

Learning Robotic Grasping Policies for Suction Cups and Parallel Jaws in Simulation

*Matthew Matl
Ruzena Bajcsy, Ed.
Ken Goldberg, Ed.*



Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2019-119

<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2019/EECS-2019-119.html>

August 16, 2019

Copyright © 2019, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Acknowledgement

(Abbreviated Version)

To Professor Ken Goldberg, for his advice, feedback, and mentorship.

To Professor Ruzena Bajcsy, for her kindness and insightful questions.

To Dr. Jeffrey Mahler, for his mentorship, guidance, and hard work.

To all of my other collaborators, for their insight and effort.

To the NDSEG fellowship, for funding.

To my family, for their love and support.

To Carolyn, my wife, by biggest supporter, and my best friend, for her constant love and encouragement.

**Learning Robotic Grasping Policies for Suction Cups and Parallel Jaws in
Simulation**

by

Matthew Matl

A project report submitted in partial satisfaction of the

requirements for the degree of

Master of Science

in

Electrical Engineering and Computer Science

in the

Graduate Division

of the

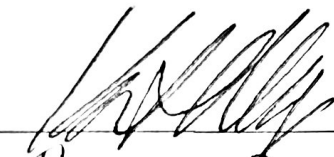
University of California, Berkeley

Committee in charge:

Professor Ken Goldberg, Chair
Professor Ruzena Bajcsy

Summer 2019

The project report of Matthew Matl, titled Learning Robotic Grasping Policies for Suction Cups and Parallel Jaws in Simulation, is approved:

Chair 
Ruzena Bajic

Date 15 AUG 2019
Date 16 AUG. 2019
Date

Abstract

Learning Robotic Grasping Policies for Suction Cups and Parallel Jaws in Simulation

by

Matthew Matl

Master of Science in Electrical Engineering and Computer Science

University of California, Berkeley

Professor Ken Goldberg, Chair

Robots in material-handling applications will need the ability to grasp and manipulate the huge variety of items that pass through warehouses every day. Handling this level of diversity requires the use of multiple end effectors, as each class of manipulator cannot handle certain types of objects. For example, parallel-jaw grippers fail on larger items, while vacuum-based end effectors cannot grasp porous or tiny objects. In this report, I present a summary of published research conducted by my colleagues and myself – specifically, Dex-Net 3.0 and Dex-Net 4.0 – that proposes methods for planning grasps for multiple classes of grippers on unknown objects.

In Dex-Net 3.0, we propose a compliant suction contact model that geometrically computes the quality of the seal between a suction cup and a target surface and measures the ability of a suction grasp to resist an external gravity wrench. We then use this model to generate millions of suction grasp attempts in simulation and train a Grasp Quality Convolutional Neural Network (GQ-CNN) to predict grasp outcomes from synthetically-rendered point clouds. We evaluate the resulting system in 350 physical trials on an ABB YuMi fitted with a pneumatic suction gripper and achieve success rates of 98% on basic objects and 82% on more complex objects.

In Dex-Net 4.0, we propose a unified contact model that brings the Dex-Net 3.0 suction model together with a point-contact model for parallel-jaw grippers. With this unified model, we generate a joint training dataset of over 5 million simulated grasps for both grippers and train GQ-CNNs for each gripper on that dataset. By selecting grasps from the GQ-CNN with the highest predicted confidence, we can decide between using a suction-cup end-effector and a parallel-jaw gripper when grasping objects from bins. In physical experiments, the Dex-Net 4.0 system consistently clears bins of up to 25 novel objects with a reliability greater than 95%.

Code, datasets, the published papers, and supplemental material can be found at <http://berkeleyautomation.github.io/dex-net>.

Contents

Contents	i
List of Figures	ii
List of Tables	v
1 Introduction and Related Work	1
1.1 Overview	1
1.2 Introduction	3
1.3 Related Work	5
2 Planning Suction Grasps	8
2.1 Introduction	8
2.2 Problem Statement	8
2.3 Compliant Suction Contact Model	10
2.4 Dex-Net 3.0 Dataset	15
2.5 Learning a Deep Robust Grasping Policy	16
2.6 Experiments	17
3 Planning Ambidextrous Grasps	21
3.1 Introduction	21
3.2 Problem Statement	21
3.3 Methods	23
3.4 Physical Experiments	28
3.5 Discussion	33
4 Conclusion and Future Work	39
Bibliography	41

List of Figures

- 2.1 The quasi-static spring model, C , used for determining when seal formation is feasible. The model contains three types of springs – perimeter, flexion, and cone springs. An initial state for C is chosen given a target point \mathbf{p} and an approach direction \mathbf{v} . Then, a contact state for C is computed so that C 's perimeter springs form a complete seal against object mesh M . Seal formation is deemed feasible if the energy required to maintain this contact state is sufficiently low in every spring. 11
- 2.2 Our compliant suction contact model. **(Left)** The quasi-static spring model used in seal formation computations. This suction cup is approximated by $n = 8$ components. Here, r is equal to the radius of the cup and h is equal to the height of the cup. $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ are the base vertices and \mathbf{a} is the apex. **(Right)** Wrench basis for the suction ring contact model. The contact exerts a constant pulling force on the object of magnitude V and additionally can push or pull the object along the contact z axis with force f_z . The suction cup material exerts a normal force $f_N = f_z + V$ on the object through a linear pressure distribution on the ring. This pressure distribution induces a friction limit surface bounding the set of possible frictional forces in the tangent plane $f_t = (f_x, f_y)$ and the torsional moment τ_z , and also induces torques τ_x and τ_y about the contact x and y axes due to elastic restoring forces in the suction cup material. 12
- 2.3 The Dex-Net 3.0 dataset. **(Left)** The Dex-Net 3.0 object dataset contains approximately 350k unique suction target points across the surfaces of 1,500 3D models from the KIT object database [31] and 3DNet [71]. Each suction grasp is classified as robust (green) or non-robust (red). Robust grasps are often above the object center-of-mass on flat surfaces of the object. **(Right)** The Dex-Net 3.0 point cloud dataset contains 2.8 million tuples of point clouds and suction grasps with robustness labels, with approximately 11.8% positive examples. 15

2.4	Pipeline for generating the Dex-Net 3.0 dataset (left to right). We first sample a candidate suction grasp from the object surface and evaluate the ability to form a seal and resist gravity over perturbations in object pose, gripper pose, and friction. The samples are used to estimate the probability of success, or robustness, for candidate grasps on the object surface. We render a point cloud for each object and associate the candidate grasp with a pixel and orientation in the depth image through perspective projection. Training datapoints are centered on the suction target pixel and rotated to align with the approach axis to encode the invariance of the robustness to image locations.	15
2.5	(Left) The experimental setup with an ABB YuMi equipped with a suction gripper. (Right) The 55 objects used to evaluate suction grasping performance. The objects are divided into three categories to characterize performance: Basic (e.g. prismatic objects), Typical, and Adversarial.	17
3.1	Learning ambidextrous grasping policies for universal picking. (Top) Synthetic training datasets of depth images, grasps, and rewards are generated from a set of 3D CAD models using analytic models based on physics and domain randomization. Specifically, a data collection policy proposes actions given a simulated heap of objects and the synthetic training environment evaluates rewards. Reward is computed consistently across grippers by considering the ability of a grasp to resist a given wrench (force and torque) based on the grasp wrench space, or set of wrenches that the grasp can resist through contact. (Middle) For each gripper, a policy is trained by optimizing a deep Grasp Quality CNN (GQ-CNN) to predict the probability of grasp success given a point cloud over a large training dataset containing millions of synthetic examples from the training environment. Datapoints are labeled as successes (blue) or failures (red) according to the analytic reward metric. (Bottom) The ambidextrous policy is deployed on the real robot to select a gripper by maximizing grasp quality using a separate GQ-CNN for each gripper.	24
3.2	Physical benchmark for evaluating universal picking policies. (Top) The robot plans a grasp to iteratively transport each object from the picking bin (green) to a receptacle (blue) using either a suction cup or parallel-jaw gripper. Grasp planning is based on 3D point clouds from an overhead Photoneo PhoXi S industrial depth camera. (Bottom) Performance is evaluated on two datasets of novel test objects not used in training. (Left) The Level 1 objects consist of prismatic and circular solids (e.g. boxes, cylinders) spanning groceries, toys, and medicine. (Right) The Level 2 objects are more challenging, including common objects with clear plastic and varied geometry such as products with cardboard “blisterpack” packaging.	30

- 3.3 Performance of the Dex-Net 4.0 ambidextrous policy on the bin picking benchmark. Error bars show the 95% confidence interval on reliability using the standard error of the mean. **(A)** Comparison with three baseline methods on the Level 1 and Level 2 objects on heaps of 25 novel objects: (1) a hand-coded heuristic for the suction cup, (2) a hand-coded heuristic for selecting between the suction cup and parallel-jaw gripper, and (3) an ambidextrous policy fine-tuned on simulated heaps from the Dex-Net 2.0 and 3.0 base GQ-CNNs and reward metrics. For reference, the best possible performance (succeeding on every pick until the bin is cleared) is illustrated with a black-dotted line. **(B)** Performance with large heaps of 50 novel objects. **(C)** Ablation study measuring the effect of training on less diverse datasets with either fewer unique heaps or fewer unique 3D object models. **(D)** Performance of two training alternatives: the Improved GQ-CNN architecture [26] and fine-tuning on 13k real datapoints. 37
- 3.4 Failure modes of the Dex-Net 4.0 policy. Error bars show the 95% confidence interval on reliability using the standard error of the mean. **(A)** Performance on Level 3: a dataset of 25 novel objects with adversarial geometry and material properties. **(B)** Evaluation of a first-order memory-based policy for universal picking that masks regions of the point cloud with an instance recognition system to avoid repeated failures. **(C)** Pathological objects which cannot be grasped with Dex-Net 4.0 due to reflectance properties such as transparency, which affect depth sensing, and material properties such as porosity and deformability (e.g. loose packaging), which affect the ability to form a vacuum seal on the object surface. 38

List of Tables

2.1	Performance of point-cloud-based grasping policies for 125 trials each on the Basic and Typical datasets and 100 trials each on the Adversarial dataset. We see that the GQ-CNN trained on Dex-Net 3.0 has the highest Average Precision (AP) (area under the precision-recall curve) on the Basic and Typical objects, suggesting that the robustness score from the GQ-CNN could be used to anticipate grasp failures and select alternative actions (e.g. probing objects) in the context of a larger system. Also, a GQ-CNN trained on the Adversarial dataset outperforms all methods on the Adversarial objects, suggesting that the performance of our model is improved when the true object models are used for training.	19
3.1	Detailed performance analysis of the Dex-Net 4.0 and baseline policies on the bin picking benchmark for five trials each on the Level 1 and Level 2 datasets of 25 novel objects each. We report the reliability, mean picks per hour (MPPH), average precision (AP), total number of grasps attempts (minimum of 125), and total number of failures.	32

Acknowledgments

This project would not have been possible without the support, encouragement, and hard work of a huge number of people, and thanking them all here is impossible. However, I would like to specifically thank a few people who made a particularly large impact on me during my time at Berkeley.

First, I'd like to thank Professor Ken Goldberg, my academic advisor. His constant advice, feedback, and attention to detail pushed me to become a better communicator and researcher, and his many years of research on robotic manipulation laid the groundwork for the work presented here. Without his support, mentorship and leadership throughout every facet of our research, this work would not have been possible, and without his sharp eye and careful feedback, the publications generated by this work would not have been nearly as polished or professional.

I'd like to also thank Professor Ruzena Bajcsy, who has been a kind and inspirational presence throughout my time at Berkeley. Her insightful questions have helped me to understand how to approach fundamental research with a scientific mind, and her encouragement has been instrumental to my success.

Dr. Jeffrey Mahler has also had a huge impact on my time at Berkeley. His mentorship helped me to find a path in my own research, and his hard work and dedication to physical experimentation in the Dex-Net project inspired me to be a better scientist. Without his work and assistance, this project would not have happened.

I had the pleasure of working with other talented graduate and undergraduate students during my time at Berkeley, including Michael Danielczuk, Ashwin Balakrishna, Andrey Kurenkov, Vishal Satish, Andrew Lee, Andrew Li, David Tseng, David Wang, Amit Talreja, Ruta Joshi, and many others. To all those who helped to shape my work in any way, I extend my heartfelt thanks.

During my time at Berkeley, I was funded by an NDSEG grant from the United States Department of Defense and the Army Research Office. I'd like to thank those organizations for the financial security and support they gave me that enabled me to research freely.

I'd like to also extend a special thanks to my family. Mom, Dad, James, Christina, Mary, Anthony, David, and John – all of you sacrificed so much to help me get to where I am, and I can never thank you enough.

Finally, to Carolyn, my wife, my biggest supporter, and my best friend. You know how much you mean to me. Nothing I do would be possible without you, and I love you so much.

Chapter 1

Introduction and Related Work

1.1 Overview

As we move further into the core of the twenty-first century, robotics continues to capture an ever-increasing slice of the general public’s imagination. The media has created grand visions of a future where dangerous and menial labor is done by machines, where elderly people can age at home with dignity with the assistance of their automated helpers, and where logistics, agriculture, and manufacturing are conducted efficiently around-the-clock by an army of intelligent robots. This grand vision has served as inspiration for roboticists around the world, but several fundamental problems still separate us from achieving this ideal.

For a robot to serve a useful purpose, it must perceive and understand the world around it and then interact with the objects and people in its vicinity to achieve its goals. Perception and scene understanding remain open problems in robotic research, with threads of inquiry running the gamut from 6-DoF pose estimation to object instance recognition to semantic understanding of natural object states and task-based affordances. Additionally, physical robotic interaction – particularly, grasping, reorientation, and intelligent placement of objects – remains a research area with many rich avenues ripe for exploration. Inspired by Professor Ken Goldberg’s many years of research on robotic grasping and Dr. Jeffrey Mahler’s initial work on the Dex-Net project, I dedicated my time at Berkeley to developing tools for enabling robots to perceive and grasp objects in unstructured environments such as warehouses, homes, and flexible factories.

My first project involved using the cloud to offer simple grasp planners as a service for lower-power robots with limited onboard computation [64]. This project helped me to find my footing in the robotic manipulation research space, and quickly led to a second project, which utilized computational geometry algorithms to semantically segment and label subcomponents of CAD models. With these semantic segmentations and a shape-descriptor similarity metric, I was able to demonstrate a way to transfer task-specific grasps from one object to similar objects (e.g. from the handle of one coffee mug to the handles of all similar

coffee mugs) [45].

While I was excited by these initial successes, they still required an exact CAD model of the target object for planning grasps. In the highly unstructured scenarios that the grand vision of robotics demands, it’s impossible to guarantee access to CAD data for every object that the robot might encounter. Jeff, Ken, and the co-authors of Dex-Net 2.0 [39] made a large breakthrough on this problem by demonstrating that it was possible to train a deep neural network to predict parallel-jaw grasp outcomes using only simulated images of grasps that were generated and labeled in a quasi-static simulation. This allowed us to plan grasps on objects that we’d never seen before without an expensive training process and significantly upgraded the ability of our robots to operate in unstructured environments.

However, while the Dex-Net 2.0 system worked well for parallel-jaw grippers, such end-effectors are inherently limited. Specifically, objects larger than the gripper’s maximum opening width present a fundamental obstacle, and cluttered or tightly-packed bins present few affordances for the insertion of the gripper’s fingers. In practice, alternative grasping methods such as vacuum-based suction-cup end effectors are often used on objects that parallel-jaw grippers cannot lift.

Inspired by this challenge, Jeff, Ken, and I set off to extend the techniques used in Dex-Net 2.0 to suction-based end effectors, which resulted in Dex-Net 3.0 [40]. We then unified the contact models from Dex-Net 2.0 and 3.0 to create a fair way to compare grasps with differing modalities, and we used this unified model to generate data for training an “ambidextrous” grasping policy to decide between using a parallel-jaw gripper or a suction cup in a bin-picking application [41]. Our efforts resulted in a state-of-the-art system for grasping unknown objects that is highly capable – even when presented complex, adversarial objects in cluttered heaps and bins. The development of the models, training systems, grasp planning techniques, and experiments used in these later stages of the Dex-Net project are the focus of this project report.

