On-Policy Dataset Synthesis for Learning Deep Robot Grasping Policies based on Fully-Convolutional Grasp Quality Neural Networks

Vishal Satish[1], Jeffrey Mahler[1], Ken Goldberg[1,2]

***Abstract*— Rapid and reliable robot grasping for a diverse set of objects has applications from home de-cluttering to warehouse automation. A promising approach is to learn deep policies from synthetic training datasets of point clouds, grasps, and rewards sampled using analytic models along with stochastic noise models for domain randomization. In this paper, we explore the effect of the distribution of synthetic training examples on the rate and reliability of the learned robot policy. To increase rate and reliability, we propose a synthetic data sampling distribution that combines grasps sampled from the policy action set with guiding samples from a robust grasping supervisor that has full state knowledge. We use this to train a robot policy based on a novel fully-convolutional network architecture that evaluates millions of 4-DOF grasp candidates (3D position and planar orientation) in parallel to maximize grasp quality. In physical robot experiments, we find that the Fully-Convolutional Grasp Quality CNN (FC-GQ-CNN) policy is able to achieve up to 296 mean picks per hour (MPPH) compared to 250 MPPH for policies based on iterative grasp sampling and evaluation. We perform sensitivity experiments to study the relationship between the granularity of the policy action space and performance. Code, datasets, videos, and supplementary material can be found at http://berkeleyautomation.github.io/fcgqcnn.**

I. Introduction

Robots in homes and warehouses must be able to rapidly and reliably plan grasps for a wide variety of objects under inherent uncertainty in sensing, physics, and control. One approach is to compute grasps for a set of known 3D objects using analytic models and to plan grasps online by matching sensor data to known objects. However, analytic methods assume a perception system that can recognize object instances, making them difficult to scale to many novel objects.

An alternative approach is to use machine learning to train a robot policy to predict the probability of success for candidates grasps based on sensor data such as images or point clouds. Recent results suggest that learned robot policies can rapidly grasp a wide variety of novel objects on a physical robot. Learning-based grasp planning approaches require a data collection policy for collecting training examples. Empirical methods collect training data based on human labeling [15], [24], [39], dataset aggregation from self-supervision [17], [27], or reinforcement learning [11]. However, these dataset collection approaches may be time-consuming and prone to mislabeled examples.

An alternative is to use a hybrid method that rapidly generates massive synthetic training datasets using analytic metrics and domain randomization for robust transfer from simulation to reality [10], [21], [34], [35]. The synthetic data distribution can be modified to mirror the object states, camera, or quality metric for a particular task. However, many systems sample training grasps from a fixed distribution that is difference from the set of actions that the policy must evaluate at runtime. For example, several approaches sample training grasp actions that are constrained to known 3D object surfaces while the learned policy samples grasps from observations [21], [22]. This may lead to reduced performance due a phenomenon known as covariate shift [14], [32]. This raises the question: Is it possible to develop faster and more reliable grasping policies by modifying the data collection policy used for sampling synthetic training examples?

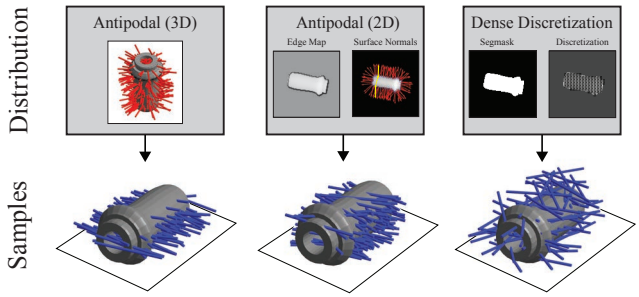

Fig. 1: Grasp action spaces used to guide synthetic data collection for training robust robot grasping policies. The Dexterity Network 2.0 [22] uses 3D antipodal parallel-jaw grasp actions sampled from object models for training (left), and ranks 2D antipodal grasps sampled from depth images during online policy evaluation (middle), which can diminish performance due to covariate shift. We propose to instead collect training data from the policy's action space by sampling grasps based on synthetic observations. We train a novel rapid robot policy based on Fully Convolutional Grasp Quality Neural Networks (FC-GQ-CNNs) using a dense discretization of a 4-DOF grasp action space (3D position and planar orientation) within an object segmask (right).

In this paper, we propose an on-policy dataset distribution that samples grasps from synthetic observations to approximate the distribution hat the policy will evaluate with a learned quality function at runtime. To guide data collection towards successful grasps, the distribution samples a mixture of grasps from the action space of the robot policy and from an algorithmic robust grasping supervisor that leverages the known geometry and pose of 3D objects to index pre-computed grasps. We use this to train an efficient single-shot grasping policy based on Fully-Convolutional Networks, an architecture introduced in computer vision for image segmentation [20] that has shown promising results for learning grasping policies from human labeled datasets [24], [39]. We develop a novel variant of this architecture that

[1] Dept. of Electrical Engineering and Computer Science;
[2] Dept. of Industrial Operations and Engineering Research;
The AUTOLAB at UC Berkeley (automation.berkeley.edu).
{vsatish, jmahler, goldberg}@berkeley.edu

evaluates grasps with four degrees of freedom (3D position and planar orientation) by parallelizing standard Grasp Quality Convolutional Neural Networks (GQ-CNNs). The architecture can rapidly produce dense and reliable grasp predictions by evaluating millions of grasps in parallel.
This paper contributes:

1) A novel dataset collection policy for sampling massive synthetic training datasets that reflects the distribution of actions that the learned policy evaluates at runtime.
2) A single-shot 4-DOF (3D position and gripper orientation) Fully-Convolutional Grasp Quality CNN (FC-GQ-CNN) architecture.
3) Empirical data from a physical robot comparing the performance of this approach to current state-of-the-art hybrid approaches, along with experiments in simulation exploring its sensitivity to the choice of action space granularity.

