 128 $\times$ 128 and corresponding actions (position and orientation of the gripper end-effector) and outputs state unit as intermediate representation. Our outcome predictor takes both current state (e.g., pre-grasp scene and action) and shape features (e.g., global shape from identity units and the local surface from the local shape projection layer) into consideration. Note that the local dense sampling transforms the surface around the gripper fingers into a foreground silhouette and a depth map at resolution 48 $\times$ 48. For the purpose of better shape representation during training, we feed observations taken from multiple viewpoints to the neural networks. During evaluation, we only provide single-view observation for the model as input.

4 Experiments

4.1 Dataset collection

VR-Grasping-101. We collected grasping demonstrations on seven categories of objects, which include a total of 101 everyday objects. To collect grasping demonstrations, we setup the HTC Vive system in Virtual Reality (VR) and assign target objects randomly to five right-handed users (three males and two females). In total, 1597 human grasps are demonstrated, with an average of 15 grasps per object (with lowest and highest number of grasps at 7 and 39 for a plate and a wine glass, respectively). We randomly split 101 objects into three sets (e.g., training, validation and testing) and make sure each set covers the seven categories (70% for training, 10% for validation and 20% for testing).
Input RGBD

Predicted 3D shape

(a) 3D Shape prediction from single-view RGBD image (seen objects)

(b) 3D Shape prediction from single-view RGBD image (novel objects)

Figure 3: Visualization: 3D shape prediction from single-view RGBD. (a) The performance on training (seen) objects. (b) The performance on testing (novel) objects.

Grasping data perturbation. In order to collect sufficient grasping demonstrations for model training and evaluation, we generate more grasps by perturbing the human demonstrations using Bullet Physics engine. In total, we collected 150K grasping demonstrations covering 101 objects. For each demonstration, we take a snapshot of the pre-grasp scene (e.g., before closing the two gripper fingers). To minimize the bias introduced from the data generation pipeline, we randomly posit the camera at a distance ranging between 35 centimetres and 45 centimetres. We draw a camera target position from a normal distribution with its mean as the object center and a desired variance (in our experiment, we use 3 centimetres as standard deviation). Furthermore, we rotate the camera position from 8 different azimuth angles (with steps of 45 degrees) and adjust the elevation from 4 different angles (e.g., 15, 30, 45, and 60 degrees). Here, we include only two elevation angles (e.g., 15 and 45 degrees) in the training set while leaving the other two angles for evaluation. Finally, we also save a state of the scene without a gripper, which is used for shape pre-training; this will be referred to as the static scene for the rest of the paper.

4.2 Implementation details

Baseline model. We adopt the vanilla prediction model as our grasping baseline. We trained the model using the ADAM optimizer with a learning rate of $10^{-5}$ for 200K iterations and a mini-batch of size of 4. As an ablation study, we added view and static scene as an additional input channel on top of the baseline model but didn’t observe significant improvements.

Geometry-aware model. As mentioned previously, we adopted the two-stage training procedure. First, we pre-trained the shape prediction model (shape encoder and shape decoder) using the ADAM optimizer with a learning rate of $10^{-5}$ for 400K iterations and a mini-batch of size of 4. In each batch, we sample 4 random viewpoints as our multi-view training. We observed that this setting led to a more stable shape prediction performance compared to single-view training. In addition, we used $L_1$ loss for foreground depth prediction and $L_2$ loss for silhouette prediction with coefficients $\lambda_D = 0.5$ and $\lambda_M = 10.0$. In the second stage, we fine-tuned the static encoder and outcome predictor using ADAM optimizer with a learning rate of $3 \times 10^{-6}$ for 200K iterations and a mini-batch of size of
4. We used cross-entropy as our objective function since the grasping prediction is formulated as a binary classification task.

In our experiments, all the models are trained using 20 GPU workers and 32 parameter servers with asynchronized updates. Both baseline and our geometry-aware model adopt convolutional encoder-decoder architecture with residual connections. The bottleneck layer (e.g., the identity unit in the geometry-aware model) is a 768 dimensional vector.

4.3 Visualization: 3D shape prediction

To evaluate the quality of generative shape prediction model, we performed inference using the shape encoder and decoder network. In our evaluations, we used single-view RGBD image and corresponding camera view matrix as input to the network. As shown in Figure 3(a), our shape prediction model is able to generate fine-grained 3D voxels from single-view input without explicitly providing 3D voxels as supervision during training. As shown in Figure 3(b), our model demonstrates reasonable generalization ability even when applying to novel object instances.