Towards the end of my Masters, I began to collaborate with other researchers to utilize the grasp planning techniques created by the Dex-Net project in practical applications and scenarios. With Michael Danielczuk, Jeff, Ken, and several other researchers, I helped to demonstrate a technique for semantically segmenting object instances in images of cluttered bins by training the Mask R-CNN network on a large dataset of synthetic depth images [10]. By combining this semantic segmentation technique with traditional instance recognition techniques and the grasp planners from the Dex-Net project, we were able to make a first attempt at “mechanical search” – the process of intelligently sorting through clutter (e.g. a large bin of diverse items) to find and extract a target item of interest [8].

In addition to this published research, I have existing threads of work that are focused on extending the capabilities of our robots past grasping and towards intelligent placement and packing. I’ve made some preliminary progress on techniques for fast 6-DoF pose estimation for known objects and methods for packing unknown objects into rectangular containers in a space-efficient manner, and although these works are not yet ready for publication, I am excited to continue working on them further into my PhD. I have also open-sourced and maintained a significant portion of our codebase for other robotics researchers to use,

including our core tools for working with images, point clouds, and kinematic structures and an easy-to-use renderer for Python programmers. I have great respect for the open-source community and wish to continue contributing tools for others to use in the future. Finally, I have also collected, filtered and pruned a dataset of over 1500 CAD models that our lab has used repeatedly for Dex-Net and other related projects. It is my hope that this dataset will continue enabling effective research in our lab in the future.

At the beginning of my PhD, the grand vision of robotics seemed to be a long way off. Now, as scientific advances pour in from every direction, the vision seems much closer. Robots can now interact with, grasp, and place items that they haven't seen before. They can search for objects that they need, identify a way to sift through clutter, and operate in environments with less structure than they required even a few short years ago. In the remainder of my PhD, I'm excited to continue building on the fundamental research needed to continue introducing robots into new environments and bringing the grand vision ever closer to completion.

1.2 Introduction

In this report, I present a summary of work that my colleagues and I published in Dex-Net 3.0 [40] at ICRA 2018 and Dex-Net 4.0 [41] in Science Robotics in 2019. Before diving into the introduction, I want to acknowledge my co-authors, especially Dr. Jeff Mahler, the Dex-Net project lead, and Professor Ken Goldberg, the project's PI, for their significant contributions to these works.

In logistics and e-commerce, workers handle thousands of different products every day, from tiny pillpacks to stuffed animals to enormous cardboard boxes. While the rapid rise of online shopping has accelerated the need for material-handling labor, automation efforts have been stymied by the sheer diversity of products that a robot would need to handle to achieve human-level productivity. The ever-changing, massively-varied inventories in modern warehouses present a host of challenges for both grasp planning and end-effector design.

In these situations, grasp planners that rely on exact knowledge of objects [49, 54] tend to scale poorly [3, 4]. Sensor noise and occlusions make it difficult to estimate object geometries and poses on-the-fly, and physical parameters such as friction coefficients and mass distributions cannot be observed directly. Recent approaches have attempted to deal with this uncertainty by using machine learning to train function approximators, such as deep neural networks, to predict grasp outcomes from images using large training datasets of empirical successes and failures. While these methods can generalize to a large number of unknown objects, collecting data from humans [30, 35, 61] or physical experiments [36, 52, 73] can be time-consuming and expensive.

With Dex-Net 2.0 [39], my colleagues demonstrated a way to obtain the generalizability of data-driven grasp planning without requiring expensive data collection. Specifically, they showed that a Grasp Quality Convolutional Neural Network (GQ-CNN) could be trained to predict real-world grasp outcomes from point clouds despite being trained exclusively on

synthetic point clouds generated from simulated grasp attempts. By training the network to maximize a grasp quality metric derived from mechanical wrench space analysis [49], the authors were able to generate a high-performance grasping policy for a parallel-jaw gripper that generalized across hundreds of different objects.

While Dex-Net 2.0 greatly reduced the cost of creating generalizable parallel-jaw grasping policies, planning for only a single class of end-effectors is insufficient for a warehousing system. Unlike human hands, robot end-effectors often have significant drawbacks that prevent them from manipulating entire classes of objects. For example, parallel-jaw grippers cannot grasp objects larger than the opening width of their jaws. In industry, a common way to boost the range of a robotic picking system is by using a combination of end-effectors – most commonly, vacuum-based suction grippers, which have been shown to handle a wide variety of common objects in applications such as the Amazon Picking Challenge [20, 48, 72, 73].

However, while a great deal of literature exists on parallel-jaw and multifinger grasp planning [4], comparatively little research has been published on planning suction grasps. These grasps are often planned directly on point clouds using heuristics such as grasping near the object centroid [20] or at the center of planar surfaces [6, 12]. These heuristics work well for prismatic objects such as boxes and cylinders, but they can fail on objects with non-planar surfaces near the object centroid (e.g. industrial parts and household objects such as staplers or children’s toys). Analytic models of suction cups for grasp planning exist, but they typically assume that a vacuum seal has already been formed and that the state (e.g. shape and pose) of the object is perfectly known [2, 33, 44]. In practice a robot may need to form seals on non-planar surfaces while being robust to external wrenches (e.g. gravity and disturbances), sensor noise, control imprecision, and calibration errors.

In order to enhance the capabilities of existing manipulation systems, we extended Dex-Net 2.0 to suction-based grippers with our work in Dex-Net 3.0. We proposed a compliant suction contact model that geometrically computes the quality of the seal between a suction cup and a target surface and measures the ability of a suction grasp to resist an external gravity wrench. Using this model, we generated millions of suction grasp attempts in simulation and trained a Grasp Quality Convolutional Neural Network (GQ-CNN) to predict grasp outcomes from synthetically-rendered point clouds. We evaluated the resulting system in 350 physical trials on an ABB YuMi fitted with a pneumatic suction gripper and achieved success rates of 98% on basic objects and 82% on more complex objects.

A natural next step was combining the Dex-Net 2.0 and 3.0 systems to create an “am-bidextrous” grasp planner that dynamically chooses the best gripper type to use. In Dex-Net 4.0, we proposed a unified contact model that brings together our physics models for fingertips and suction cups. Using this model, we generated a training dataset of over 5 million simulated parallel-jaw and suction grasps on objects piled into bins. By training a separate GQ-CNN for each gripper on this data and maximizing over the predictions of the two networks, we can dynamically select which gripper to use in a bin-picking system. In physical experiments, the Dex-Net 4.0 system consistently cleared bins of up to 25 novel objects with a reliability greater than 95%.

In this project report, I describe our models, methods, and experiments from Dex-Net 3.0 and 4.0, including these primary contributions:

1. A compliant suction contact model that quantifies seal formation using a quasi-static spring system and the ability to resist external wrenches (e.g. gravity) using a contact wrench basis derived from the ring of contact between the cup and object surface.
2. A method for training a GQ-CNN to predict suction grasp quality from point clouds by generating simulated training data with the compliant suction contact model.
3. Physical robot experiments measuring the precision of the GQ-CNN-based suction grasp planning policy with and without knowledge of the target object’s shape and pose.
4. A POMDP framework for “ambidextrous” robot grasping with a reward function based on a unified contact model.
5. An ambidextrous grasping policy trained on simulated rollouts of this POMDP that plans a grasp by maximizing over the predictions of separate GQ-CNNs for each gripper.
6. Experiments evaluating the performance of this policy in a bin picking application with a parallel-jaw and suction-cup gripper.

The final Dex-Net 4.0 policy achieves up to 95% grasp reliability on heaps of up to 25 diverse, novel objects.

1.3 Related Work

End-effectors based on suction are widely used in industrial applications such as warehouse order fulfillment, handling limp materials such as fabric [33], and robotics applications such as the Amazon Picking Challenge [6], underwater manipulation [62], or wall climbing [2]. Our methods for planning suction grasps and generating policies for choosing between suction-based grippers and finger-based grippers build on models of deformable materials, analyses of the wrenches that suction cups and fingertips can exert, and data-driven grasp planning.

Suction Models

Several models for the deformation of stiff rubber-like materials exist in the literature. Provot et al. [55] modeled sheets of stiff cloth and rubber using a spring-mass system with several types of springs. Hosseini et al. [1] provided a survey of more modern constitutive models of rubber that are often used in Finite Element Analysis (FEA) packages for more realistic physics simulations. In order to rapidly evaluate whether a suction cup can form a seal

against an object’s surface, we model the cup as a quasi-static spring system with a topology similar to the one in [55] and estimate the deformation energy required to maintain a seal.

In addition, several models have been developed to check for static equilibrium assuming a seal between the suction cup and the object’s surface. Most models consider the suction cup to be a rigid object and model forces into the object along the surface normal, tangential forces due to surface friction, and pulling forces due to suction [33, 62, 68]. Bahr et al. [2] augmented this model with the ability to resist moments about the center of suction to determine the amount of vacuum pressure necessary to keep a climbing robot attached to a vertical wall. Mantriota [44] modeled torsional friction due to a contact area between the cup and object similar to the soft finger contact model used in grasping [29]. Our model extends these methods by combining models of torsional friction [44] and contact moments [2] in a compliant model of the ring of contact between the cup and object.

Grasp Planning

The goal of grasp planning is to select a configuration for an end-effector that enables a robot to perform a task via contact with an object while resisting external perturbations [4], which can be arbitrary [14] or task-specific [34]. A common approach for parallel-jaw grippers is to select a configuration that maximizes a quality metric (or reward) based on wrench space analysis [49], robustness to perturbations [70], or a model learned from human labels [30, 35, 61] or self-supervision [52, 36, 73].

Several similar metrics exist for evaluating suction grasps. One common approach is to evaluate whether or not a set of suction cups can lift an object by applying an upwards force [33, 62, 67, 68]. Domae et al. [12] developed a geometric model to evaluate suction success by convolving target locations in images with a desired suction contact template to assess planarity. Heuristics for planning suction grasps from point clouds have also been used extensively in the Amazon Robotics Challenge. In 2015, Team RBO [13] won by pushing objects from the top or side until suction was achieved, and Team MIT [72] came in second place by suctioning on the centroid of objects with flat surfaces. In 2016, Team Delft [20] won the challenge by approaching the estimated object centroid along the inward surface normal. In 2017, Cartman [48] won the challenge and ranked suction grasps according to heuristics such as maximizing distance to the segmented object boundary and MIT [74] performed well using a neural network trained to predict grasp affordance maps from human labeled RGB-D point clouds. In Dex-Net 3.0, we present a novel metric that evaluates whether a single suction cup can resist external wrenches under perturbations in object / gripper poses, friction coefficient, and disturbing wrenches.

In applications such as the Amazon Robotics Challenge, many teams expanded the range of their systems by equipping robots with more than one end-effector (e.g. both a parallel-jaw gripper and a suction cup) [20, 48, 72, 73]. The policies that decide which gripper to use at runtime are typically hand-coded by domain experts, which can make them difficult to tune and challenging to extend to new cameras, grippers, and robots. In Dex-Net 4.0, we

present a method for learning this sort of “ambidextrous” policy by generating training data for both grippers in simulation with a reward function based on a unified contact model.

Dex-Net 3.0 and 4.0 also extend empirical, data-driven approaches to grasp planning based on images and point clouds [4]. A popular approach is to use human labels of graspable regions in RGB-D images [35] or point clouds [30] to learn a grasp detector with computer vision techniques. As labeling may be tedious for humans, an alternative is to automatically collect training data from a physical robot [36, 52]. To reduce the time-cost of data collection, recent research has proposed to generate labels in simulation using physical models of contact [27, 30] and domain randomization [27, 50, 63, 60, 65] to assist in attenuating the problems associated with sim-to-reality transfer [25, 51, 58, 65]. Mahler et al. [39] demonstrated that a GQ-CNN trained on Dex-Net 2.0, a dataset of 6.7 million point clouds, grasps, and quality labels computed in simulation with robust quasi-static analysis, could be used to successfully plan parallel-jaw grasps across a wide variety of objects with 99% precision. Later, Mahler and Goldberg [38] extended the Dex-Net 2.0 technique to bin picking by simulating sequences of simulated grasps on objects piled on tables or in totes. In both Dex-Net 3.0 and 4.0, we use a similar approach to generate a dataset of point clouds, grasps, and robustness labels for suction cups and parallel-jaw grippers.

Chapter 2

Planning Suction Grasps

2.1 Introduction

In this chapter, I present the results of Dex-Net 3.0, our paper describing a method for planning suction grasps from point clouds that does not require explicit object models or expensive manual data collection. The method presented here works by (1) creating a suitable model of suction contact that can be used to evaluate grasps in simulation, (2) using that model to generate simulated point clouds of grasps with associated success and failure labels, and (3) using that simulated data to train a GQ-CNN to directly predict grasp outcomes from point clouds. In the following sections, we formalize the problem of suction grasp planning, describe our contact model and data generation procedure, and explain our experiments that tested the efficacy of the method.

My contributions to this section of our work include the following:

- Inventing, programming, and testing the seal formation model.
- Contributing to the final formulation of the wrench resistance metric.
- Designing, implementing, and running physical experiments.
- Writing and revising the final manuscript.

2.2 Problem Statement

Given a point cloud from a depth camera, our goal is to find a robust suction grasp (target point and approach direction) for a robot to lift an object in isolation on a planar work-surface and transport it to a receptacle. We compute the suction grasp that maximizes the probability that the robot can hold the object under gravity and perturbations sampled from a distribution over sensor noise, control imprecision, and random disturbing wrenches.

Assumptions

Our stochastic model makes the following assumptions:

1. Quasi-static physics (e.g. inertial terms are negligible) with Coulomb friction.
2. Objects are rigid and made of non-porous material.
3. Each object is singulated on a planar worksurface in a stable resting pose [16].
4. A single overhead depth sensor with known intrinsics, position, and orientation relative to the robot.
5. A vacuum-based end-effector with known geometry and a single disc-shaped suction cup made of linear-elastic material.

Definitions

A robot observes a single-view **point cloud** or depth image, \mathbf{y} , containing a singulated object. The goal is to find the most robust suction **grasp** \mathbf{u} that enables the robot to lift an object and transport it to a receptacle, where grasps are parametrized by a target point $\mathbf{p} \in \mathbb{R}^3$ and an approach direction $\mathbf{v} \in \mathcal{S}^2$. Success is measured with a binary **grasp reward function** R , where $R = 1$ if the grasp \mathbf{u} successfully transports the object, and $R = 0$ otherwise.

The robot may not be able to predict the success of suction grasps exactly from point clouds for several reasons. First, the success metric depends on a **state** \mathbf{x} describing the object’s geometric, inertial, and material properties \mathcal{O} and the pose of the object relative to the camera, T_o , but the robot does not know the true state due to: (a) noise in the depth image and (b) occlusions due to the single viewpoint. Second, the robot may not have perfect knowledge of external wrenches (forces and torques) on the object due to gravity or external disturbances.

This probabilistic relationship is described by an **environment** consisting of a grasp success distribution modeling $\mathbb{P}(R \mid \mathbf{x}, \mathbf{u})$, the ability of a grasp to resist random disturbing wrenches, and an observation model $p(\mathbf{y} \mid \mathbf{x})$. This model induces a probability of success for each grasp conditioned on the robot’s observation:

Definition 1 *The robustness of a grasp \mathbf{u} given a point cloud \mathbf{y} is the probability of grasp success under uncertainty in sensing, control, and disturbing wrenches: $Q(\mathbf{y}, \mathbf{u}) = \mathbb{P}(R \mid \mathbf{y}, \mathbf{u})$.*

Our environment model is described in Section 2.4.