We find that the FC-GQ-CNN policy trained on the on-policy distribution is able to efficiently evaluate millions of grasps while maintaining an evaluation time per grasp on the order of $10^{-}3$ms. In bin picking with 25 novel objects on an ABB YuMi, the policy improves rate and reliability to 296 mean picks per hour (MPPH) compared to 250 MPPH for policies based on iterative grasp sampling and ranking.

II. Related Work

A. Learning for Grasp Planning

The goal of grasp planning is to find a gripper configuration that maximizes a quality metric. Initial approaches to the problem utilized analytic approaches (see [28] for a survey). However, the inability of these approaches to generalize to novel objects has lead to the use of empirical and hybrid approaches, the latter of which utilizes massive synthetic training datasets generated with analytic models.

Combined with advances in deep learning, these approaches utilize policies that query a neural network to locate the highest quality grasp. These fall into two categories: *Discriminative* approaches utilize a neural network to rank grasps based on a quality metric and optimization techniques to probe the state space for grasp candidates as does Dex-net 2.0 [22]. *Generative* approaches instead directly generate a grasp set given sensor data, and may use heuristics to select the optimal grasp from this set. One popular approach is to regress to grasp coordinates in image space [15], [29]. Another notable method is Grasp Pose Detection (GPD) [34], which detects 6-DOF grasp poses based on a CNN that takes as input a 15-channel image computed from projections of a 3D voxel grid. 3D GANs have also been used to generate gripper configurations from a 3D voxel grid [36].

These deep approaches have been trained on massive datasets of human-labeled [12], [15], [29], self-supervised [7], [18], [25], [27] or synthetic [3], [5], [10], [22], [37] grasps, images, and quality labels. A popular human-labeled dataset is the Cornell Grasping dataset developed by Lenz et al. [15], which consists of 1k RGB-D images labeled with grasps parametrized by oriented bounding boxes. The Cornell Grapsing dataset has been extensively used to train CNN-based models for singulated objects [13], [24], [29]. Self-supervised datasets have been collected from grasp attempts on a physical robot. Pinto and Gupta [27] collected over 40k grasp attempts on a Baxter to train a CNN, whereas Levine et al. [18] expanded this approach even further by collecting over 800k datapoints using numerous robot arms. Synthetic datasets have been used to train state-of-the-art hybrid approaches. Notably, Mahler et al. [22] trained a Grasp Quality CNN (GQ-CNN) on a synthetic dataset of millions of grasps generated automatically through physics-based and geometric models. We explore the effect of the distribution of synthetic training examples on the rate and reliability of learned policies.

B. Fully-Convolutional CNNs

Fully-convolutional CNNs were developed in the field of computer vision for tasks requiring dense pixel-wise discrimination such as image segmentation [20]. They have also been used for object detection [4] and visual tracking [38] as opposed to sparser approaches such as YOLO [30], SSD [19], and Mask-RCNN [8]. In these detection tasks, fully-convolutional networks allow for dense approaches that evaluate thousands of regions of the image in parallel while maintaining reasonable computation time by utilizing highly optimized matrix operations at the GPU level.

Several successful empirical approaches have taken advantage of Fully Convolutional Networks (FCNs). Zeng et al. [39] trained FCNs on hundreds of human-labeled images to predict the probability of success for four grasp primitive actions, rapidly generating the probability of success for all grasp candidates in parallel. Morrison et al. [24] used FCNs to increase grasp planning frequency to 50Hz, using a discriminative head to predict the probability of grasp success and separate network heads to generate the grasp angle and gripper width. In comparison, we study the distribution of data used to train a novel 4-DOF architecture and show that an on-policy distribution can lead to increased reliability.

C. Training Distribution

Learning-based approaches to grasp planning require a large labeled dataset of training data, and the distribution of the training data may affect the performance of the learned policy. Prior approaches have used a distribution based on human labels [15], [39], random exploration [18], [27], or the set of antipodal grasps on 3D mesh surfaces [22]. The fields of IL [32] and RL [33] have considered how to optimize the distribution of training data to improve learning efficiency and to reduce covariate shift. In IL, approaches are either on-policy, using supervisor labels on actions taken by the current learned policy [32], or off-policy [14], using actions taken by the supervisor. In RL, a common approach is to sample actions using epsilon greedy, which mixes random actions from the action set with actions preferred by the current trained policy [33]. Supervised actor-critic [31] approaches to RL, such as Actor-Mimic [26], use a supervisor policy to guide the distribution of actions taken to train a policy. Several methods incorporate similarity to supervisor

actions into the RL reward function, such as Deep Learning from Demonstrations [9] and Guided Policy Search [16]. In comparison, we consider data collection for supervised learning and use a training dataset distribution based on a robust grasping supervisor that uses a database of 3D object models to index grasps.

III. Problem Statement

We consider the problem of learning a robot grasping policy for a wide variety of novel objects that maximizes Mean Picks Per Hour (MPPH), the number of objects that are successfully grasped per hour. This depends on rate, defined as how fast the policy can plan and execute grasps, and reliability, defined as the percentage of successful grasps.