4.4 Model evaluation: grasping outcome prediction

With a learned geometry-aware representation, our model achieves better classification performance in predicting the grasping outcome. We compared the classification accuracy of the baseline model and our geometry-aware model by conducting extensive evaluations on novel objects (from the testing set) from multiple observation viewpoints. For each human demonstration, we prepared 100 random grasps through perturbation (among which 50% of them are success grasps) and computed the average accuracy on 100 grasps (i.e., random guess achieves 50% accuracy). To investigate the model performance due to viewpoint changes, we repeat the evaluation experiment for four different elevation angles (e.g., 15, 30, 45, and 60 degrees). The results are summarized in Table 1 and Table 2.

Overall, the geometry-aware model performs consistently better than baseline model in outcome classification. As we can see, “teapot” and “plate” are comparatively more challenging categories for outcome prediction, since “teapot” has irregular shape parts (e.g., tip and handle) and “plate” has a fairly flat shape. When it comes to novel elevation angles (e.g., compare Table 1 and Table 2), our geometry-aware model is less affected, especially for categories like “teapot” and “plate” where viewpoint invariant shape understanding is crucial.

Analysis: local shape inference via projection. One advantage of our generative shape prediction component is that we can obtain additional local shape information via projection (see the red-dashed box in Figure 2(c)). At testing time, our shape prediction component first generates the 3D voxels given 2D observation (at a distance). With the 3D voxels as part of the intermediate representation, we can further acquire the local shape by running a projection from the gripper’s perspective (i.e., simply treat the gripper as another virtual camera). To further understand the advantages of our generative shape prediction component, we visualized the intermediate local shape representation projected from predicted 3D voxels. As shown in Figure 3, our generative shape prediction component provides reasonably accurate local shape estimation that is useful for grasping outcome prediction.

Application: analysis-by-synthesis grasping optimization. With improved prediction over the grasping outcome, a natural question is whether this improvement can be used to guide better grasping
| Method / Category | bottle | bowl | cup | plate | mug | sugarbowl | teapot | all |
|------------------|--------|------|-----|-------|-----|----------|-------|-----|
| baseline (15)    | 72.81  | 73.36 | 73.26 | 66.92 | 72.23 | 70.45 | 66.13 | 71.42 |
| geo-aware (15)   | 78.83  | 79.32 | 77.60 | 68.88 | 78.25 | 76.09 | 73.69 | 76.55 |
| baseline (45)    | 71.02  | 74.16 | 73.50 | 63.31 | 74.23 | 72.70 | 64.19 | 71.32 |
| geo-aware (45)   | 78.77  | 80.63 | 78.06 | 70.13 | 79.29 | 77.52 | 72.88 | 77.25 |

Table 1: Outcome prediction accuracy from seen elevation angles.

| Method / Category | bottle | bowl | cup | plate | mug | sugarbowl | teapot | all |
|------------------|--------|------|-----|-------|-----|----------|-------|-----|
| baseline (30)    | 71.15  | 72.98 | 71.65 | 61.90 | 71.01 | 70.06 | 61.88 | 69.50 |
| geo-aware (30)   | 79.17  | 77.71 | 77.23 | 67.00 | 75.95 | 75.06 | 70.66 | 75.27 |
| baseline (60)    | 68.45  | 73.05 | 72.50 | 61.27 | 74.40 | 71.30 | 63.25 | 70.18 |
| geo-aware (60)   | 77.40  | 78.52 | 76.24 | 68.13 | 79.39 | 76.15 | 70.34 | 75.76 |

Table 2: Outcome prediction accuracy from novel elevation angles.

| Method / Category | bottle | bowl | cup | plate | mug | sugarbowl | teapot | all |
|------------------|--------|------|-----|-------|-----|----------|-------|-----|
| baseline + CEM   | 48.60  | 64.28 | 55.44 | 45.99 | 61.00 | 63.97 | 63.08 | 55.85 |
| geo-aware + CEM  | 56.73  | 68.84 | 60.31 | 50.09 | 67.21 | 59.87 | 69.22 | 61.46 |
| rel. improvement (%) | 16.72 | 7.09 | 8.77 | 8.92 | 10.18 | 9.73 | 10.03 |

Table 3: Grasping optimization on novel objects: success rate by optimizing for up to 20 steps.

planning. Given a seed grasping proposal, we conducted grasping optimization by sequentially proposing grasping locations until a grasp success. For grasping optimization, we performed a simplified version of cross-entropy method (CEM) \cite{rubinstein2004cross,levine2016end}. We initialized with a failure grasp in order to force the model to find better grasping location (e.g., position and orientation). At each iteration, we sample 10 random directions and selected the top one based on the score returned by the neural network (output of outcome predictor). We repeat the iterations until success with an upper bound of 20 steps. We ran the same evaluation for both the baseline model and our geometry-aware model. To account for the variations in observation viewpoints and initial seeds, we repeat the evaluation for eight times per testing demonstration in our dataset and reported the average success rate after 20 iterations (marked as failure only if there is no success in 20 steps). As shown in Table 3, CEM guided our geometry-aware model performance consistently better than baseline model. We believe that the improvement results from the explicit use of modeling the object shape in our geometry-aware model. Our model achieved the most significant improvement in the “bottle” category, since a bottle shape is relatively easy to reconstruct. Our improvement in the “bowl” category is less significant, partly due to the failure in predicting its concave shape for testing object instances. Figure 5 shows example grasping optimization trials with different types of objects. The baseline model was stuck at the local region while our geometry-aware model was able to transit from one side of the object to the other optimal position and orientation.