Objective

Our ultimate goal is to find a grasp that maximizes robustness given a point cloud, $\pi^*(\mathbf{y}) = \operatorname{argmax}_{\mathbf{u} \in \mathcal{C}} Q(\mathbf{y}, \mathbf{u})$, where \mathcal{C} specifies constraints on the set of available grasps, such as collisions or kinematic feasibility. We approximate π^* by optimizing the weights θ of a deep Grasp Quality Convolutional Neural Network (GQ-CNN) $Q_\theta(\mathbf{y}, \mathbf{u})$ on a training dataset $\mathcal{D} = \{(\mathbf{y}_i, \mathbf{u}_i, R_i)\}_{i=1}^N$ consisting of reward values, point clouds, and suction grasps sampled from our stochastic model of grasp success. Our optimization objective is to find weights θ that minimize the cross-entropy loss \mathcal{L} over \mathcal{D} :

$$\theta^* = \operatorname{argmin}_{\theta \in \Theta} \sum_{i=1}^N \mathcal{L}(R_i, Q_\theta(\mathbf{y}_i, \mathbf{u}_i)). \quad (2.2.1)$$

2.3 Compliant Suction Contact Model

To quantify grasp robustness, we develop a quasi-static spring model of the suction cup material and a model of contact wrenches that the suction cup can apply to the object through a ring of contact on the suction cup perimeter. Under our model, the reward $R = 1$ if:

1. A seal is formed between the perimeter of the suction cup and the object surface.
2. Given a seal, the suction cup is able to resist an external wrench on the object due to gravity and disturbances.

Seal Formation

A suction cup can lift objects due to an air pressure differential induced across the membrane of the cup by a vacuum generator that forces the object into the cup. If a gap exists between the perimeter of the cup and the object, then air flowing into the gap may reduce the differential and cause the grasp to fail. Therefore, a tight seal between the cup and the target object is important for achieving a successful grasp.

To determine when seal formation is possible, we model circular suction cups as a conical spring system \mathcal{C} parameterized by real numbers (n, r, h) , where n is the number of vertices along the contact ring, r is the radius of the cup, and h is the height of the cup. See Fig. 2.2 for an illustration.

Rather than performing a computationally expensive dynamic simulation with a spring-mass model to determine when seal formation is feasible, we make simplifying assumptions to evaluate seal formation geometrically. Specifically, we compute a configuration of \mathcal{C} that achieves a seal by projecting \mathcal{C} onto the surface of the target object’s triangular mesh \mathcal{M} and evaluate the feasibility of that configuration under quasi-static conditions as a proxy for the dynamic feasibility of seal formation.

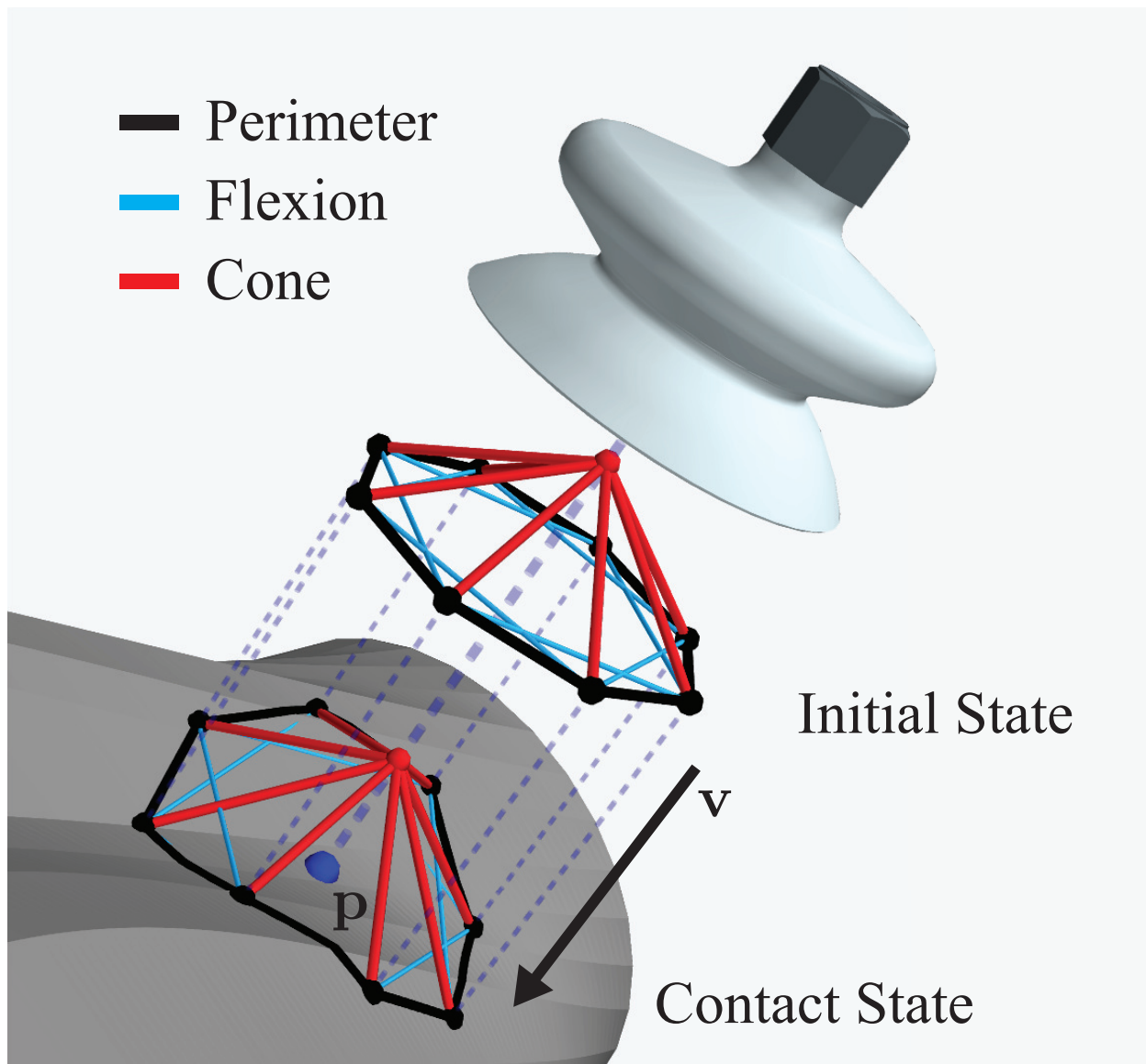


Figure 2.1: The quasi-static spring model, C , used for determining when seal formation is feasible. The model contains three types of springs – perimeter, flexion, and cone springs. An initial state for C is chosen given a target point \mathbf{p} and an approach direction \mathbf{v} . Then, a contact state for C is computed so that C 's perimeter springs form a complete seal against object mesh M . Seal formation is deemed feasible if the energy required to maintain this contact state is sufficiently low in every spring.

In our model, C has two types of springs – *structural* springs that represent the physical structure of the suction cup and *flexion* springs that do not correspond to physical structures but instead are used to resist bending along the cup's surface. Dynamic spring-mass systems with similar structures have been used in prior work to model stiff sheets of rubber [55]. The undeformed structural springs of C form a right pyramid with height h and with a base that

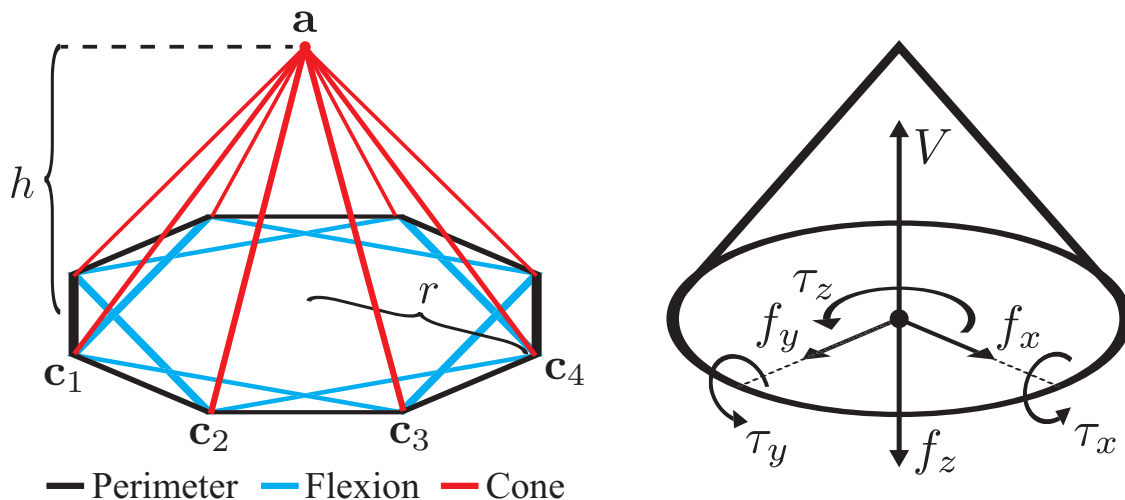


Figure 2.2: Our compliant suction contact model. **(Left)** The quasi-static spring model used in seal formation computations. This suction cup is approximated by $n = 8$ components. Here, r is equal to the radius of the cup and h is equal to the height of the cup. $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ are the base vertices and \mathbf{a} is the apex. **(Right)** Wrench basis for the suction ring contact model. The contact exerts a constant pulling force on the object of magnitude V and additionally can push or pull the object along the contact z axis with force f_z . The suction cup material exerts a normal force $f_N = f_z + V$ on the object through a linear pressure distribution on the ring. This pressure distribution induces a friction limit surface bounding the set of possible frictional forces in the tangent plane $f_t = (f_x, f_y)$ and the torsional moment τ_z , and also induces torques τ_x and τ_y about the contact x and y axes due to elastic restoring forces in the suction cup material.

is a regular n -gon with circumradius r . Let $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n, \mathbf{a}\}$ be the set of vertices of the undeformed right pyramid, where each \mathbf{v}_i is a base vertex and \mathbf{a} is the pyramid's apex. We define the model's set of springs as follows:

- **Perimeter (Structural) Springs** – Springs linking vertex \mathbf{v}_i to vertex \mathbf{v}_{i+1} , $\forall i \in \{1, \dots, n\}$.
- **Cone (Structural) Springs** – Springs linking vertex \mathbf{v}_i to vertex \mathbf{a} , $\forall i \in \{1, \dots, n\}$.
- **Flexion Springs** – Springs linking vertex \mathbf{v}_i to vertex \mathbf{v}_{i+2} , $\forall i \in \{1, \dots, n\}$.

In the model, a complete seal is formed between \mathcal{C} and \mathcal{M} if and only if each of the perimeter springs of \mathcal{C} lies entirely on the surface of \mathcal{M} . Given a target mesh \mathcal{M} with a target grasp $\mathbf{u} = (\mathbf{p}, \mathbf{v})$ for the gripper, we choose an initial configuration of \mathcal{C} such that \mathcal{C} is undeformed and the approach line (\mathbf{p}, \mathbf{v}) passes through \mathbf{a} and is orthogonal to the base of \mathcal{C} . Then, we make the following assumptions to determine a final static contact configuration of \mathcal{C} that forms a complete seal against \mathcal{M} (see Fig. 2.1):

- **The perimeter springs** of \mathcal{C} must not deviate from the original undeformed regular n -gon when projected onto a plane orthogonal to \mathbf{v} . This means that their locations can be computed by projecting them along \mathbf{v} from their original locations onto the surface of \mathcal{M} .

- **The apex, \mathbf{a} ,** of \mathcal{C} must lie on the approach line (\mathbf{p}, \mathbf{v}) and, given the locations of \mathcal{C} 's base vertices, must also lie at a location that keeps the average distance along \mathbf{v} between \mathbf{a} and the perimeter vertices equal to h .

Given this configuration, a seal is feasible if:

- The cone faces of \mathcal{C} do not collide with \mathcal{M} during approach or in the contact configuration.
- The surface of \mathcal{M} has no holes within the contact ring traced out by \mathcal{C} 's perimeter springs.
- The energy required in each spring to maintain the contact configuration of \mathcal{C} is below a real-valued threshold E modeling the maximum deformation of the suction cup material against the object surface.

We threshold the energy in individual springs rather than the total energy for \mathcal{C} because air gaps are usually caused by local geometry.

Wrench Space Analysis

To determine the degree to which the suction cup can resist external wrenches such as gravity, we analyze the set of wrenches that the suction cup can apply.

Wrench Resistance

The object wrench set for a grasp using a contact model with m basis wrenches is $\Lambda = \{\mathbf{w} \in \mathbb{R}^6 \mid \mathbf{w} = G\alpha \text{ for some } \alpha \in \mathcal{F}\}$, where $G \in \mathbb{R}^{6 \times m}$ is a set of basis wrenches in the object coordinate frame, and $\mathcal{F} \subseteq \mathbb{R}^m$ is a set of constraints on contact wrench magnitudes [49].

Definition 2 *A grasp \mathbf{u} achieves wrench resistance with respect to \mathbf{w} if $-\mathbf{w} \in \Lambda$ [34, 49].*

We encode wrench resistance as a binary variable W such that $W = 0$ if \mathbf{u} resists \mathbf{w} and $W = 1$ otherwise.

Suction Contact Model

Many suction contact models acknowledge normal forces, vacuum forces, tangential friction, and torsional friction [2, 33, 44, 62] similar to a point contact with friction or soft finger model [49]. However, under this model, a single suction cup cannot resist torques about axes in the contact tangent plane, implying that any torque about such axes would cause the suction cup to drop an object. This defies our intuition since empirical evidence suggests that a single point of suction can robustly transport objects [13, 20].

We hypothesize that these torques are resisted through an asymmetric pressure distribution on the ring of contact between the suction cup and object, which occurs due to passive

elastic restoring forces in the material. Fig. 2.2 illustrates the suction ring contact model. The grasp map G is defined by the following basis wrenches:

1. **Actuated Normal Force** (f_z): The force that the suction cup material applies by pressing into the object along the contact z axis.
2. **Vacuum Force** (V): The magnitude of the constant force pulling the object into the suction cup coming from the air pressure differential.
3. **Frictional Force** ($f_f = (f_x, f_y)$): The force in the contact tangent plane due to the normal force between the suction cup and object, $f_N = f_z + V$.
4. **Torsional Friction** (τ_z): The torque resulting from frictional forces in the ring of contact.
5. **Elastic Restoring Torque** ($\tau_e = (\tau_x, \tau_y)$): The torque about axes in the contact tangent plane resulting from elastic restoring forces in the suction cup pushing on the object along the boundary of the contact ring.

The magnitudes of the contact wrenches are constrained due to (a) the friction limit surface [29], (b) limits on the elastic behavior of the suction cup material, and (c) limits on the vacuum force. In the suction ring contact model, \mathcal{F} is approximated by a set of linear constraints for efficient computation of wrench resistance:

$$\begin{array}{lll}
 \textbf{Friction:} & \sqrt{3}|f_x| \leq \mu f_N & \sqrt{3}|f_y| \leq \mu f_N & \sqrt{3}|\tau_z| \leq r\mu f_N \\
 \textbf{Material:} & \sqrt{2}|\tau_x| \leq \pi r \kappa & \sqrt{2}|\tau_y| \leq \pi r \kappa & \\
 \textbf{Suction:} & f_z \geq -V & &
 \end{array}$$

Here μ is the friction coefficient, r is the radius of the contact ring, and κ is a material-dependent constant modeling the maximum stress for which the suction cup has linear-elastic behavior. These constraints define a subset of the friction limit ellipsoid and cone of admissible elastic torques under a linear pressure distribution about the ring of the cup. Furthermore, we can compute wrench resistance using quadratic programming due to the linearity of the constraints.

Robust Wrench Resistance

We evaluate the robustness of candidate suction grasps by evaluating seal formation and wrench resistance over distributions on object pose, grasp pose, and disturbing wrenches:

Definition 3 *The robust wrench resistance metric for \mathbf{u} and \mathbf{x} is $\lambda(\mathbf{u}, \mathbf{x}) = \mathbb{P}(W \mid \mathbf{u}, \mathbf{x})$, the probability of success under perturbations in object pose, gripper pose, friction, and disturbing wrenches.*

In practice, we evaluate robust wrench resistance by taking J samples, evaluating binary wrench resistance for each, and computing the sample mean: $\frac{1}{J} \sum_{j=1}^J W_j$.