The goal is for a robot to iteratively grasp and transport a single object from a bin to a receptacle based on point clouds from a depth camera. The *state* $\mathbf{x}$ includes the geometry, pose, and material properties of each object. The robot acquires a *point cloud observation* $\mathbf{y}$ represented as a depth image. Then, the robot uses a *grasp policy* π_θ, defined by neural network weights θ, that takes as input an observation $\mathbf{y}$ and returns a grasp $\mathbf{u} = \pi_\theta(\mathbf{y})$. A grasp is specified as the 3D position and planar orientation of a parallel-jaw gripper. Upon executing the grasp, the robot receives a binary *reward* $R(\mathbf{x}, \mathbf{u}) \in \{0, 1\}$ based on whether or not an object is successfully grasped and transported to the receptacle. The grasp attempt has a duration consisting of the combined sensing time t_s, grasp computation time t_c, and grasp execution time t_e, in fraction of hours, which we assume to be constant.

The learning objective is to maximize MPPH:

$$\max_{\theta \in \Theta} \mathbb{E}\left[\frac{R(\mathbf{x}, \pi_\theta(\mathbf{y}))}{t_s + t_c + t_e}\right] = \max_{\theta \in \Theta} \mathbb{E}\left[R(\mathbf{x}, \pi_\theta(\mathbf{y}))\right] \quad \text{(III.1)}$$

The expectation is taken with respect to the grasping environment, a distribution over possible rewards, observations, and actions based on the policy:

$$p(R, \mathbf{x}, \mathbf{y}, \mathbf{u} \mid \theta) = \underbrace{\pi(\mathbf{u} \mid \mathbf{y}, \theta)}_{\text{policy}} \underbrace{p(R \mid \mathbf{x}, \mathbf{u})}_{\text{reward}} \underbrace{p(\mathbf{y} \mid \mathbf{x})}_{\text{observation}} \underbrace{p(\mathbf{x})}_{\text{state}} \quad \text{(III.2)}$$

IV. Learning Objective

We follow the approach of using supervised learning to train a policy based on a quality function Q_θ that predicts the probability of success for a given grasp using a deep neural network with parameters θ [2], [15], [10], [22]. The policy maximizes this function over all grasps in the action space $\mathcal{U}(\mathbf{y})$ to select a grasp:

$$\pi_\theta(\mathbf{x}) = \underset{\mathbf{u} \in \mathcal{U}(\mathbf{y})}{\operatorname{argmax}}\ Q_\theta(\mathbf{y}, \mathbf{u}) \quad \text{(IV.1)}$$

To train the network, we minimize the cross-entropy loss between the predicted grasp quality and reward:

$$\min_{\theta \in \Theta} \mathbb{E}\left[\mathcal{L}(R, Q_\theta(\mathbf{y}, \mathbf{u}))\right]$$

Here the expectation is taken with respect to a dataset distribution defined by a dataset collection policy τ that may be independent of the policy parameters:

$$p(R, \mathbf{x}, \mathbf{y}, \mathbf{u} \mid \theta) = \underbrace{\tau(\mathbf{u} \mid \mathbf{x}, \mathbf{y})}_{\text{policy dist.}} \underbrace{p(R \mid \mathbf{x}, \mathbf{u})}_{\text{reward dist.}} \underbrace{p(\mathbf{y} \mid \mathbf{x})}_{\text{observation dist.}} \underbrace{p(\mathbf{x})}_{\text{state dist.}} \quad \text{(IV.2)}$$

The distribution τ is designed to reflect a diverse set of actions that may be evaluated by the learned quality function at runtime. Note that this is distinct from the distribution of actions planned by the policy, as the quality function must evaluate a diverse set of grasp candidates and discard poor actions. In prior work, τ is sampled off-policy by collecting data from a human supervisor [24], [39], the current best policy [17], [27], or 3D antipodal grasps [22].

V. On-Policy Dataset Synthesis

The hybrid approach to learning robust grasping policies samples training datasets from a synthetic dataset distribution that is the product of a simulated training environment $\xi(R, \mathbf{x}, \mathbf{y} \mid \mathbf{u})$ and a data collection policy $\tau(\mathbf{u} \mid \mathbf{x}, \mathbf{y})$. The training environment ξ models the distribution of rewards, states, and point clouds using analytic models based on physics and geometry [22] with domain randomization for robust sim-to-real transfer [35]. The data collection policy τ attempts to sample a diverse set of actions that the learned quality function may need to evaluate at runtime. Nonetheless, several hybrid methods such as the Dexterity Network (Dex-Net) 2.0 [22], [21] use different distributions of grasp actions for training and policy deployment, which may reduce performance due to covariate shift [32], [14].

Drawing inspiration from approaches in imitation learning [14], [32] and reinforcement learning [16], [31], we propose an on-policy dataset distribution. The distribution uses a data collection policy that samples grasps from the action space $\mathcal{U}(\mathbf{y})$ that the policy evaluates with the learned quality function at runtime (see Equation IV.1). To increase the percent of successful grasp actions, the distribution uses guiding samples from a robust grasping supervisor that plans robust grasps analytically using full knowledge of 3D object geometry and pose.

Formally, the data collection policy is:

$$\tau(\mathbf{u} \mid \mathbf{x}, \mathbf{y}) = (1 - \epsilon) Unif(\mathcal{U}(\mathbf{y})) + \epsilon \Omega(x)$$

where $\mathcal{U}(\mathbf{y})$ is the grasp action space sampled from a point cloud observation $\mathbf{y}$ of the state of 3D objects $\mathbf{x}$, and $\Omega(\mathbf{x})$ is the Dex-Net 1.0 robust grasping supervisor [23]. The parameter ϵ controls the percentage of actions to sample from the supervisor. A larger value of ϵ may increase covariate shift as more actions are sampled from the supervisor, while smaller values of ϵ may skew the distribution toward many negative examples and require larger training datasets.