5 Conclusions

In this work, we studied the grasping interaction from a geometry-aware learning agent’s perspective. We proposed an encoder-decoder network that performs shape prediction as well as grasping outcome prediction with a learning-free OpenGL projection layer. Compared to the baseline, experimental results demonstrated improved performance in outcome prediction thanks to generative shape training. Guided by the improved outcome predictor, we achieved better planning via analysis-by-synthesis grasping optimization. We have demonstrated the benefits of having geometry-aware representation in perception and motion planning. In the future, we will explore possibilities that performs robotic control with our geometry-aware representation.

Acknowledgments

We thank Kurt Konolige, Erwin coumans, Vincent Vanhoucke, Ethan Holly, Marek Fiser, Eric Jang, Jie Tan, Lajanugan Logeswaran, Ruben Villegas, the Google Brain Team and X for the help with the project.
Figure 5: Visualization: grasping optimization with CEM based on the grasping prediction output. In each row, we selected three representative steps in grasping optimization (in sequential order from left to right). Red box represents a failure grasp while green box represents a successful grasp.

References

H. Dang and P. K. Allen. Semantic grasping: planning task-specific stable robotic grasps. *Autonomous Robots*, 37(3):301–316, 2014.

A. Dosovitskiy and V. Koltun. Learning to act by predicting the future. *arxiv preprint: 1611.01779*, 2016.

C. Finn, I. Goodfellow, and S. Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in Neural Information Processing Systems*, pages 64–72, 2016.

C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen. The columbia grasp database. In *Robotics and Automation, 2009. ICRA’09. IEEE International Conference on*, pages 1710–1716. IEEE, 2009.

M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015.

E. Johns, S. Leutenegger, and A. J. Davison. Deep learning a grasp function for grasping under gripper pose uncertainty. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 4461–4468. IEEE, 2016.

D. Katz, A. Venkatraman, M. Kazemi, J. A. Bagnell, and A. Stentz. Perceiving, learning, and exploiting object affordances for autonomous pile manipulation. *Autonomous Robots*, 37(4):369–382, 2014.

I. Lenz, H. Lee, and A. Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.

B. Leon, S. Ulbrich, R. Diankov, G. Puche, M. Przybylski, A. Morales, T. Asfour, S. Moisio, J. Bohg, J. Kuffner, et al. Opengrasp: A toolkit for robot grasping simulation. *SIMPAR*, 2010.

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*, page 0278364917710318, 2016.

M. Li, K. Hang, D. Kragic, and A. Billard. Dexterous grasping under shape uncertainty. *Robotics and Autonomous Systems*, 75:352–364, 2016.
J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg. Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1957–1964. IEEE, 2016.

J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arxiv preprint: 1703.09312*, 2017.

L. Montesano and M. Lopes. Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions. *Robotics and Autonomous Systems*, 60(3):452–462, 2012.

E. Nikandrova and V. Kyrki. Category-based task specific grasping. *Robotics and Autonomous Systems*, 70:25–35, 2015.

J. Oh, X. Guo, H. Lee, R. L. Lewis, and S. Singh. Action-conditional video prediction using deep networks in atari games. In *Advances in Neural Information Processing Systems*, pages 2863–2871, 2015.

L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 3406–3413. IEEE, 2016.

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*, pages 3–18. Springer, 2016.

D. J. Rezende, S. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess. Unsupervised learning of 3d structure from images. In *Advances In Neural Information Processing Systems*, pages 4997–5005, 2016.

R. Rubinstein and D. Kroese. *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*. Springer-Verlag, 2004.

A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic grasping of novel objects using vision. *The International Journal of Robotics Research*, 27(2):157–173, 2008.

N. Vahrenkamp, L. Westkamp, N. Yamanobe, E. E. Aksoy, and T. Asfour. Part-based grasp planning for familiar objects. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pages 919–925. IEEE, 2016.

J. Varley, C. DeChant, A. Richardson, A. Nair, J. Ruales, and P. Allen. Shape completion enabled robotic grasping. *arxiv preprint: 1609.08546*, 2016.

X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In *Advances in Neural Information Processing Systems*, pages 1696–1704, 2016.

J. Yang, S. E. Reed, M.-H. Yang, and H. Lee. Weakly-supervised disentangling with recurrent transformations for 3d view synthesis. In *Advances in Neural Information Processing Systems*, pages 1099–1107, 2015.