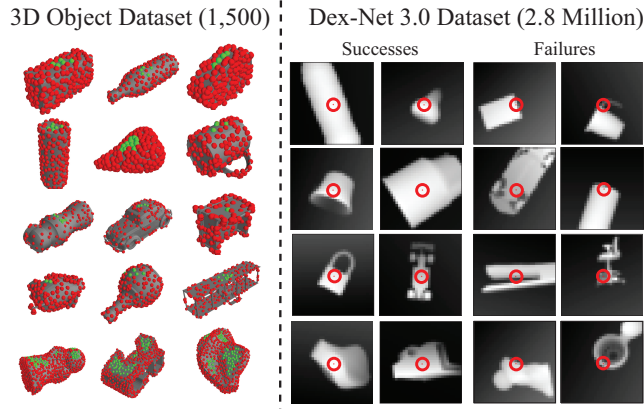


Figure 2.3: The Dex-Net 3.0 dataset. (Left) The Dex-Net 3.0 object dataset contains approximately 350k unique suction target points across the surfaces of 1,500 3D models from the KIT object database [31] and 3DNet [71]. Each suction grasp is classified as robust (green) or non-robust (red). Robust grasps are often above the object center-of-mass on flat surfaces of the object. (Right) The Dex-Net 3.0 point cloud dataset contains 2.8 million tuples of point clouds and suction grasps with robustness labels, with approximately 11.8% positive examples.

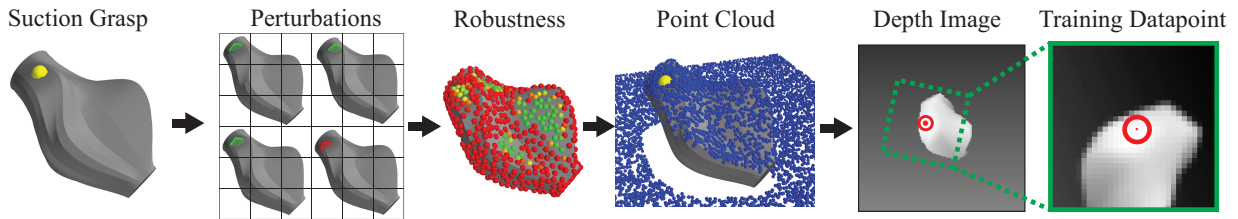


Figure 2.4: Pipeline for generating the Dex-Net 3.0 dataset (left to right). We first sample a candidate suction grasp from the object surface and evaluate the ability to form a seal and resist gravity over perturbations in object pose, gripper pose, and friction. The samples are used to estimate the probability of success, or robustness, for candidate grasps on the object surface. We render a point cloud for each object and associate the candidate grasp with a pixel and orientation in the depth image through perspective projection. Training datapoints are centered on the suction target pixel and rotated to align with the approach axis to encode the invariance of the robustness to image locations.

2.4 Dex-Net 3.0 Dataset

To learn to predict grasp robustness based on noisy point clouds, we generate the Dex-Net 3.0 training dataset of point clouds, grasps, and grasp reward labels by sampling tuples $(R_i, \mathbf{u}_i, \mathbf{y}_i)$ from a joint distribution $p(R, \mathbf{x}, \mathbf{y}, \mathbf{u})$ modeled as the product of distributions on:

- **States:** $p(\mathbf{x})$: A uniform distribution over a discrete dataset of objects and their stable poses and uniform continuous distributions over the object planar pose and camera poses in a bounded region of the workspace.
- **Grasp Candidates:** $p(\mathbf{u}|\mathbf{x})$: A uniform random distribution over contact points on the object surface.

- **Grasp Rewards** $p(R \mid \mathbf{u}, \mathbf{x})$: A stochastic model of wrench resistance for the gravity wrench that is sampled by perturbing the gripper pose according to a Gaussian distribution and evaluating the contact model described in Section 2.3.
- **Observations** $p(\mathbf{y} \mid \mathbf{x})$: A depth sensor noise model with multiplicative and Gaussian process pixel noise.

Fig. 2.3 illustrates a subset of the Dex-Net 3.0 object and grasp dataset. The parameters of the sampling distributions and compliant suction contact model $(n, r, h, E, V, \mu, \kappa, \epsilon)$ (see Section 2.3) were set by maximizing average precision of the Q values using grid search for a set of grasps attempted on an ABB YuMi robot on a set of known 3D printed objects (see Section 2.6).

Our pipeline for generating training tuples is illustrated in Fig. 2.4. We first sample state by selecting an object at random from a database of 3D CAD models and sampling a friction coefficient, planar object pose, and camera pose relative to the worksurface. We generate a set of grasp candidates for the object by sampling points and normals uniformly at random from the surface of the object mesh. We then set the binary reward label $R = 1$ if a seal is formed and robust wrench resistance (described in Section 2.3) is above a threshold value ϵ . Finally, we sample a point cloud of the scene using rendering and a model of image noise [42]. The grasp success labels are associated with pixel locations in images through perspective projection [18].

2.5 Learning a Deep Robust Grasping Policy

We use the Dex-Net 3.0 dataset to train a GQ-CNN that takes as input a single-view point cloud of an object resting on the table and a candidate suction grasp defined by a target 3D point and approach direction, and outputs the robustness, or estimated probability of success, for the grasp on the visible object.

Our GQ-CNN architecture is identical to Dex-Net 2.0 [39] except that we modify the pose input stream to include the angle between the approach direction and the table normal. The point cloud stream takes a depth image centered on the target point and rotated to align the middle column of pixels with the approach orientation similar to a spatial transforming layer [24]. The end-effector depth from the camera and orientation are input to a fully connected layer in a separate pose stream and concatenated with conv features in a fully connected layer. We train the GQ-CNN with using stochastic gradient descent with momentum using an 80-20 training-to-validation image-wise split of the Dex-Net 3.0 dataset. Training took approximately 12 hours on three NVIDIA Titan X GPUs. The learned GQ-CNN achieves 93.5% classification accuracy on the held-out validation set.

We use the GQ-CNN in a deep robust grasping policy to plan suction target grasps from point clouds on a physical robot. The policy uses the Cross Entropy Method (CEM) [37, 39, 57]. CEM samples a set of initial candidate grasps uniformly at random from the set of surface points and inward-facing normals on a point cloud of the object, then iteratively



Figure 2.5: **(Left)** The experimental setup with an ABB YuMi equipped with a suction gripper. **(Right)** The 55 objects used to evaluate suction grasping performance. The objects are divided into three categories to characterize performance: Basic (e.g. prismatic objects), Typical, and Adversarial.

resamples grasps from a Gaussian Mixture Model fit to the grasps with the highest predicted probability of success.

2.6 Experiments

We ran experiments to characterize the precision of robust wrench resistance when object shape and pose are known and the precision of our deep robust grasping policy for planning grasps from point clouds for three object classes.

Object Classes

We created a dataset of 55 rigid and non-porous objects including tools, groceries, office supplies, toys, and 3D printed industrial parts. We separated objects into three categories, illustrated in Fig. 2.5:

1. *Basic*: Prismatic solids (e.g. rectangular prisms, cylinders). Includes 25 objects.
2. *Typical*: Common objects with varied geometry and many accessible, approximately planar surfaces. Includes 25 objects.
3. *Adversarial*: 3D-printed objects with complex geometry (e.g. curved or narrow surfaces) that are difficult to access. Includes 5 objects.

For object details, see <http://bit.ly/2xMcx3x>.

Experimental Protocol

We ran experiments with an ABB YuMi with a Primesense Carmine 1.09 and a suction system with a 15mm diameter silicone single-bellow suction cup and a VM5-NC VacMotion vacuum generator with a payload of approximately 0.9kg. The experimental workspace is illustrated in the left panel of Fig. 2.5. In each experiment, the operator iteratively presented

a target object to the robot and the robot planned and executed a suction grasp on the object. The operator labeled successes based on whether or not the robot was able to lift and transport the object to the side of the workspace. For each method, we measured:

1. **Average Precision (AP).** The area under the precision-recall curve, which measures precision over possible thresholds on the probability of success predicted by the policy. This is useful for industrial applications where a robot may take an alternative action (e.g. asking for help) if the planned grasp is predicted to fail.
2. **Success Rate.** The fraction of all grasps that were successful.

All experiments ran on a Desktop running Ubuntu 14.04 with a 2.7 GHz Intel Core i5-6400 Quad-Core CPU and an NVIDIA GeForce 980 GPU.

Performance on Known Objects

To assess performance of our robustness metric independently from the perception system, we evaluated whether or not the metric was predictive of suction grasp success when object shape and pose were known using the 3D printed Adversarial objects (right panel of Fig. 2.5). The robot was presented one of the five Adversarial objects in a known stable pose, selected from the top three most probable stable poses. We hand-aligned the object to a template image generated by rendering the object in a known pose on the table. Then, we indexed a database of grasps precomputed on 3D models of the objects and executed the grasp with the highest metric value for five trials. In total, there were 75 trials per experiment.

We compared the following metrics:

1. **Planarity-Centroid (PC3D).** The inverse distance to the object centroid for sufficiently planar patches on the 3D object surface.
2. **Spring Stretch (SS).** The maximum stretch among virtual springs in the suction contact model.
3. **Wrench Resistance (WR).** Our model without perturbations.
4. **Robust Wrench Resistance (RWR).** Our model.

The RWR metric performed best with 99% AP compared to 93% AP for WR, 89% AP for SS, and 88% for PC3D.

Performance on Novel Objects

We also evaluated the performance of GQ-CNNs trained on Dex-Net 3.0 for planning suction target points from a single-view point cloud. In each experiment, the robot was presented one object from either the Basic, Typical, or Adversarial classes in a pose randomized by shaking the object in a box and placing it on the table. The object was imaged with a depth sensor

	Basic		Typical		Adversarial	
	AP (%)	Success Rate (%)	AP (%)	Success Rate (%)	AP (%)	Success Rate (%)
Planarity	81	74	69	67	48	47
Centroid	89	92	80	78	47	38
Planarity-Centroid	98	94	94	86	64	62
GQ-CNN (ADV)	83	77	75	67	86	81
GQ-CNN (DN3)	99	98	97	82	61	58

Table 2.1: Performance of point-cloud-based grasping policies for 125 trials each on the Basic and Typical datasets and 100 trials each on the Adversarial dataset. We see that the GQ-CNN trained on Dex-Net 3.0 has the highest Average Precision (AP) (area under the precision-recall curve) on the Basic and Typical objects, suggesting that the robustness score from the GQ-CNN could be used to anticipate grasp failures and select alternative actions (e.g. probing objects) in the context of a larger system. Also, a GQ-CNN trained on the Adversarial dataset outperforms all methods on the Adversarial objects, suggesting that the performance of our model is improved when the true object models are used for training.

and segmented using 3D bounds on the workspace. Then, the grasping policy executed the most robust grasp according to a success metric. In this experiment the human operators were blinded from the method they were evaluating to remove bias in human labels.

We compared policies that optimized the following metrics:

1. **Planarity.** The inverse sum of squared errors from an approach plane for points within a disc with radius equal to that of the suction cup.
2. **Centroid.** The inverse distance to the object centroid.
3. **Planarity-Centroid (PC).** The inverse distance to the centroid for planar patches on the 3D object surface.
4. **GQ-CNN (ADV).** Our GQ-CNN trained on synthetic data from the Adversarial objects (to assess the ability of the model to fit complex objects).
5. **GQ-CNN (DN3).** Our GQ-CNN trained on synthetic data from 3DNet [71], KIT [31], and the Adversarial objects.

Table 2.1 details performance on the Basic, Typical, and Adversarial objects. On the Basic and Typical objects, we see that the Dex-Net 3.0 policy is comparable to PC in terms of success rate and has near-perfect AP, suggesting that failed grasps often have low robustness and can therefore be detected. On the adversarial objects, GQ-CNN (ADV) significantly outperforms GQ-CNN (DN3) and PC, suggesting that this method can be used to successfully grasp objects with complex surface geometry as long as the training dataset closely matches the objects seen at runtime. The DN3 policy took an average of 3.0 seconds per grasp.

Failure Modes

The most common failure mode was attempting to form a seal on surfaces with surface geometry that prevent seal formation. This is partially due to the limited resolution of the depth sensor, as our seal formation model is able to detect the inability to form a seal on such surfaces when the geometry is known precisely. In contrast, the planarity-centroid metric performs poorly on objects with non-planar surfaces near the object centroid.

Chapter 3

Planning Ambidextrous Grasps

3.1 Introduction

Given the results of Dex-Net 2.0 [39], Dex-Net 2.1 [38], and Dex-Net 3.0 [40], a logical next step was to devise a method for generating policies that can choose between a set of end-effectors when picking unknown objects out of bins. In this chapter, we present our 2019 Science Robotics journal paper [41], Dex-Net 4.0, in which we devise a unified contact model for finger-based and suction-based end-effectors and use that model to generate simulated training data for two grippers (a parallel-jaw gripper and a single suction-cup gripper). Using this simulated data, we train a GQ-CNN for each gripper to predict grasp success probabilities from point clouds and execute an “ambidextrous” grasping policy by comparing the GQ-CNN outputs at runtime. Experiments suggest that a policy trained in this manner can outperform ambidextrous policies that use advanced heuristics to make gripper choices when picking out of bins of 25 unique, novel objects.

As a note, the following sections are replicated from [41]. All credit to all authors of that journal paper, especially Dr. Jeffrey Mahler and Professor Ken Goldberg. My contributions to this section of our work include the following:

- Revising the suction contact model for inclusion in the unified contact model.
- Revising and improving code for dataset generation, including physics simulation and rendering.
- Running physical experiments and collecting data.
- Revising the final manuscript.

3.2 Problem Statement

We consider the problem of ambidextrous grasping of a wide range of novel objects from cluttered heaps using a robot with a depth camera and two or more available grippers,

such as a vacuum-based suction cup gripper and/or a parallel-jaw gripper. To provide context for the metrics and approaches considered in experiments, we formalize this problem as a Partially Observable Markov Decision Process (POMDP) [28] in which a robot plans grasps in order to maximize expected reward (probability of grasp success) given imperfect observations of the environment.

A robot with an overhead depth camera views a heap of novel objects in a bin. On grasp attempt t , a robot observes a point cloud \mathbf{y}_t from the depth camera. The robot uses a policy $\mathbf{u}_t = \pi(\mathbf{y}_t)$ to plan a grasp action \mathbf{u}_t for a gripper g consisting of a 3D rigid position and orientation of the gripper $\mathbf{T}_g = (\mathbf{R}_g, \mathbf{t}_g) \in \mathbf{SE}(3)$. Upon executing \mathbf{u}_t , the robot receives a reward $R_t = 1$ if it successfully lifts and transports exactly one object from the bin to a receptacle and $R_t = 0$ otherwise. The observations and rewards depend on a latent state \mathbf{x}_t that is unknown to the robot and describes the geometry, pose, center of mass, and material properties of each object. After either the bin is empty or T total grasp attempts, the process terminates.

These variables evolve according to an environment distribution that reflects sensor noise, control imprecision, and variation in the initial bin state:

- **Initial State Distribution.** Let $p(\mathbf{x}_0)$ be a distribution over possible states of the environment that the robot is expected to handle due to variation in objects and tolerances in camera positioning.
- **Observation Distribution.** Let $p(\mathbf{y}_t | \mathbf{x}_t)$ be a distribution over observations given a state due to sensor noise and tolerances in the camera optical parameters.
- **Transition Distribution.** Let $p(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{u}_t)$ be a distribution over next states given the current state and grasp action due to imprecision in control and physics.