To increase the rate of grasp computation, we use this data collection policy to train a Fully Convolutional Grasp Quality CNN (FC-GQ-CNN) on a 4-DOF action space (3D position and planar orientation). Fig. 1 illustrates the dense discretization of the 4-DOF grasp action space that is evaluated at runtime and used to sample training data.

VI. Fully-Convolutional 4-DOF Architecture

To achieve a more efficient policy, we seek to improve MPPH by reducing t_c, the computation time for each grasp. Current state-of-the-art approaches implement (IV.1) as an optimization loop that repeatedly queries a deep neural network for the probability of success on single grasps, $Q_\theta(\mathbf{y}, \mathbf{u})$, using for example the *cross-entropy* method (CEM) [18], [22] to hone in on the best grasp after many iterations.

One drawback of this optimization loop is that it must be implemented in a serial fashion and requires computational overhead every time the network must be queried with a new batch of predictions. The iterative optimization also often involves many parameters which may be difficult to tune. One approach to alleviate these issues has been proposed by Zeng [39] and Morrison [24]: a deep Fully-Convolutional Network (FCN) architecture that can produce an extremely dense yet efficient set of predictions over the entire state space. This reduces the search for the highest quality grasp to a straightforward argmax of a the network output.

We also hypothesize that a dense set of predictions over the action space might be able to find a grasp closer to the global optimum than a sparse approach such as the cross-entropy method, which might never explore certain optimal grasps due to choice of parameters and the limited number of grasps it can explore within a reasonable computational budget.

The denser we can make the FCN output, the more efficiently we can cover the state space, and the more work we can offload from the policy to the neural network inference, which can be highly optimized on current state-of-the-art GPUs. With this goal in mind, we propose a novel 4-DOF Fully-Convolutional Grasp Quality CNN (FC-GQ-CNN) architecture by parameterizing the network using 3D gripper position and planar orientation.

A. Architecture

Similar to the approach used by Morrison et al. [24], we train a standard CNN but during policy inference convert all fully connected layers to convolutional layers to deploy the standard CNN as a Fully-Convolutional Network (FCN). This foregoes the need to train a set of deconvolution layers and the need for densely-labeled ground-truth images during training. We can instead train the CNN on much smaller cropped images with a single grasp each, which are easier to generate with analytic approaches [22].

1) Training CNN Architecture: We first design a 4-DOF CNN architecture that is invariant in the xy image space so that it can be directly transformed to a FCN during policy inference. We take inspiration from the GQ-CNN [22], extending it to 4-DOF by incorporating the grasp angle θ. This angular GQ-CNN architecture takes as input a cropped thumbnail depth image centered on the grasp center pixel, $\mathbf{y}$, along with the corresponding grasp depth relative to the camera, $\mathbf{z}$. It computes a set of k success probabilities, each corresponding to a planar gripper angle.

Unlike in the GQ-CNN, we cannot incorporate depth using a separate network stream. The separate stream presents a computational bottleneck down the road in the FCN architecture because its output must be expensively tiled across the output of the final convolution layer, which can be fairly large for larger input sizes. We instead incorporate the depth $\mathbf{z}$ into the network by subtracting it from the depth image $\mathbf{y}$, thus transforming the depth image into the grasp frame of reference. Following standard preprocessing conventions, we normalize the transformed depth images by subtracting the mean and dividing by the standard deviation of the training data.

2) Fully-Convolutional GQ-CNN Architecture: By converting each of the fully connected layers of the angular GQ-CNN into a convolution layer, we define the FC-GQ-CNN architecture. This is a valid transformation because of the one-to-one mapping between convolution and fully connected layer weights. Illustrated in Fig. 2 (Top) and detailed in the caption, it takes as input a full-size depth image $\mathbf{y}$ and corresponding gripper depth relative to the camera $\mathbf{z}$, and outputs a dense grid of outputs over the image space, with k success probabilities predicted at each point in this grid in the angular space. The entire angular GQ-CNN network behaves as one huge convolution filter, and each inference pass through the FC-GQ-CNN is the equivalent of a convolution operation with this filter. The effective stride over the input is determined by the amount of pooling present in the convolution layers of the angular GQ-CNN architecture, specifically each pooling by a factor of p will increase the stride by a corresponding factor.

VII. Policy Learning

A. Policy

The benefit of using a deep learning model that can output dense predictions is that computation can be offloaded to highly optimized GPU operations and the policy itself, which is often implemented on the CPU, is simple and straightforward.

In particular for the FC-GQ-CNN policy(see Fig. 2 Bottom), given a depth image $\mathbf{y}$ and FC-GQ-CNN network Q_θ, it consists of only two steps:

1) Bin the range $min(\mathbf{y}) - max(\mathbf{y})$ into n depth bins, and evaluate $Q_\theta(\mathbf{y}, \mathbf{z}_i)$ for each $i \in n$ where $\mathbf{z}_i$ is the center of bin i.
2) Take the *argmax* of all the predictions $Q_\theta(\mathbf{y}, \mathbf{z}_i)$ and return 3D position and orientation.