The goal is to learn a policy π to maximize the rate of reward, or mean picks per hour (MPPH) ρ , up to a maximum of T grasp attempts:

$$\rho(\pi) = \mathbb{E} \left[\frac{\left(\sum_{t=0}^{T-1} R_t \right)}{\left(\sum_{t=0}^{T-1} \Delta_t \right)} \right] \quad (3.2.1)$$

where T is the number of grasp attempts and Δ_t is the duration of executing grasp action \mathbf{u}_t , in hours. The expectation is taken with respect to the environment distribution:

$$p(\mathbf{x}_0, \mathbf{y}_0, \dots, \mathbf{x}_T, \mathbf{y}_T | \pi) = p(\mathbf{x}_0) \prod_{t=0}^{T-1} p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_{t+1} | \mathbf{x}_t, \pi(\mathbf{y}_t))$$

It is common to measure performance in terms of the mean rate ν and reliability Φ (also known as the success rate) of a grasping policy for a given range of objects:

$$\nu = 1/\mathbb{E}[\Delta_t] \quad \Phi(\pi) = \mathbb{E} \left[\frac{1}{T} \sum_{t=0}^{T-1} R_t \right]$$

If the time per grasp is constant, the MPPH is the product of rate and reliability: $\rho = \nu\Phi$.

3.3 Methods

Overview

We propose a hybrid approach to ambidextrous grasping that learns policies on synthetic training datasets generated using analytic models and domain randomization over a diverse range of objects, cameras, and parameters of physics for robust transfer from simulation to reality [50, 60, 65]. The method optimizes for a policy to maximize MPPH under the assumption of a constant time per grasp: $\pi^* = \operatorname{argmax}_{\pi} \Phi(\pi)$.

To learn a policy, the method uses a training dataset generation distribution based on models from physics and geometry, μ , to computationally synthesize a massive training dataset of point clouds, grasps, and reward labels for heterogeneous grippers. The distribution μ consists of two stochastic components: (1) a synthetic training environment $\xi(\mathbf{y}_0, R_0, \dots, \mathbf{y}_T, R_T | \pi)$ that can sample paired observations and rewards given a policy and (2) a data collection policy $\tau(\mathbf{u}_t | \mathbf{x}_t, \mathbf{y}_t)$ that can sample a diverse set of grasps using full state knowledge. The synthetic training environment simulates grasp outcomes by evaluating rewards according to the ability of a grasp to resist forces and torques due to gravity and random perturbations. The environment also stochastically samples heaps of 3D objects in a bin and renders depth images of the scene using domain randomization over the camera position, focal length, and optical center pixel. The dataset collection policy evaluates actions in the synthetic training environment using algorithmic supervision to guide learning toward successful grasps.

We explore large-scale supervised learning on samples from μ to train the ambidextrous policy π_{θ} across a set of two or more available grippers \mathcal{G} , as illustrated in Fig. 3.1. First, we sample a massive training dataset $\mathcal{D} = \{(R_i, \mathbf{y}_i, \mathbf{u}_i)\}_{i=1}^N$ from a software implementation of μ . Then, we learn a Grasp Quality Convolutional Neural Network $Q_{\theta,g}(\mathbf{y}, \mathbf{u}) \in [0, 1]$ to estimate the probability of success for a grasp with gripper g given a depth image. Specifically, we optimize the weights θ_g to minimize the cross entropy loss \mathcal{L} between the GQ-CNN prediction and the true reward over the dataset \mathcal{D} :

$$\theta_g^* = \operatorname{argmin}_{\theta_g \in \Theta} \sum_{(R_i, \mathbf{u}_i, \mathbf{y}_i) \in \mathcal{D}_g} \mathcal{L}(R_i, Q_{\theta}(\mathbf{y}_i, \mathbf{u}_i)) \quad (3.3.1)$$

where \mathcal{D}_g denotes the subset of the training dataset \mathcal{D} containing only grasps for gripper g . We construct a robot policy π_{θ} from the GQ-CNNs by planning the grasp that maximizes quality across all available grippers:

$$\pi_{\theta}(\mathbf{y}_t) = \operatorname{argmax}_{g \in \mathcal{G}} \left\{ \max_{\mathbf{u}_g \in \mathcal{U}_g} Q_{\theta,g}(\mathbf{y}_t, \mathbf{u}_g) \right\}.$$

where \mathcal{U}_g is a set of candidate grasps for gripper g sampled from the depth image.

To evaluate the method, we learn the Dexterity Network (Dex-Net) 4.0 ambidextrous policy on the Dex-Net 4.0 training dataset, which contains 5 million synthetic point clouds,

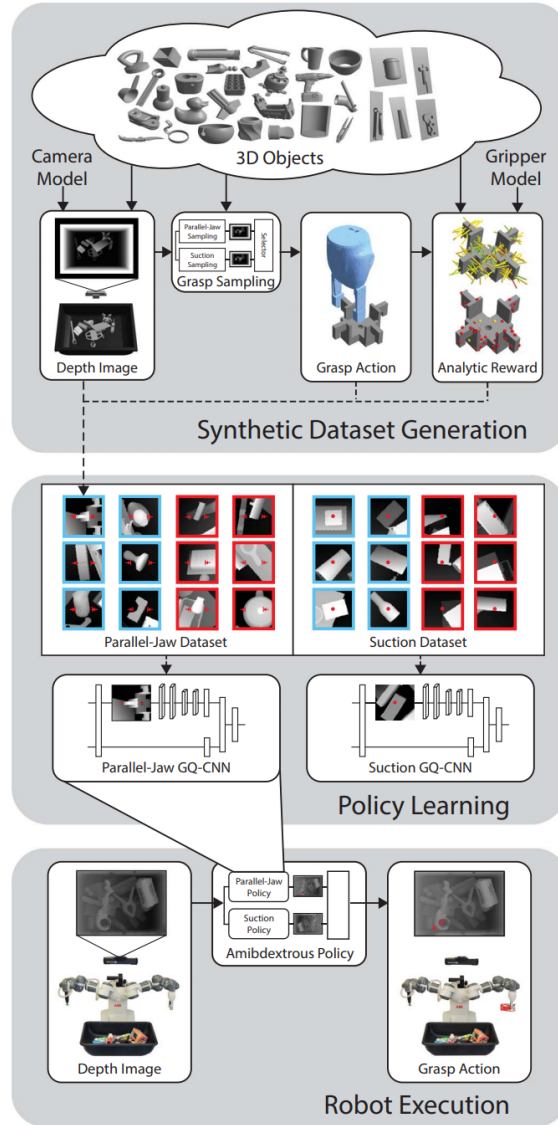


Figure 3.1: Learning ambidextrous grasping policies for universal picking. (Top) Synthetic training datasets of depth images, grasps, and rewards are generated from a set of 3D CAD models using analytic models based on physics and domain randomization. Specifically, a data collection policy proposes actions given a simulated heap of objects and the synthetic training environment evaluates rewards. Reward is computed consistently across grippers by considering the ability of a grasp to resist a given wrench (force and torque) based on the grasp wrench space, or set of wrenches that the grasp can resist through contact. (Middle) For each gripper, a policy is trained by optimizing a deep Grasp Quality CNN (GQ-CNN) to predict the probability of grasp success given a point cloud over a large training dataset containing millions of synthetic examples from the training environment. Datapoints are labeled as successes (blue) or failures (red) according to the analytic reward metric. (Bottom) The ambidextrous policy is deployed on the real robot to select a gripper by maximizing grasp quality using a separate GQ-CNN for each gripper.

grasps, and reward labels. Dex-Net 4.0 is generated from 5,000 unique 3D object heaps with approximately 2.5 million datapoints each for a vacuum-based suction cup gripper and a parallel-jaw gripper.

Details on the simulation environment, data collection policy, and network training are provided in the following sections.

Synthetic Training Environment

The Dex-Net 4.0 synthetic training environment is based on the following assumptions: (1) quasi-static physics (e.g., inertial terms are negligible) with Coulomb friction, (2) objects are rigid and made of non-porous material, (3) the robot has a single overhead depth sensor with known intrinsics, and (4) the robot has two end-effectors with known geometry: a vacuum-based gripper consisting of a single disc-shaped linear-elastic suction cup and a parallel-jaw gripper. Dex-Net 4.0 uses the following POMDP definition:

States. Let $\mathbf{x} = (\mathcal{O}_1, \dots, \mathcal{O}_m, \mathcal{C}, \mathbf{w}_1, \dots, \mathbf{w}_m)$ denote the state of the environment at time t , consisting of a single overhead depth camera, a set of objects, and a perturbation wrench on each object (e.g. gravity, disturbances). Each object state \mathcal{O}_i specifies the geometry \mathcal{M}_i , pose $\mathbf{T}_{\mathbf{o},i}$, friction coefficient γ_i , and center of mass \mathbf{z}_i . The camera state \mathcal{C} specifies the intrinsic parameters \mathcal{I} and pose $\mathbf{T}_{\mathbf{c}}$. Each wrench \mathbf{w}_i is specified as a vector $\mathbf{w}_i \in \mathbb{R}^6$.

Grasp Actions. Let $\mathbf{u}_s \in \mathcal{U}_s$ denote a suction grasp in 3D space defined by a suction gripper \mathcal{G}_s and a rigid pose of the gripper $\mathbf{T}_s = (\mathbf{R}_s, \mathbf{t}_s) \in \mathbf{SE}(\mathbf{3})$, where the rotation $\mathbf{R}_s \in \mathbf{SO}(\mathbf{3})$ defines the orientation of the suction tip and the translation $\mathbf{t}_s \in \mathbb{R}^3$ specifies the target location for the center of the suction disc. Let $\mathbf{u}_p \in \mathcal{U}_p$ denote a parallel-jaw grasp in 3D space defined by a parallel-jaw gripper \mathcal{G}_p and a rigid pose of the gripper $\mathbf{T}_p = (\mathbf{R}_p, \mathbf{t}_p) \in \mathbf{SE}(\mathbf{3})$, where the rotation $\mathbf{R}_p \in \mathbf{SO}(\mathbf{3})$ defines the grasp axis and approach direction, and the translation $\mathbf{t}_p \in \mathbb{R}^3$ specifies the target center point of the jaws. The set of all possible grasps is $\mathcal{U} = \mathcal{U}_s \cup \mathcal{U}_p$.

Point Clouds. Let $\mathbf{y} = \mathbb{R}_+^{H \times W}$ be a 2.5D point cloud represented as a depth image with height H and width W taken by a camera with known intrinsics [18].

State Distribution. The initial state distribution $\xi(\mathbf{x}_0)$ is the product of distributions on [38]:

1. *Object Count (m):* Poisson distribution with mean λ truncated to $[1, 10]$.
2. *Object Heap (\mathcal{O}):* Uniform distribution over 3D object models and the pose from which each model is dropped into the heap. Objects are sampled without replacement.
3. *Depth Camera (\mathcal{C}):* Uniform distribution over the camera pose and intrinsic parameters.
4. *Coulomb Friction (γ):* Truncated Gaussian constrained to $[0, 1]$.

The initial state is sampled by drawing an object count m , drawing a subset of m objects, and then dropping the objects one-by-one from a fixed height above the table and running

dynamic simulation with pybullet [7] until all objects have approximately zero velocity. The 3D object models are sampled from a dataset of 1,664 3D objects models selected to reflect a broad range of products that are commonly encountered in applications such as warehousing, manufacturing, or home decluttering. The dataset is augmented with synthetic “blisterpack” meshes to reflect cardboard-backed products encountered in retail applications. Augmentation was performed by placing each source mesh in a quasi-static stable resting pose [16] on an infinite planar worksurface and attaching a thin, flat segment to the mesh at the triangle(s) touching the worksurface.

Observation Distribution. Depth image observations are rendered using the open source Python library “meshrender” using randomization in the camera focal length and optical center pixel. No noise was added to the rendered images in order to model the high resolution Photoneo PhoXi industrial depth camera used in experiments.

Reward Distribution. Binary rewards occur when a quasi-static equilibrium is feasible between the grasp and an external wrench perturbation (e.g., due to gravity or inertia). Let $\mathcal{O}_i \in \mathbf{x}_t$ be an object contacted by the gripper when executing action \mathbf{u}_t . Then we measure grasp success with a binary-valued metric $S(\mathbf{x}_t, \mathbf{u}_t) \in \{0, 1\}$ that measures whether or not:

- The gripper geometry in the pose specified by \mathbf{u}_t is collision-free.
- The gripper contacts exactly one object \mathcal{O}_i when executing the grasp parameterized by \mathbf{u}_t .
- The grasp can resist a random disturbing force and torque (wrench) $\mathbf{w}_t = \mathbf{w}_g + \epsilon_w$ on the grasped object with over 50% probability, where \mathbf{w}_g is the fixed wrench due to gravity and ϵ_w is a random wrench sampled from a zero-mean Gaussian $\mathcal{N}(\mathbf{0}, \sigma_w^2 \mathbf{I})$.

Given an object consisting of a geometry \mathcal{M} in pose \mathbf{T}_o , the gripper g (geometry and physical parameters such as friction) and grasp pose \mathbf{T}_g are used to determine the contacts \mathbf{c} , or set of points and normals between the fingers and object. This set of contacts is used to compute the set of wrenches Λ that the grasp can apply to the object under quasi-static physics and a point contact model. Specifically, the wrench space for grasp \mathbf{u} using a contact model with m basis wrenches is $\Lambda(\mathbf{u}) = \{\mathbf{w} \in \mathbb{R}^6 \mid \mathbf{w} = G(\mathbf{u})\alpha \text{ for some } \alpha \in \mathcal{F}(\mathbf{u})\}$, as defined in [40]. The grasp matrix $G(\mathbf{u}) \in \mathbb{R}^{6 \times m}$ is a set of basis wrenches in the object coordinate frame specifying the set of wrenches that the grasp can apply through contact via active (e.g. joint torques) and passive (e.g. inertia) means. The wrench constraint set $\mathcal{F}(\mathbf{u}) \subseteq \mathbb{R}^m$ limits contact wrench magnitudes based on the capabilities of the gripper [49]. Finally, the grasp wrench space is used to measure grasp reward based on wrench resistance, or the ability of the grasp to resist a perturbation wrench \mathbf{w} (e.g. due to gravity) as defined in [40]. The grasp reward $R = 1$ if the probability of wrench resistance is greater than a threshold over M samples from the stochastic model.

Data Collection Policy

The dataset collection policy $\tau(\mathbf{u}_t \mid \mathbf{x}_t, \mathbf{y}_t)$ samples a mixture of actions from the point cloud and from an algorithmic supervisor $\Omega(\mathbf{x})$ that guides data toward successful grasps. Grasp actions are sampled from the point cloud using the sampling techniques of [39] and [40] to model the set of actions that the policy will evaluate when tested on real point clouds. Since this distribution may contain a very small percentage of successful actions, we sample actions with high expected reward from an algorithmic supervisor that evaluates grasps using full state information [38]. The supervisor pre-computes grasps on a set of known 3D objects in a database (such as in Dex-Net 1.0 [43]) that are robust to different possible orientations of each object. Since the state of each object in the heap is not known ahead of time, the supervisor estimates the probability of success, or quality, for each grasp over a large range of possible object orientations using the Monte-Carlo grasp computation methods of Dex-Net 2.0 [39] and Dex-Net 3.0 [40]. Given a full state of the heap, the supervisor computes the set of collision-free grasps with quality above a threshold for each object and then samples a grasp uniformly at random from the candidate set. Formally, the supervisor-guided data collection policy is:

$$\tau(\mathbf{u}_t \mid \mathbf{x}_t, \mathbf{y}_t) = \begin{cases} \Omega(\mathbf{x}_t) & \text{with prob. } \epsilon \\ \text{Unif}(\mathcal{U}_g(\mathbf{y}_t)) & \text{otherwise} \end{cases}$$

where $\mathcal{U}_g(\mathbf{y})$ is the set of candidate actions sampled from the point cloud with equal numbers of suction and parallel-jaw grasps. We use $\epsilon = 1\%$ to favor actions sampled from the policy’s own action space.

Training Details

The Dex-Net 4.0 training dataset contains a large set of labeled actions for each point cloud in order to improve the computational efficiency of generating a single datapoint. Specifically, datapoints are generated using a one-timestep Monte-Carlo evaluation of reward for a large set of grasp actions on each unique object state. This leads to faster dataset collection and can eliminate the need for fine-tuning, which is prone to a phenomenon known as “catastrophic forgetting” that can lead to unpredictable failures of the grasping policy [32]. Every sampled state from $\xi(\mathbf{x})$ has five associated depth images in Dex-Net 4.0, representing 3D point clouds captured from randomized camera poses and intrinsic optical parameters. Each image sampled from $\xi(\mathbf{y} \mid \mathbf{x})$ has a set of labeled actions for each gripper with associated quality metrics. The intrinsic parameters for the virtual cameras are sampled around the nominal values of a Photoneo PhoXi S industrial depth sensor. Images are converted to 96×96 pixel training thumbnails translated to move the grasp center to the thumbnail center pixel and rotated to align the grasp approach direction or axis with the middle row of pixels for the suction and parallel-jaw grippers, respectively.

The GQ-CNN architectures are similar to those used in Dex-Net 2.0 [39] and Dex-Net 3.0 [40] with two primary changes. First, we remove local response normalization as experi-

ments suggest that it does not affect training performance. Second, we modify the sizes and pooling of the following layers : conv1_1 (16 9×9 filters, $1\times$ pooling), conv1_2 (16 5×5 filters, $2\times$ pooling), conv2_1 (16 5×5 filters, $1\times$ pooling), conv2_2 (16 5×5 filters, $2\times$ pooling), fc3 (128 output neurons), pc1 (16 output neurons), and fc4 (128 output neurons).

We train each GQ-CNN using stochastic gradient descent with momentum for 50 epochs using an 80-20 training-to-validation image-wise split of the Dex-Net 4.0 dataset. We use a learning rate of 0.01 with an exponential decay of 0.95 every 0.5 epochs, a momentum term of 0.9, and an ℓ_2 weight regularization of 0.0005. We initialize the weights of the model by sampling from a zero-mean Gaussian with variance $\frac{2}{n_i}$, where n_i is the number of inputs to the i -th network layer [19]. To augment the dataset during training, we reflect each image about its vertical and horizontal axes and rotate each image by 180° since these lead to equivalent grasps. Training took approximately 24 hours on a single NVIDIA Titan Xp GPU. The learned GQ-CNNs achieve 96% and 98% classification accuracy for the suction and parallel-jaw grippers, respectively, on the held-out validation set.

Implementation of Policies

We use the trained GQ-CNNs to plan grasps from point clouds on a physical robot with derivative-free optimization to search for the highest quality grasp across both grippers. The policy optimizes for the highest quality grasp from each gripper separately uses the Cross Entropy Method (CEM) [36, 39, 40, 57], and then selects the grasps with the highest estimated quality across the grippers. To avoid grasping the bin, grasps are constrained to the foreground by subtracting out the background pixels of the bin using a reference depth image of an empty bin. Grasps are also constrained to be collision-free with the bin to avoid damage to the robot. Given the constraints, CEM samples a set of initial candidate grasps uniformly at random from a point cloud and then iteratively re-samples grasps from a Gaussian Mixture Model fit to the grasps with the highest estimated quality. For the suction cup gripper, initial candidate grasps are sampled by selecting a 3D point and choosing an approach direction along the inward-facing surface normal. For the parallel-jaw gripper, initial candidate grasps are sampled by finding antipodal point pairs using friction cone analysis.

3.4 Physical Experiments

Overview

We execute over 2,500 grasp attempts on a physical robot system with a parallel-jaw and suction cup gripper to characterize the reliability of the Dex-Net 4.0 policy on a bin picking task with 50 novel test objects. The experiments aim to evaluate (1) the reliability and range of the Dex-Net 4.0 policy compared to a set of baselines, (2) the effect of training dataset

diversity, neural network architecture, and learning from real data, and (3) the failure modes of the Dex-Net 4.0 policy.

To analyze performance, we select a dataset of 50 objects with diverse shapes, sizes, and material properties. The dataset includes retail products, groceries, tools, office supplies, toys, and 3D printed industrial parts. We separate objects into two difficulty levels with 25 objects each (illustrated in Fig. 3.2):

- *Level 1*: Prismatic and circular solids (e.g. rectangular prisms, spheres, cylinders).
- *Level 2*: Common household objects including examples with clear plastic covers (“blisterpack”), varied geometry, and masses up to 500g (the payload of the ABB YuMi).

A complete listing of objects with purchase links can be found at <http://bit.ly/2wGPsvZ>.

For each trial, we place all objects in the bin and allow the robot to iteratively execute the next grasp planned by the policy given a depth image from an overhead 3D camera until either no objects remain or a maximum number of attempts is reached.

Experimental Procedure

Each experiment consists of five independent trials in which the bin is filled with a random configuration of one or more objects and the robot attempts to pick each object from the bin and transport it to a receptacle. Before each experiment, the camera pose is registered to the robot using a chessboard. In each trial, the operator sets the full dataset of objects in the bin by shaking the objects in a box, placing the box upside down in the bin, and mixing the bin manually to ensure that objects rested below the rim of the bin. Then the robot iteratively attempts to pick objects from the bin. On each attempt, the grasping policy receives as input a point cloud of the objects in the bin and returns a grasp action for exactly one of the grippers, consisting of a pose for the gripper relative to the base of the robot. Then the ABB RAPID linear motion planner and controller are used to move to the target pose, establish contact with the object, and drop the object in the receptacle. The operator labels the grasp as successful if the robot lifts and transports the object to the receptacle on the side of the workspace. The operator also labels the identity of each grasped object. A trial is considered complete after either all objects are removed, 75 total attempts, or 10 consecutive failures. All experiments ran on a Desktop running Ubuntu 16.04 with a 3.4 GHz Intel Core i7-6700 Quad-Core CPU and an NVIDIA Titan Xp GPU.

Comparison with Baseline Policies

We evaluate the Dex-Net 4.0 ambidextrous policy against three baselines in five independent trials:

1. **Heuristic (Suction)**. Ranks planar grasps based on the inverse distance to the centroid of an object [48], where the object centroid is estimated as the mean pixel of

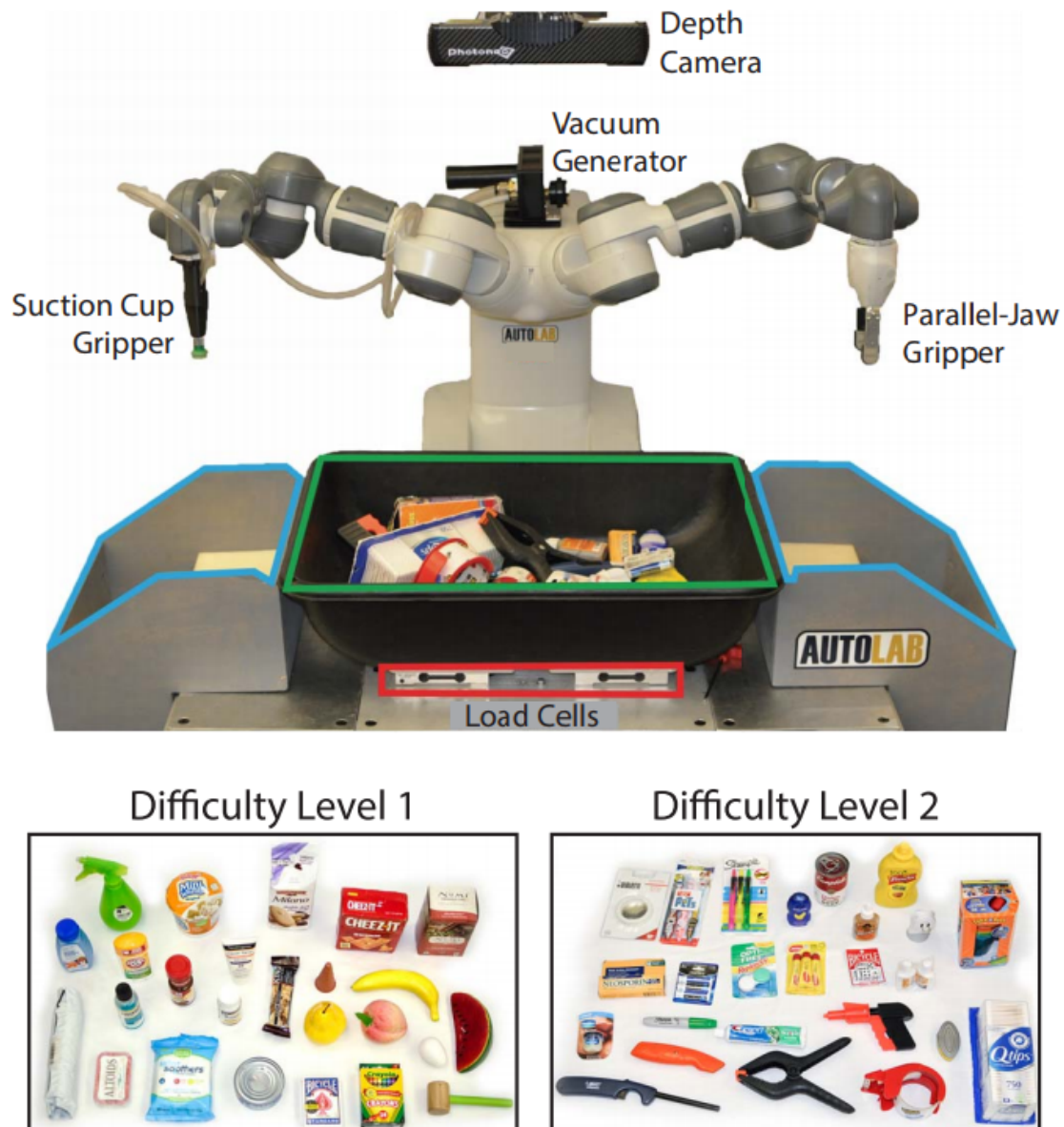


Figure 3.2: Physical benchmark for evaluating universal picking policies. (Top) The robot plans a grasp to iteratively transport each object from the picking bin (green) to a receptacle (blue) using either a suction cup or parallel-jaw gripper. Grasp planning is based on 3D point clouds from an overhead Photoneo PhoXi S industrial depth camera. (Bottom) Performance is evaluated on two datasets of novel test objects not used in training. (Left) The Level 1 objects consist of prismatic and circular solids (e.g. boxes, cylinders) spanning groceries, toys, and medicine. (Right) The Level 2 objects are more challenging, including common objects with clear plastic and varied geometry such as products with cardboard “blisterpack” packaging.

an object instance segmask from a tuned Euclidean Clustering segmentation algorithm from the Point Cloud Library (PCL) [59]. Planarity is determined by evaluating the

mean squared error (MSE) of all 3D points within a sphere of radius $10mm$ (based on the suction cup size) to the best-fit plane for the points. Grasps are considered planar if either (a) the MSE is less than an absolute threshold or (b) the MSE is within the top 5% of all candidate grasps. The hyperparameters are hand-coded to optimize performance on the physical robot.

2. **Heuristic (Composite).** Ranks grasps planned with the suction heuristic above and a parallel-jaw heuristic based on antipodality. The parallel-jaw heuristic ranks antipodal grasps based on the inverse distance to the estimated centroid of an object, determining antipodality based on estimated point cloud surface normals. The height of the gripper above the bin surface is a constant offset from the highest point within the region of the grasp. The grasp closest to the estimated object centroid across both grippers is selected for execution.
3. **Dex-Net 2&3 Composite.** Ranks grasps based on the estimated quality from separate GQ-CNNs trained to estimate the quality of parallel-jaw and suction cup grasps in clutter. The GQ-CNNs are trained by fine-tuning the Dex-Net 2.0 and 3.0 base networks on simulated heaps with imitation learning from an algorithmic supervisor [38].

Fig. 3.3 shows the performance on the two object datasets. Dex-Net 4.0 achieves the highest success rate across all object datasets with a reliability of 97% and 95% on the Level 1 and Level 2 objects, respectively. The policy uses the suction gripper on 82% of grasps. The baseline method has a reliability of 93% and 80%, respectively. Analysis of the number of objects picked versus the number of attempts suggests that the baseline methods take longer to clear the last few objects from the bin, sometimes altogether failing to clear several of the objects.

We detail additional metrics for each policy in Table 3.1, including the reliability and MPPH of the learned quality functions. The Dex-Net 4.0 policy has the highest reliability on both the Level 1 and Level 2 objects. The policy has a slightly lower MPPH than the suction heuristic on the Level 1 objects since the heuristic can be evaluated more rapidly than the GQ-CNN.

To further quantify the range of the Dex-Net 4.0 ambidextrous policy, we measure the performance of grasping each of the 50 objects from Level 1 and Level 2 in isolation in the bin for 5 attempts each. Dex-Net 4.0 achieved 98% reliability versus 52% and 94% for the Dex-Net 2.0 and 3.0 policies, respectively.

Performance with Large Heaps

To investigate whether heap size affects performance, we benchmark the policy on a dataset of 50 test objects combining all objects from both the Level 1 and Level 2 datasets. Fig. 3.3 displays the results for five independent trials with each policy. Dex-Net 4.0 has the highest reliability at 90%. In comparison, the performance of the heuristics is relatively unchanged

Policy	Level 1					Level 2				
	Reliability (%)	MPPH	AP (%)	# Attempts	# Failures	Reliability (%)	MPPH	AP (%)	# Attempts	# Failures
Heuristic (Suction)	93	331	95	135	10	80	304	87	156	31
Heuristic (Composite)	91	281	93	139	14	76	238	83	168	43
Dex-Net 2&3 Composite	91	306	93	135	10	76	255	64	168	43
Dex-Net 4.0	97	309	100	129	4	95	312	99	131	6

Table 3.1: Detailed performance analysis of the Dex-Net 4.0 and baseline policies on the bin picking benchmark for five trials each on the Level 1 and Level 2 datasets of 25 novel objects each. We report the reliability, mean picks per hour (MPPH), average precision (AP), total number of grasps attempts (minimum of 125), and total number of failures.

with success rates near 80%. Some are due to failed attempts to lift objects from underneath others.

Effects of Training Dataset Diversity

We quantify the importance of dataset diversity by training the GQ-CNNs on three alternative synthetic training datasets:

1. **Fewer Unique Objects.** 100 unique 3D objects in 2,500 unique heaps.
2. **Very Few Unique Heaps.** 1,664 unique 3D objects in 100 unique heaps.
3. **Fewer Unique Heaps.** 1,664 unique 3D objects in 500 unique heaps.

Fig. 3.3 displays the performance on the Level 1 and Level 2 objects. The policies have reduced reliability and appear to be particularly sensitive to the number of unique heaps used in training.

Varying the Neural Network Architecture

We study whether or not changes to the neural network architecture affect the performance of the resulting policy by training on the Dex-Net 4.0 dataset using the “Improved GQ-CNN” architecture [26]. As seen in Fig. 3.3, the architecture has comparable performance to the standard GQ-CNN architecture.

Training on Physical Grasp Outcomes

We also explore whether performance can be improved by training on labeled grasp attempts on a physical system. We use a dataset of over 13,000 labeled grasp attempts collected over six months of experiments and demonstrations of the system. Approximately 5,000 data-points were labeled by human operators and the remaining 8,000 were labeled automatically using weight differences measured by load cells.

We train ten variants of the Dex-Net 4.0 ambidextrous grasping policy on a dataset of 13,000 real grasp attempts: training from scratch and fine-tuning either all fully connected

(FC) layers or only the last FC layer (fc4) on varying ratios of real to simulated data: 1-to-0, 1-to-1, 1-to-10, and 1-to-100. Each variant was evaluated on the adversarial Level 3 objects on the physical robot, and the highest performing policy was the Dex-Net 4.0 policy with the last FC layer fine tuned on the 1-to-10 combined real and synthetic dataset. The best-performing empirically-trained policy has comparable reliability to the original Dex-Net 4.0 policy on the physical benchmark, as shown in Fig. 3.3, and does not lead to significant performance increases.