B. FC-GQ-CNN Training

We train the angular GQ-CNN on 96x96 depth image thumbnails of individual grasps. We optimize the parameters of the angular GQ-CNN network using backpropagation with stochastic gradient descent and momentum. The network output consists of all k angular predictions, however each training sample corresponds to only one specific angle. Given a depth image with grasp angle θ, we first map θ to the corresponding angular bin, then we backpropagate only through the network output corresponding to that particular

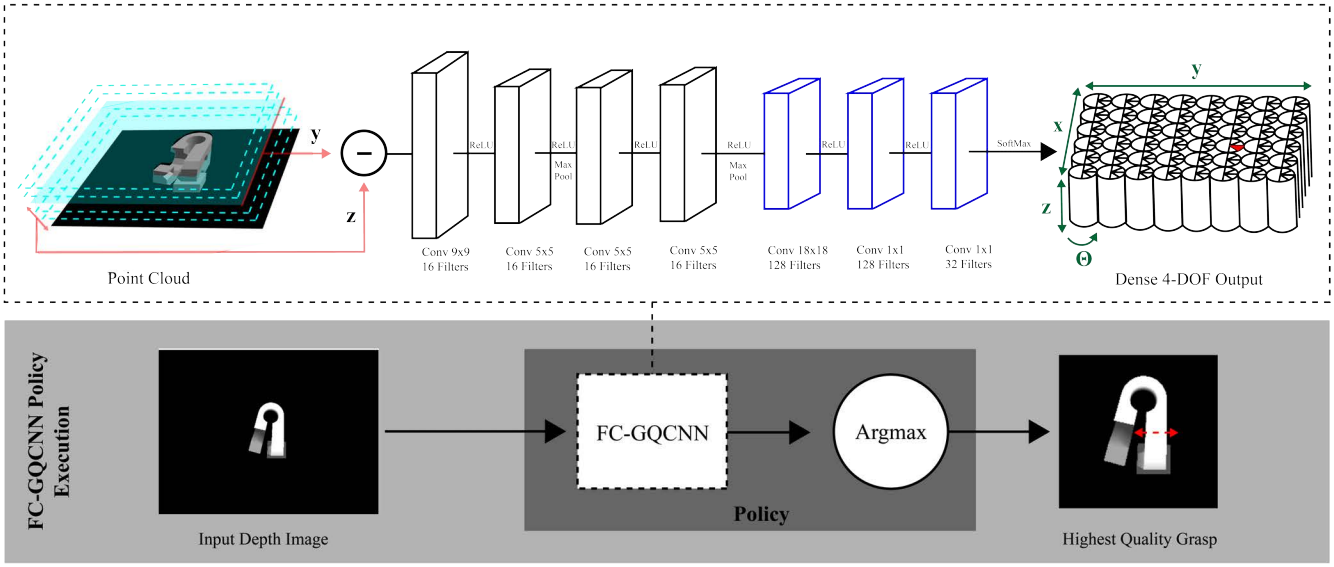

Fig. 2: Architecture for a grasping policy based on a Fully Convolutional Grasp Quality CNN (FC-GQ-CNN). (Top) Evaluation of depth image $\mathbf{y}$ using the FC-GQ-CNN. On the right we can see the dense 4-DOF output. Note the convolutional layers in the latter half of the network (highlighted in blue), which were originally fully connected layers in the angular GQ-CNN architecture. (Bottom) Given a full-size depth image, the policy evaluates the FC-GQ-CNN and takes the argmax to return the highest-quality grasp, highlighted in red in the 4-DOF FC-GQ-CNN output.

angular bin. The network weights are initialized using a Kaiming initializer. The network architecture and optimization framework are written in Python using Tensorflow. All training was done on Ubuntu 16.04 with an NVIDIA Titan Xp and an Intel Core i7-6850K clocked at 3.6 GHz.

VIII. Experiments

We run experiments on singulated objects in a quasi-static simulator to characterize the effect of training distribution and policy action space granularity on FC-GQ-CNN policy performance. In addition to simulation, we perform experiments on a physical robot with a set of 25 novel objects placed in a bin to simulate clutter. All experiments were performed on a desktop running Ubuntu 16.04 with an NVIDIA Titan Xp and an Intel Core i7-6850k clocked at 3.6GHz. Physical experiments were performed on an ABB YuMi with custom silicone fingertips [6] and a high-res Photoneo PhoXi depth sensor (See Fig. 3).

A. Object Set

We train on objects sampled from 1664 3D CAD models we collected from Thingiverse [1], augmented with synthetic backing to simulate the blister-pack found in industrial packaging.

B. Training Distribution

We characterize the effect of training distribution on policy performance in simulation on a set of 10 objects with varying geometry. The 10-object subset was chosen from the 1,664 object dataset to reflect common geometries [1]. We train an FC-GQ-CNN policy on datasets of varying size (measured in unique states per object) sampled with the following data collection policies:

1) Uniform 3D Antipodal Action Space (APD-3D) [23]

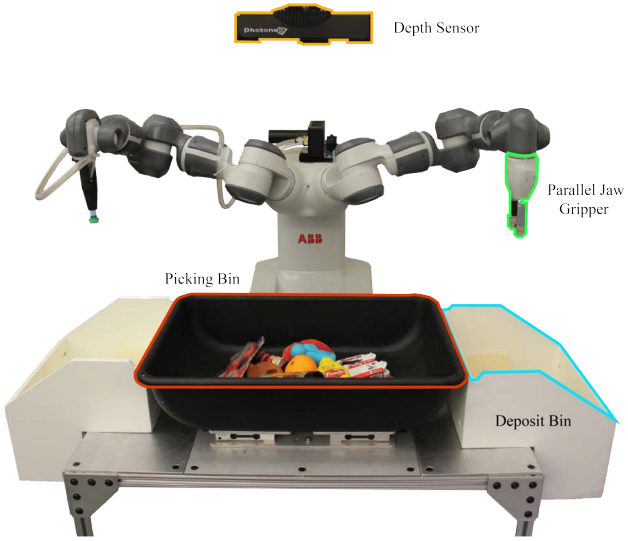

Fig. 3: The experimental setup consisting of an ABB YuMi with custom silicone fingertips and a high-res Photoneo PhoXi depth sensor. The experimental objective is to move objects from the picking bin to the deposit bin.