Adversarial Objects

To probe the boundaries of the range of the Dex-Net 4.0 policy, we evaluate its performance on a third object dataset which contains 25 novel objects with few accessible and perceptible grasps due to adversarial geometry, transparency, specularly, and deformability. The results are illustrated in Fig. 3.4. Dex-Net 4.0 is still the highest performing policy, but the reliability is reduced to 63%.

Failures of the Dex-Net 4.0 policy often occur several times in sequence. To characterize these sequential failures, we explore a first-order memory-based policy to encourage exploration upon repeated failures, a technique that has been used to improve performance in the Amazon Robotics Challenge [74]. To avoid repeated grasp failures, this memory system that associates regions of the point cloud with past failures. A grasp is considered a failure if the weight reading from the load cells changes less than $5g$ after a grasp attempt. When a failure occurs, the point cloud is segmented using the PCL library [59] and the segment corresponding to the grasped object is associated with a region in a grayscale image. The segmented image patch is featurized using the VGG-16 network and stored in a failure database corresponding to the gripper. On the next grasp attempt, the point cloud segments are matched to the failure database using the VGG-16 featurization. If a match is found, the region in the current image is marked as forbidden to the grasp sampler for the Dex-Net 4.0 policy. Furthermore, if more than three consecutive failures occur, then the memory system rejects the planned grasp and uses a pushing policy [9] to perturb the objects in the bin. The addition of memory increased the reliability to 80% at 242 MPPH.

3.5 Discussion

Experiments suggest that ambidextrous policies trained on Dex-Net 4.0 achieve high reliability on novel objects on a physical robot, with reliability over 95% on heaps of 25 novel objects at over 300 mean picks per hour. Dex-Net 4.0 outperforms hand-coded baselines similar to those used in applications and an ambidextrous policy based on previous versions of Dex-Net that use separate reward functions for each gripper. This suggests that learning with consistent reward functions across grippers can lead to increased reliability on a physical robot.

Experiments also suggest that performance is sensitive to several factors. Heaps containing more objects lead to decreased reliability as the policy attempts to lift objects that are occluded by others in the heap. Performance also depends on the diversity of the training dataset, with more diverse datasets leading to higher performance on a physical robot. Finally, performance varies based on the test objects, with more complex geometries and material properties leading to reduced reliability. Use of a memory system to avoid repeated failures can help compensate for this reduction, increasing reliability on adversarial objects from 63% to 80%. Interestingly, experiments suggest that reliability is not significantly affected by changes to the neural network architecture or fine-tuning on up to 13k real datapoints.

Benefits of Ambidextrous Grasping

The experimental results highlight the advantage of using a set of two or more heterogeneous grippers. Although a policy with only a single suction cup can achieve high reliability on the Level 1 prismatic and circular objects, performance drops to 80% on the Level 2 objects with more complex geometry. In comparison, the ambidextrous grasping policy utilizes the parallel-jaws on 20% of grasp attempts to achieve 95% reliability on the Level 2 objects. Furthermore, a consistent reward appears to be important for learning an ambidextrous policy to reliably decide between grippers. However, this study only considers a single combination of grippers. Future research could study applications to new grippers such as two-finger underactuated hands or multi-suction-cup arrays. Future work could also consider extensions of ambidextrous grasping such as simultaneous grasping with multiple arms or planning grasps for three or more grippers.

Physics-Based Reward Design

The results of this paper also suggest that analytic quasi-static grasp quality metrics [34, 53] with domain randomization can be used as a computationally-efficient reward function for learning ambidextrous grasping policies that are robust to sensor noise and imprecision. This stands in contrast to past research [3, 4] that has criticized quasi-static metrics for making strong assumptions and considering only a necessary, not sufficient, condition for dynamic grasp stability. Experiments suggest that the Dex-Net 4.0 policy generalizes to objects with deformable surfaces, moving parts, and transparencies that do not satisfy the assumptions of the analytic metrics. This may be because grasps with high analytic quality over a diverse range of 3D objects tend to correlate with grasp affordances: geometric features of objects that facilitate grasping such as handles or flat suctionable surfaces. Further studies may be necessary to understand why grasps are often dynamically stable in practice. One hypothesis is that material compliance in the fingertips acts as a passive stability controller. Future research might also investigate whether or not this result generalizes to additional grippers such as multifingered [50] or soft hands [11].

Bias-Variance Tradeoff in Dataset Collection

Experiments suggest that a policy fine-tuned on 13k examples collected from physical experiments does not significantly improve the Dex-Net 4.0 ambidextrous grasping policy trained on only synthetic data. This may appear counterintuitive since the model used to generate synthetic training data cannot possibly model the exact behavior of the real-world system and therefore may induce bias [17]. This may relate to the bias-variance tradeoff in machine learning [15]. Although the tradeoff is typically analyzed in terms of the function class, the results of this paper suggest that the training data distribution is also relevant. Using a biased model for rapid data collection may improve the scale and consistency of training datasets, leading to better performance on a physical system in comparison to methods based on smaller training datasets with high rates of mislabeled examples. Future research could consider novel methods for learning with a combination of synthetic and real data, such as using analytic models to guide empirical data collection.

Application to Different Sensors and Grippers

The Dex-Net 4.0 method for training Universal Picking policies could be applied to other objects, cameras, and grippers by implementing a new dataset generation distribution and training a new GQ-CNN on samples from this distribution. For example, objects could be virtually placed in structured configurations, such as packed in boxes or placed on shelves, and camera intrinsic parameters could be set to model a different sensor. However, the experiments in this paper are limited in scope. This study only evaluates performance on heaps of 50 unique, randomly-arranged objects, which do not represent all possible object geometries. Furthermore, the hardware benchmark uses only one industrial high-resolution depth camera positioned directly overhead. The experiments only test a single parallel-jaw and vacuum suction cup gripper. Finally, the constant-time assumption that relates MPPH maximization to supervised learning (Section 3.3) may not be applicable to all robot picking systems. For example, there may be a time-cost for switching grippers due time spent physically mounting each tool. Future studies could evaluate performance in new applications with variations in objects, cameras, grippers, and robots.

Opportunities for Future Research

The most common failure modes of the policy are (1) attempting to grasp objects that are occluded due to overlap in the heap and (2) repeated failures on objects with adversarial geometry and material properties. A subset of objects that cannot yet be reliably grasped with Dex-Net 4.0 is pictured in Fig. 3.4. One category includes objects imperceptible to a depth camera, such as those with transparent or specular surfaces. Another category is characterized by structured surface variations such as parallel lines or buttons on a remote, which can trigger false positives in the suction network. Other classes include porous objects and objects with loose packaging.

Some failure modes could be addressed by increasing the diversity of objects in the training dataset or improving the dataset generation model. Rendering synthetic color images using domain randomization [65] could enable the system to grasp transparent, specular, or highly textured objects. Models of deformation and porosity could be used to reduce suction failures due to incorrect assumptions of the Dex-Net 4.0 model. The reward model could also be extended to compute the wrench set from all contacts between objects instead of only considering the grippers and gravity, which could reduce failures due to object overlap.

Other extensions could significantly increase the reliability and range. The observed performance increase on the Level 3 objects using a first-order memory system suggests that reinforcement learning could be used to reduce repeated failures. Furthermore, training on larger datasets of empirically-collected data could reduce the simulation-to-reality gap. Another way to increase rate is to use feedback policies that actively regrasp dropped objects based on visual servoing [36, 47, 69], force sensing [21, 56, 66], or tactile sensing [5, 22, 23, 46].

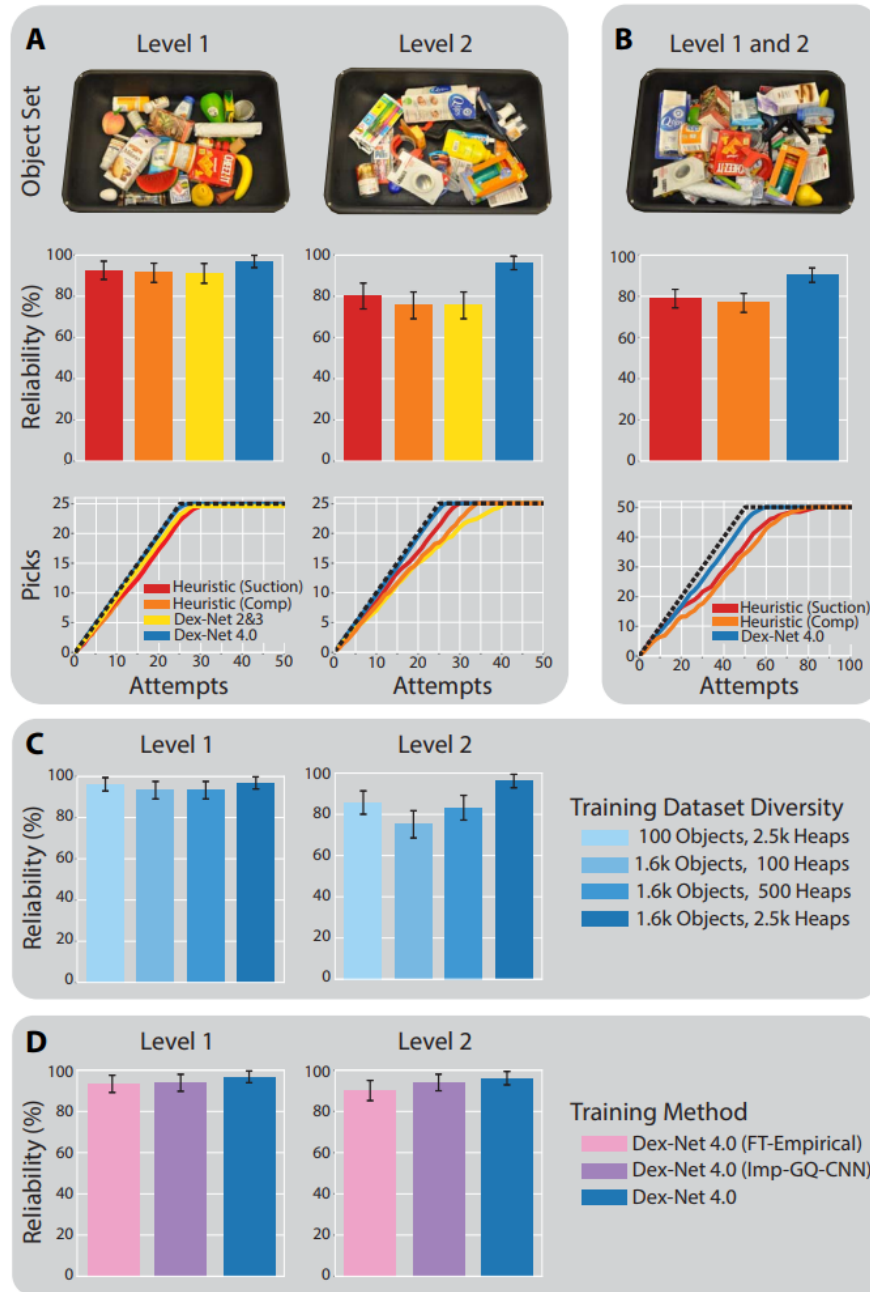


Figure 3.3: Performance of the Dex-Net 4.0 ambidextrous policy on the bin picking benchmark. Error bars show the 95% confidence interval on reliability using the standard error of the mean. **(A)** Comparison with three baseline methods on the Level 1 and Level 2 objects on heaps of 25 novel objects: (1) a hand-coded heuristic for the suction cup, (2) a hand-coded heuristic for selecting between the suction cup and parallel-jaw gripper, and (3) an ambidextrous policy fine-tuned on simulated heaps from the Dex-Net 2.0 and 3.0 base GQ-CNNs and reward metrics. For reference, the best possible performance (succeeding on every pick until the bin is cleared) is illustrated with a black-dotted line. **(B)** Performance with large heaps of 50 novel objects. **(C)** Ablation study measuring the effect of training on less diverse datasets with either fewer unique heaps or fewer unique 3D object models. **(D)** Performance of two training alternatives: the Improved GQ-CNN architecture [26] and fine-tuning on 13k real datapoints.

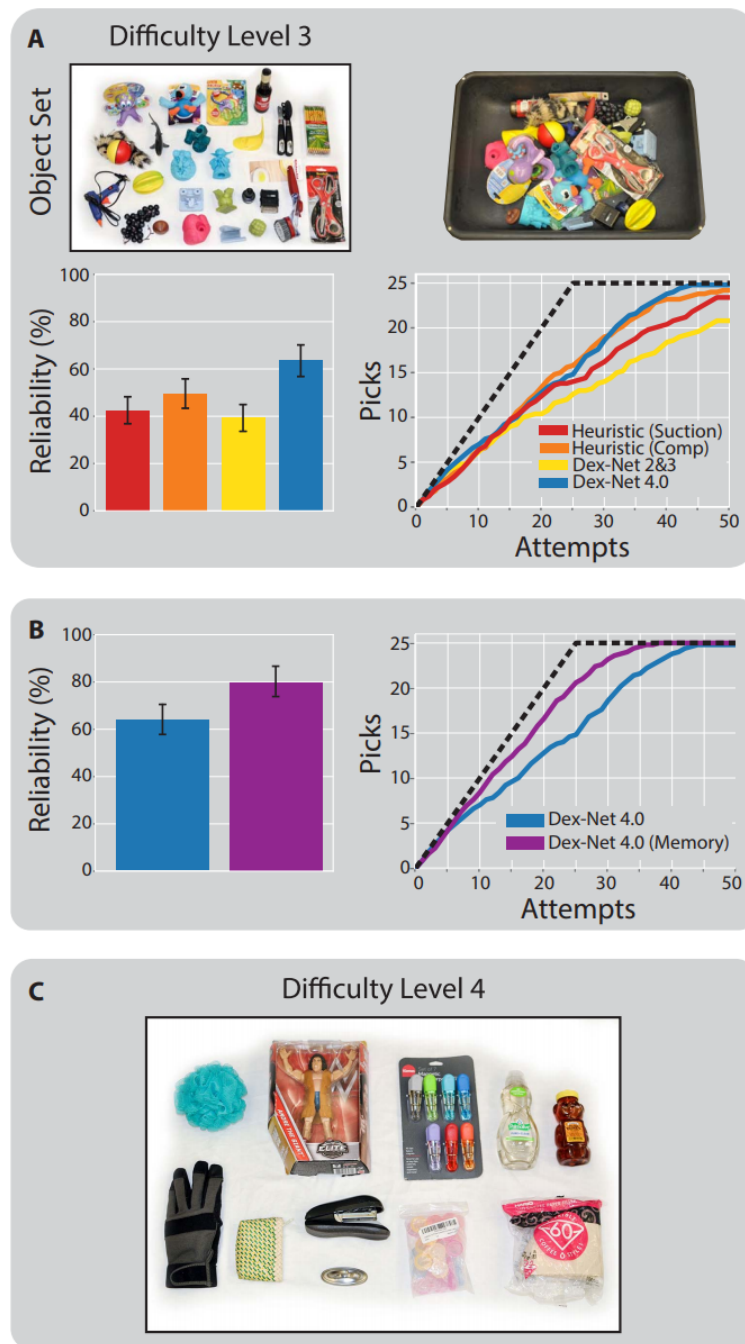


Figure 3.4: Failure modes of the Dex-Net 4.0 policy. Error bars show the 95% confidence interval on reliability using the standard error of the mean. (A) Performance on Level 3: a dataset of 25 novel objects with adversarial geometry and material properties. (B) Evaluation of a first-order memory-based policy for universal picking that masks regions of the point cloud with an instance recognition system to avoid repeated failures. (C) Pathological objects which cannot be grasped with Dex-Net 4.0 due to reflectance properties such as transparency, which affect depth sensing, and material properties such as porosity and deformability (e.g. loose packaging), which affect the ability to form a vacuum seal on the object surface.

Chapter 4

Conclusion and Future Work

The body of this report focused on my contributions to two projects – Dex-Net 3.0, which created a new contact model for suction-cup grippers and used it to generate simulated training data for a point-cloud-based grasping policy, and Dex-Net 4.0, which unified this model with a point contact model for parallel-jaw grippers and used the unified model to generate training data for an ambidextrous grasping policy that can decide between two grippers at runtime. The framework we have developed is generalizable across virtually any gripper that uses fingers or circular suction cups, which allows a system running Dex-Net to apply to a wider variety of objects than prior systems. While the final ambidextrous policy achieved state-of-the-art performance, there are still many avenues for both improving the performance of such policies and utilizing them in more sophisticated robotic manipulation systems.

First, the contact models we use in our work are somewhat limited by their simple, quasi-static nature. In the future, we hope to investigate other methods of modeling contact. For finger-based grippers, our colleagues have already begun investigating area-based contact models that more accurately capture the set of forces and torques that fingers can apply. For suction-based grippers, we would like to incorporate a simulated measure of vacuum pressure and air flow that provides a more informative and realistic measure of the forces applied by a cup dynamically. In addition, we would like to investigate methods for efficiently modeling the dynamics of grasping in simulation. These dynamics are particularly important for deformable objects such as plastic bags, water bottles that are half-filled, or books, which violate many of the assumptions of our models and are often dropped during motion between the grasp point and the placement location due to these unmodeled effects. By incorporating more accurate physical models for the interactions between robot grippers and these types of objects, we should be able to generate better training data and therefore train a more accurate grasping policy.

Additionally, the Dex-Net method doesn't currently plan for task affordances when grasping. For example, if the robot needed to use a hammer after grasping it, it would need to grasp it near the handle and should probably avoid grasping it with a flexible suction cup. We would like to explore methods for building task affordance information into our Dex-Net

simulations so that we can train robots to perform more complex tasks in simulation. This could tie in to my prior work on semantic mesh segmentation and grasp transfer between similar objects [45], and could serve as a building block for additional robotic capabilities.

As discussed in the introduction, the Dex-Net grasp planner has already been used in other research applications, including mechanical search [8], a framework for searching for and extracting a target item from a diverse heap or bin containing many other items. Currently, our research systems like this one simply drop objects that they've grasped over a hard-coded target area. In future work, we'd like to extend our grasp-planning methods to plan placements, which would allow us to create complete robotic applications that could be useful in e-commerce logistics and flexible manufacturing settings. In flexible manufacturing, where precise part placements are required and CAD data is plentiful, we plan to continue working to develop methods for precise in-hand 6-DoF pose estimation and efficient part reorientation. For more unstructured environments such as warehouses, we plan to research methods for identifying rough object dimensions from sensor data and planning space-efficient packing in rectangular containers such as cardboard boxes.

By building on the potential created by our work on the Dex-Net project and continuing to explore techniques for endowing robots with new skills via simulation, we hope to bring the grand vision of robotics even closer to reality. In the remainder of my PhD, I hope to continue building up fundamental knowledge in the areas of pose estimation, part placement, and robotic bin packing to continue advancing towards this goal.

Bibliography

- [1] A. Ali, M. Hosseini, and B. Sahari, “A review of constitutive models for rubber-like materials,” *American Journal of Engineering and Applied Sciences*, vol. 3, no. 1, pp. 232–239, 2010.
- [2] B. Bahr, Y. Li, and M. Najafi, “Design and suction cup analysis of a wall climbing robot,” *Computers & electrical engineering*, vol. 22, no. 3, pp. 193–209, 1996.
- [3] R. Balasubramanian, L. Xu, P. D. Brook, J. R. Smith, and Y. Matsuoka, “Physical human interactive guidance: Identifying grasping principles from human-planned grasps,” *IEEE Trans. Robotics*, vol. 28, no. 4, pp. 899–910, 2012.
- [4] J. Bohg, A. Morales, T. Asfour, and D. Kragic, “Data-driven grasp synthesis: A survey,” *IEEE Trans. Robotics*, vol. 30, no. 2, pp. 289–309, 2014.
- [5] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine, “More than a feeling: Learning to grasp and regrasp using vision and touch,” in *Proc. IEEE Int. Conf. Robotics and Automation*, 2018, pp. 3300–3307.
- [6] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman, “Analysis and observations from the first amazon picking challenge,” *IEEE Trans. Automation Science and Engineering*, 2016.
- [7] E. Coumans *et al.*, “Bullet physics library,” *Open source: bulletphysics.org*, vol. 15, no. 49, p. 5, 2013.
- [8] M. Danielczuk, A. Kurenkov, A. Balakrishna, M. Matl, D. Wang, R. Martín-Martín, A. Garg, S. Savarese, and K. Goldberg, “Mechanical search: Multi-step retrieval of a target object occluded by clutter,” *arXiv preprint arXiv:1903.01588*, 2019.
- [9] M. Danielczuk, J. Mahler, C. Correa, and K. Goldberg, “Linear push policies to increase grasp access for robot bin picking,” in *Proc. IEEE Conf. on Automation Science and Engineering*, 2018.

- [10] M. Danielczuk, M. Matl, S. Gupta, A. Li, A. Lee, J. Mahler, and K. Goldberg, “Segmenting unknown 3d objects from real depth images using mask r-cnn trained on synthetic point clouds,” *arXiv preprint arXiv:1809.05825*, 2018.
- [11] R. Deimel and O. Brock, “A novel type of compliant and underactuated robotic hand for dexterous grasping,” *Int. Journal of Robotics Research*, vol. 35, no. 1-3, pp. 161–185, 2016.
- [12] Y. Domae, H. Okuda, Y. Taguchi, K. Sumi, and T. Hirai, “Fast graspability evaluation on single depth maps for bin picking with general grippers,” in *Proc. IEEE Int. Conf. Robotics and Automation*. IEEE, 2014, pp. 1997–2004.
- [13] C. Eppner, S. Höfer, R. Jonschkowski, R. M. Martin, A. Sieverling, V. Wall, and O. Brock, “Lessons from the amazon picking challenge: Four aspects of building robotic systems.” in *Proc. Robotics: Science and Systems*, 2016.
- [14] C. Ferrari and J. Canny, “Planning optimal grasps,” in *Proc. IEEE Int. Conf. Robotics and Automation*, 1992, pp. 2290–2295.
- [15] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer, 2001, vol. 1.
- [16] K. Goldberg, B. V. Mirtich, Y. Zhuang, J. Craig, B. R. Carlisle, and J. Canny, “Part pose statistics: Estimators and experiments,” *IEEE Trans. Robotics and Automation*, vol. 15, no. 5, pp. 849–857, 1999.
- [17] 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.
- [18] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [19] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proc. IEEE Int. Conf. on Computer Vision*, 2015, pp. 1026–1034.
- [20] C. Hernandez, M. Bharatheesha, W. Ko, H. Gaiser, J. Tan, K. van Deurzen, M. de Vries, B. Van Mil, J. van Egmond, R. Burger, *et al.*, “Team delft’s robot winner of the amazon picking challenge 2016,” *arXiv preprint arXiv:1610.05514*, 2016.
- [21] F. Hogan, E. Grau, and A. Rodriguez, “Reactive planar manipulation with convex hybrid mpc,” in *Proc. IEEE Int. Conf. Robotics and Automation*, 2018, pp. 247–253.
- [22] F. R. Hogan, M. Bauza, O. Canal, E. Donlon, and A. Rodriguez, “Tactile regrasp: Grasp adjustments via simulated tactile transformations,” *arXiv preprint arXiv:1803.01940*, 2018.

- [23] R. D. Howe, “Tactile sensing and control of robotic manipulation,” *Advanced Robotics*, vol. 8, no. 3, pp. 245–261, 1993.
- [24] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, “Spatial transformer networks,” in *Proc. Advances in Neural Information Processing Systems*, 2015, pp. 2017–2025.
- [25] S. James, A. J. Davison, and E. Johns, “Transferring end-to-end visuomotor control from simulation to real world for a multi-stage task,” in *Proc. Conference on Robot Learning*, 2017, pp. 334–343.
- [26] M. Jaśkowski, J. Świątkowski, M. Zajac, M. Klimek, J. Potiuk, P. Rybicki, P. Polatowski, P. Walczyk, K. Nowicki, and M. Cygan, “Improved gq-cnn: Deep learning model for planning robust grasps,” *arXiv preprint arXiv:1802.05992*, 2018.
- [27] 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*. IEEE, 2016, pp. 4461–4468.
- [28] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, “Planning and acting in partially observable stochastic domains,” *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.
- [29] I. Kao, K. Lynch, and J. W. Burdick, “Contact modeling and manipulation,” in *Springer Handbook of Robotics*. Springer, 2008, pp. 647–669.
- [30] D. Kappler, J. Bohg, and S. Schaal, “Leveraging big data for grasp planning,” in *Proc. IEEE Int. Conf. Robotics and Automation*, 2015.
- [31] A. Kasper, Z. Xue, and R. Dillmann, “The kit object models database: An object model database for object recognition, localization and manipulation in service robotics,” *The International Journal of Robotics Research*, vol. 31, no. 8, pp. 927–934, 2012.
- [32] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, *et al.*, “Overcoming catastrophic forgetting in neural networks,” *Proceedings of the National Academy of Sciences*, vol. 13, pp. 3521–3126, 2017.
- [33] R. Kolluru, K. P. Valavanis, and T. M. Hebert, “Modeling, analysis, and performance evaluation of a robotic gripper system for limp material handling,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 28, no. 3, pp. 480–486, 1998.
- [34] R. Krug, Y. Bekiroglu, and M. A. Roa, “Grasp quality evaluation done right: How assumed contact force bounds affect wrench-based quality metrics,” in *Proc. IEEE Int. Conf. Robotics and Automation*. IEEE, 2017, pp. 1595–1600.

- [35] I. Lenz, H. Lee, and A. Saxena, “Deep learning for detecting robotic grasps,” *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.
- [36] 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*, vol. 37, no. 4-5, pp. 421–436, 2018.
- [37] 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.
- [38] J. Mahler and K. Goldberg, “Learning deep policies for robot bin picking by simulating robust grasping sequences,” in *Proc. Conference on Robot Learning*, 2017, pp. 515–524.
- [39] 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*, 2017.
- [40] J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, and K. Goldberg, “Dex-net 3.0: Computing robust vacuum suction grasp targets in point clouds using a new analytic model and deep learning,” in *Proc. IEEE Int. Conf. Robotics and Automation*, 2018.
- [41] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, “Learning ambidextrous robot grasping policies,” *Science Robotics*, vol. 4, no. 26, p. eaau4984, 2019.
- [42] 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*. IEEE, 2016.
- [43] —, “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*. IEEE, 2016, pp. 1957–1964.
- [44] G. Mantriota, “Theoretical model of the grasp with vacuum gripper,” *Mechanism and machine theory*, vol. 42, no. 1, pp. 2–17, 2007.
- [45] M. Matl, J. Mahler, and K. Goldberg, “An algorithm for transferring parallel-jaw grasps between 3d mesh subsegments,” in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*. IEEE, 2017, pp. 756–763.
- [46] A. Molchanov, O. Kroemer, Z. Su, and G. S. Sukhatme, “Contact localization on grasped objects using tactile sensing,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, 2016, pp. 216–222.

- [47] 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*, 2018.
- [48] D. Morrison, A. W. Tow, M. McTaggart, R. Smith, N. Kelly-Boxall, S. Wade-McCue, J. Erskine, R. Grinover, A. Gurman, T. Hunn, *et al.*, “Cartman: The low-cost cartesian manipulator that won the amazon robotics challenge,” in *Proc. IEEE Int. Conf. Robotics and Automation*. IEEE, 2018, pp. 7757–7764.
- [49] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- [50] OpenAI, “Learning dextrous in-hand manipulation,” *arXiv preprint arXiv:1808.00177*, 2018.
- [51] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, “Sim-to-real transfer of robotic control with dynamics randomization,” in *Proc. IEEE Int. Conf. Robotics and Automation*. IEEE, 2018, pp. 1–8.
- [52] 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*, 2016.
- [53] F. T. Pokorny and D. Kragic, “Classical grasp quality evaluation: New algorithms and theory,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, 2013, pp. 3493–3500.
- [54] D. Prattichizzo and J. C. Trinkle, “Grasping,” in *Springer handbook of robotics*. Springer, 2008, pp. 671–700.
- [55] X. Provot *et al.*, “Deformation constraints in a mass-spring model to describe rigid cloth behaviour,” in *Graphics interface*. Canadian Information Processing Society, 1995, pp. 147–147.
- [56] L. Righetti, M. Kalakrishnan, P. Pastor, J. Binney, J. Kelly, R. C. Voorhies, G. S. Sukhatme, and S. Schaal, “An autonomous manipulation system based on force control and optimization,” *Autonomous Robots*, vol. 36, no. 1-2, pp. 11–30, 2014.
- [57] R. Y. Rubinstein, A. Ridder, and R. Vaisman, *Fast sequential Monte Carlo methods for counting and optimization*. John Wiley & Sons, 2013.
- [58] A. A. Rusu, M. Večerík, T. Rothörl, N. Heess, R. Pascanu, and R. Hadsell, “Sim-to-real robot learning from pixels with progressive nets,” in *Proc. Conference on Robot Learning*, 2017, pp. 262–270.
- [59] R. B. Rusu and S. Cousins, “3d is here: Point cloud library (pcl),” in *Proc. IEEE Int. Conf. Robotics and Automation*. IEEE, 2011, pp. 1–4.

- [60] F. Sadeghi and S. Levine, “Cad2rl: Real single-image flight without a single real image,” in *Proc. Robotics: Science and Systems*, 2017.
- [61] A. Saxena, J. Driemeyer, and A. Y. Ng, “Robotic grasping of novel objects using vision,” *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.
- [62] H. S. Stuart, M. Bagheri, S. Wang, H. Barnard, A. L. Sheng, M. Jenkins, and M. R. Cutkosky, “Suction helps in a pinch: Improving underwater manipulation with gentle suction flow,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, 2015, pp. 2279–2284.
- [63] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, “Grasp pose detection in point clouds,” *Int. Journal of Robotics Research*, vol. 36, no. 13-14, pp. 1455–1473, 2017.
- [64] N. Tian, M. Matl, J. Mahler, Y. X. Zhou, S. Staszak, C. Correa, S. Zheng, Q. Li, R. Zhang, and K. Goldberg, “A cloud robot system using the dexterity network and berkeley robotics and automation as a service (brass),” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1615–1622.
- [65] 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*. IEEE, 2017, pp. 23–30.
- [66] M. R. Tremblay and M. R. Cutkosky, “Estimating friction using incipient slip sensing during a manipulation task,” in *Proc. IEEE Int. Conf. Robotics and Automation*. IEEE, 1993, pp. 429–434.
- [67] N. C. Tsourveloudis, R. Kolluru, K. P. Valavanis, and D. Gracanin, “Suction control of a robotic gripper: A neuro-fuzzy approach,” *Journal of Intelligent & Robotic Systems*, vol. 27, no. 3, pp. 215–235, 2000.
- [68] A. J. Valencia, R. M. Idrovo, A. D. Sappa, D. P. Guingla, and D. Ochoa, “A 3d vision based approach for optimal grasp of vacuum grippers,” in *Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM), 2017 IEEE International Workshop of*. IEEE, 2017, pp. 1–6.
- [69] 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*, 2017.
- [70] J. Weisz and P. K. Allen, “Pose error robust grasping from contact wrench space metrics,” in *Proc. IEEE Int. Conf. Robotics and Automation*. IEEE, 2012, pp. 557–562.
- [71] W. Wohlkinger, A. Aldoma, R. B. Rusu, and M. Vincze, “3dnet: Large-scale object class recognition from cad models,” in *Proc. IEEE Int. Conf. Robotics and Automation*, 2012.

- [72] K.-T. Yu, N. Fazeli, N. Chavan-Daffe, O. Taylor, E. Donlon, G. D. Lankenau, and A. Rodriguez, “A summary of team mit’s approach to the amazon picking challenge 2015,” *arXiv preprint arXiv:1604.03639*, 2016.
- [73] 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,” in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2018.
- [74] 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*. IEEE, 2018.