2) Uniform 2D Antipodal Action Space with Supervisor Guidance (APD-2D-SUP) [22]
3) 4D Discrete Action Space with Supervisor Guidance (FC-GQ-CNN-SUP)

We choose these specific distributions as a spectrum from fully off-policy (1) to our proposed on-policy approach (3). Policy (2) is chosen as an intermediate because it contains grasps closer to those evaluated by the learned FC-GQ-CNN policy, but still constrained by antipodality.

We evaluate the reliability of the resulting learned policies

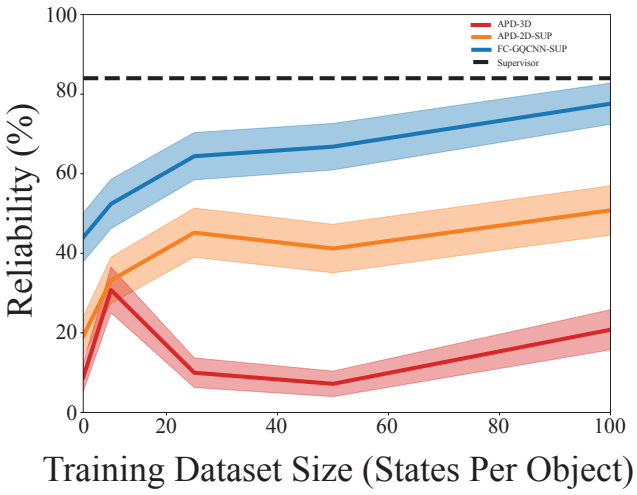

Fig. 4: Reliability of the FC-GQ-CNN policy versus the size of the training dataset over 250 evaluations using three data collection policies: 3D antipodal grasps sampled from object models [23], 2D antipodal grasps sampled from depth images (APD-2D-SUP) with supervisor guidance [22], and the proposed supervisor guidance with a dense discretization of the 4-DOF grasp action space (FC-GQ-CNN-SUP). Training dataset size is measured in terms of the number of unique states of 10 3D objects used to generate training data. Training with the FC-GQ-CNN-SUP distribution (the proposed method) leads to comparable performance with a supervisor that uses full state information to plan robust grasps, while the other distributions lead to decreased reliability that appears to asymptote below the performance of the supervisor.

in simulation for 250 evaluations of grasping each object from the 10-object training set. On each evaluation, the object placed in a stable resting pose in a given 2D position on a planar worksurface, the policy plans a parallel-jaw grasp, and the grasp is evaluated with robust quasi-static wrench resistance [**?**] for a known direction of gravity. To test whether or not performance differences are due to sample approximation error, we evaluate reliability over increasing dataset sizes by varying the number of unique positions and orientations of each object from 5 to 100.

Fig. 4 shows the results. Across all dataset sizes, the policy trained with the FC-GQ-CNN-SUP on-policy training distribution performs significantly better than the other policies. Furthermore, it is the only policy that reaches comparable performance to the supervisor as the dataset size is increased. The policies trained on APD-3D and APD-2D-SUP appear to asymptote well below the performance of the supervisor. We hypothesize that APD-2D-SUP outperforms APD-3D because it is a larger subset of the FC-GQ-CNN-SUP training distribution, but lacks sufficient coverage of the learned policy action space to achieve the performance of the supervisor policy.

C. Sensitivity

The granularity of the policy action space can have a significant impact on the speed and reliability of dense approaches, in particular the trade-off between the two. A very high granularity will result in a very precise policy, however producing a dense-enough output for this granularity will be computationally expensive and require a long Grasp Computation Time (GCT). On the other extreme, a low granularity will result in a policy that is quick to evaluate due to significantly reduced computation, but is imprecise because it never evaluates many grasps, some of which could be robust.

We characterize the effect of the policy action space granularity on performance in simulation using the same set of 10 objects as used in the training distribution experiments. We train an FC-GQ-CNN on the FC-GQ-CNN-SUP distribution. However, now we independently vary the number of angular bins k and stride s in the FC-GQ-CNN architecture, and the number of depth bins d used in the FC-GQ-CNN policy. Fig. 5 shows the results of experiments. The goal is to maximize reliability while minimizing Grasp Computation Time (GCT). We find that the optimal choice of these parameters is $p = 4, n = 16, k = 16$, that is we evaluate the image in pixel-wise strides of 4, bin the depth into 16 bins, and have angular bins of size $180/16 = 11.25\,\text{deg}$.

D. Novel Objects in Clutter

Robots in warehouses must be able to pick not only singulated objects, but more importantly objects in dense clutter. In order to test generalization and performance in clutter, we train an FC-GQ-CNN on the FC-GQ-CNN-SUP distribution with objects placed in heaps (which simulates real-world clutter). We then test the policy's ability to clear a bin of 25 novel objects (Fig. 6). We compare performance on 5 rollouts against a carefully tuned parallel jaw heuristic and GQ-CNN [22] trained using the APD-2D training distribution, which is on-policy for the standard GQ-CNN. We find that the FC-GQ-CNN policy performs best overall, achieving 296 MPPH. Furthermore, the policy appears to be able to locate more robust grasps than the iterative-optimization-based policy, as hypothesized. This is substantiated by the high average precision of the GQ-CNN policy, which indicates that failing grasps had low predicted quality. Results are shown in Table I.

E. Efficiency of FC-GQ-CNN Policy

We can quantify the efficiency of the proposed FC-GQ-CNN policy with the sheer millions of grasps it evaluates in a single pass of the policy and the amortized time per individual grasp as shown in Table II. This 1,200x speedup in computation time per grasp significantly outperforms previous iterative sampling and ranking policies such as the cross-entropy method.

IX. Discussion and Future Work

In this paper, we explored how to choose the optimal training distribution in order to maximize policy performance measured by rate and reliability. We presented the a novel on-policy data collection policy that combines grasps sampled from the policy action space along with guiding samples from a supervisor with full state knowledge to guide the distribution towards more robust grasps. We used this distribution to train a novel 4-DOF FC-GQ-CNN policy,

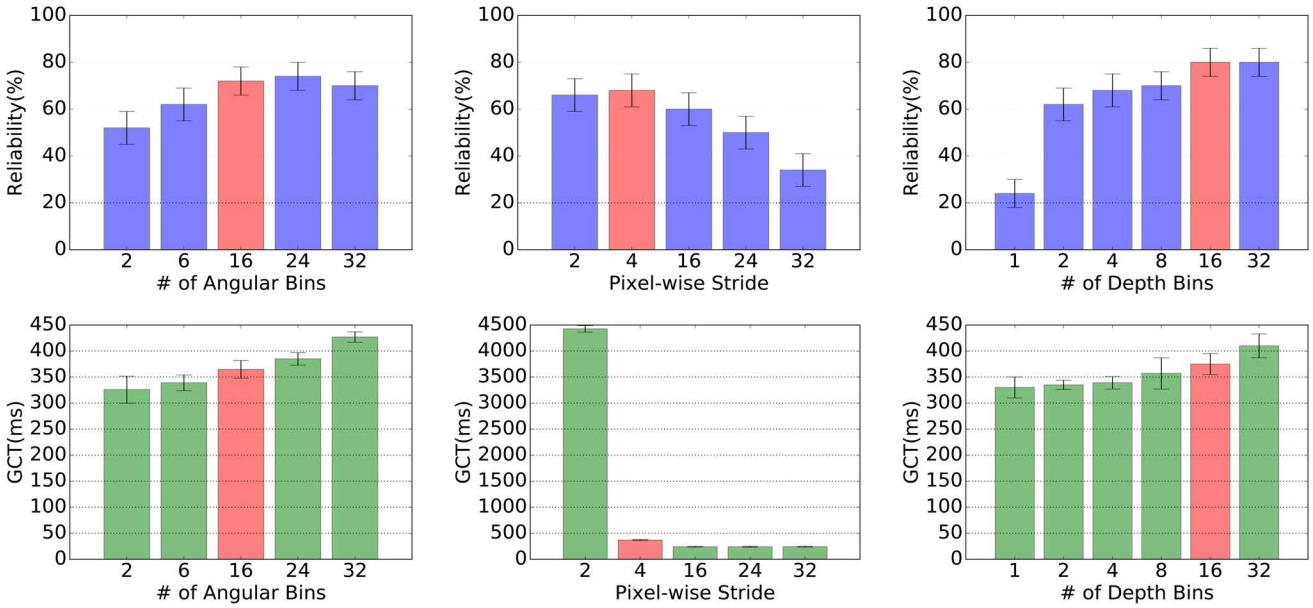

Fig. 5: Sensitivity of FC-GQ-CNN performance to granularity of action space, in particular varying # of angular bins, pixel-wise stride, and # of depth bins. The objective is to minimize Grasp Computation Time (GCT) while maximizing reliability. Highlighted in red are the optimal choice of values.

Policy	Reliability(%)	AP(%)	GCT(s)	MPPH
Parallel Jaw Heuristic	53.4	77.1	2.0	162
GQ-CNN	75.8	96.0	1.5	250
GQ-CNN(*)	81.2	93.8	3.0	236
FC-GQ-CNN	**85.6**	**95.2**	**0.6**	**296**

TABLE I: Performance of PJ Heuristic, GQ-CNN, and FC-GQ-CNN on bin picking with 25 novel objects on a physical robot. GQ-CNN(*) is a version of GQ-CNN with increased cross-entropy samples, increasing performance of the cross-entropy method at the cost of speed. The FC-GQ-CNN outperforms the GQ-CNN and PJ heuristic in rate and reliability. Seeing that GQ-CNN(*) achieves a lower reliability despite maintaining high precision, we hypothesize that the FC-GQ-CNN is able to locate grasps closer to the global optimum than GQ-CNN is unable to find, presumably limited by the iterative optimization approach.

Policy	GCT(s)	# Evaluations	CTPG(ms)
GQ-CNN	1.485	400	3.7125
FC-GQ-CNN	0.625	**2,008,064**	**0.0003**

TABLE II: Comparison of the number of grasps evaluated and evaluation time per grasp for GQ-CNN and FC-GQ-CNN on a 386x516 depth image. The FC-GQ-CNN efficiently evaluates a vastly larger set of grasps: ((386 - 96) / 4 + 1) * ((516 - 96) / 4 + 1) * 16 * 16 = 2,008,064 with a far lower computation time per grasp (CTPG). This may enable the policy to find higher-quality grasps than policies that evaluate a fewer grasps with iterative optimization.

Fig. 6: (Left) The 25 novel objects with diverse geometries used to test generalization. (Right) The objects arranged in a bin to simulate the dense clutter found in many de-cluttering and warehouse automation tasks.

which quickly and efficiently evaluates millions of grasp candidates in order to find the best one. Experiments on a physical robot demonstrate that this policy achieves 296 MPPH, compared to the 250 MPPH for policies based on iterative grasp sampling and evaluation.

In future work, we hope to extend the FC-GQ-CNN architecture to predict grasp quality for multiple depths at once. The hope is that moving this out of the policy and into the network architecture will result in a further speedup. We aim to also consider 6-DOF grasps that approach at an angle by parameterizing grasps in spherical coordinates. Furthermore, we plan to consider multi-fingered grippers or suction approaches such as Dex-Net 3.0 [22]. Finally, we plan to study alternate neural network architectures such as an end-to-end composite architecture consisting of a initial bounding box detector that extracts regions of the image with high probability of having robust grasps, along with an FC-GQ-CNN to evaluate these regions. This way we can expect further computational speedups as the FC-GQ-CNN no longer needs to evaluate the entire image.

ACKNOWLEDGMENTS

This research was performed at the AUTOLAB at UC Berkeley

in affiliation with the Berkeley AI Research (BAIR) Lab, Berkeley Deep Drive (BDD), the Real-Time Intelligent Secure Execution (RISE) Lab, and the CITRIS "People and Robots" (CPAR) Initiative. The authors were supported in part by donations from Siemens, Google, Amazon Robotics, Toyota Research Institute, Autodesk, ABB, Samsung, Knapp, Loccioni, Honda, Intel, Comcast, Cisco, Hewlett-Packard and by equipment grants from PhotoNeo, NVidia, and Intuitive Surgical. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Sponsors. We thank our colleagues who provided helpful feedback and suggestions, in particular Matthew Matl, Bill DeRose, Mike Danielczuk, Jonathan Lee, Michelle Lu, Lucas Manuelli, David Tseng, Daniel Seita, and Kate Sanders.

REFERENCES

[1] Thingiverse online 3d object database. [Online]. Available: https://www.thingiverse.com/

[2] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesisa survey," *IEEE Trans. Robotics*, vol. 30, no. 2, pp. 289–309, 2014.

[3] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige *et al.*, "Using simulation and domain adaptation to improve efficiency of deep robotic grasping," *arXiv preprint arXiv:1709.07857*, 2017.

[4] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.

[5] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," *arXiv preprint arXiv:1803.11469*, 2018.

[6] M. Guo, D. V. Gealy, J. Liang, J. Mahler, A. Goncalves, S. McKinley, J. A. Ojea, and K. Goldberg, "Design of parallel-jaw gripper tip surfaces for robust grasping," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2831–2838.

[7] A. Gupta, A. Murali, D. Gandhi, and L. Pinto, "Robot learning in homes: Improving generalization and reducing dataset bias," *arXiv preprint arXiv:1807.07049*, 2018.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2980–2988.

[9] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, G. Dulac-Arnold *et al.*, "Deep q-learning from demonstrations," *arXiv preprint arXiv:1704.03732*, 2017.

[10] E. Johns, S. Leutenegger, and A. J. Davison, "Deep learning a grasp function for grasping under gripper pose uncertainty," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4461–4468.

[11] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke *et al.*, "Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation," *arXiv preprint arXiv:1806.10293*, 2018.

[12] D. Kappler, J. Bohg, and S. Schaal, "Leveraging big data for grasp planning," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2015.

[13] S. Kumra and C. Kanan, "Robotic grasp detection using deep convolutional neural networks," *arXiv preprint arXiv:1611.08036*, 2016.

[14] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg, "Dart: Noise injection for robust imitation learning," *arXiv preprint arXiv:1703.09327*, 2017.

[15] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *Int. Journal of Robotics Research (IJRR)*, vol. 34, no. 4-5, pp. 705–724, 2015.

[16] S. Levine and V. Koltun, "Guided policy search," in *International Conference on Machine Learning*, 2013, pp. 1–9.

[17] 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," *Int. Journal of Robotics Research (IJRR)*, vol. 37, no. 4-5, pp. 421–436, 2018.

[18] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *arXiv preprint arXiv:1603.02199*, 2016.

[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[20] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[21] J. Mahler and K. Goldberg, "Learning deep policies for robot bin picking by simulating robust grasping sequences," in *Conference on Robot Learning*, 2017, pp. 515–524.

[22] 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," in *Proc. Robotics: Science and Systems (RSS)*, 2017.

[23] 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 *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2016.

[24] D. Morrison, P. Corke, and J. Leitner, "Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach," in *Proc. Robotics: Science and Systems (RSS)*, 2018.

[25] J. Oberlin and S. Tellex, "Autonomously acquiring instance-based object models from experience," in *Int. S. Robotics Research (ISRR)*, 2015.

[26] E. Parisotto, J. L. Ba, and R. Salakhutdinov, "Actor-mimic: Deep multitask and transfer reinforcement learning," *arXiv preprint arXiv:1511.06342*, 2015.

[27] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2016.

[28] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer handbook of robotics*. Springer, 2008, pp. 671–700.

[29] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1316–1322.

[30] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[31] M. T. Rosenstein and A. G. Barto, "Reinforcement learning with supervision by a stable controller," in *American Control Conference, 2004. Proceedings of the 2004*, vol. 5. IEEE, 2004, pp. 4517–4522.

[32] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 627–635.

[33] R. S. Sutton, A. G. Barto *et al.*, *Reinforcement learning: An introduction*. MIT press, 1998.

[34] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp pose detection in point clouds," *The International Journal of Robotics Research*, vol. 36, no. 13-14, pp. 1455–1473, 2017.

[35] 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 *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 23–30.

[36] M. Veres, M. Moussa, and G. W. Taylor, "Modeling grasp motor imagery through deep conditional generative models," *arXiv preprint arXiv:1701.03041*, 2017.

[37] U. Viereck, A. t. Pas, K. Saenko, and R. Platt, "Learning a visuomotor controller for real world robotic grasping using easily simulated depth images," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[38] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3119–3127.

[39] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo *et al.*, "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2018.