Sensing and Actuation Technologies for Dexterous Manipulation in Constrained Environments

Patrick Lancaster

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2022

Reading Committee:
Joshua R. Smith, Chair
Siddhartha S. Srinivasa, Chair
Sawyer Fuller

Program Authorized to Offer Degree:
Paul G. Allen School of Computer Science and Engineering

University of Washington

Abstract

Sensing and Actuation Technologies for
Dexterous Manipulation in Constrained Environments

Patrick Lancaster

Co-Chairs of the Supervisory Committee:
Zeutschel Professor Joshua R. Smith
Paul G. Allen School of Computer Science and Engineering
and Department of Electrical and Computer Engineering

Boeing Endowed Professor Siddhartha S. Srinivasa
Paul G. Allen School of Computer Science and Engineering

A primary reason why dexterous robot manipulation remains challenging is because we have not equipped our robots with sufficiently robust perception and control systems. In all but the most structured and unconstrained environments, roboticists must make compromises in designing these systems due to an artificially limited set of available sensing and actuation modalities. Since most environments cannot be outfitted with large arrays of cameras, we accept the use of one or a few cameras that are sensitive to lighting conditions and whose view can easily become occluded by environmental clutter or even the robot itself. The range of motion demanded by dexterous manipulation require robots to be highly articulated, forcing us to choose between heavy, power consumptive robots that use many motors or significantly less controllable robots that use fewer.

Here, we develop alternative forms of sensing and actuation, and demonstrate their ability to facilitate dexterous manipulation in real robotic systems. We design a fingertip embedded proximity sensor that is robust to occlusion, and propose a framework that allows the robot to estimate the pose of general objects using these types of sensors. With respect to actuation, we design electrostatic brake equipped joints that are four times lighter and a thousand

times more power efficient than their motor driven counterparts. We demonstrate that underactuated robots augmented with such brakes achieve independent positional control of each of their joints while maintaining low weight and power consumption. Finally, we integrate both fingertip embedded proximity sensors and electrostatic brakes into a single robot hand. By combining both of these technologies with adapted Bayesian state estimation and model predictive control algorithms, we demonstrate how alternative forms of sensing and actuation increase the speed and precision of in-hand manipulation.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would first like to thank my advisors, Sidd Srinivasa and Josh Smith, for their incredible support and guidance throughout my PhD. Sidd, thank you for teaching me to hold my work to the highest possible standard. This lens has pushed me to critically think about which research problems to pursue, find the passion needed to discount any fears of failure, and thoroughly understand and articulate the benefits and limitations of the solutions I propose. Josh, thank you for giving me your wholehearted support while also allowing me the freedom to grow as an independent researcher. You have consistently pointed me in interesting research directions and then trusted me to find the path forward as I saw fit, which I will always appreciate. Regardless of whether it was a robot fingertip, robot mouth, robot elephant trunk, robot hand, or more than 50 MuSHR cars, it was an invaluable experience and absolute joy to build robots with the both of you, for which I cannot thank you enough.

I would also like to thank my other committee members, Dieter Fox and Sawyer Fuller, for their feedback. Both of your works have been hugely inspiring to me, and your comments were vital for scoping the final portions of my PhD.

I am fortunate to have had the opportunity to be a part of both the Personal Robotics Lab and the Sensor Systems Lab. Thank you to the members of both groups for providing the mentorship, discussion, collaboration, and outlets for breaks from research that make these labs such great places to work. This also includes the undergrad students that allowed me the opportunity to mentor them, who I thank for their contributions to my work. I am grateful to Chris Mavrogiannis for all of his support during the second half of my PhD. Thank you Chris for encouraging me to be ambitious in the work that I pursue and how I present it. I owe a large debt of gratitude to Lisa Merlin for helping me purchase materials

and equipment. I would also like to thank Ben Waters for mentoring me at the beginning of my PhD and espousing the importance of helping other lab members. Thank you to Jim Youngquist for inspiring me with your approach to computer science, as well as teaching me how to play racquetball. Also, thank you to Boling Yang for the kindness, discussions, and encouragement throughout our time as students. I am very thankful for all of the time we have spent together, both inside and outside of the lab.

I would like to thank my family for all of their love and support up to this point. I especially thank my mom, whose hard work and care allowed me to pursue this path. Finally, thank you to Li for being by my side since the beginning of this journey, it would not have been possible without you.

DEDICATION

to my grandparents, Linda Chang and James Lee

Chapter 1
INTRODUCTION

Robots that fail to embrace a diverse toolset of sensing and actuation will not be able to robustly perform manipulation in our uncertain, unstructured world. The dominant modalities of sensing and actuation for robot manipulation are vital building blocks of robots that can achieve many tasks, but these modalities' incredible success in previous problems can lead to them being used as hammers for new problems that are not nails. Therefore failure to investigate, develop, and employ multi-modal sensing and actuation systems will soon result in a plateauing of robot manipulation efficiency and capability.

Even the most widespread modalities for sensing and actuation have their own unique drawbacks that can degrade the robot's ability to perform manipulation. A camera's view of an object can be occluded by other objects, or even by the robot itself. Depending on the type of camera, image sensor data can be sensitive to lighting conditions and specularities. The most dexterous manipulators typically have many degrees of freedom, where each joint may be actuated by one or more electromechanical motors. Actuating a large number of joints with motors may result in the robot being prohibitively heavy, large, and power consumptive. These challenges are relatively easy to solve in controlled settings, but real-world manipulation scenarios remain difficult.

The deficiencies of single modality perception and actuation systems have been adequately addressed in controlled research settings, but these solutions are unlikely to extend to unstructured environments. Disaster zones, deep under the sea, and even most indoor environments cannot be instrumented with the controlled lighting and multi-camera systems that facilitate robust perception in research labs. Robots in research environments enjoy

access to virtually unlimited electrical power, but the need for untethered operation in many real-world environments requires the robot to draw power from on-board batteries. Motors used for actuation can consume a significant amount of the robot's battery power, and their weight further increases the amount of energy the robot must expend to move and/or manipulate. Relying on any single modality for sensing or actuation limits the environments in which robots can perform manipulation and the quality of the manipulation itself.

Recognizing that no single modality of sensing or actuation will be able to perform optimally in all tasks and environments is key to increasing the manipulation capabilities of our robots. This observation motivates the overarching goal of this thesis:

Design novel robot sensors and actuators that succeed in the constrained environments that previous modalities fail.

These new modalities will augment the capabilities of existing sensing and actuation systems, allowing robots to succeed in a wider variety of manipulation tasks and scenarios. This thesis first develops and analyzes new techniques for sensing and actuation separately, and then studies their combined effect on manipulation when integrated into a single robot hand.

1.1 Proximity Sensor Based Pose Estimation

While there are many different aspects of perception that are important for manipulation, understanding of the object's pose is perhaps most fundamental. Before the robot can reason about which actions it should take to manipulate an object, it must know where that object is with respect to its manipulator. We believe that end-effector embedded optical proximity sensors facilitate pose estimation in many of the situations in which conventional sensors fail.

Unlike cameras mounted to the robot or its environment, end-effector embedded proximity sensors are robust to occlusion because they operate much closer to the manipulation site [Azad et al., 2007, Saxena et al., 2008]. While tactile sensors can similarly avoid occlusion, they require contact in order to detect objects [Li et al., 2014, Donlon et al., 2018].

Making contact before the object is well localized could unintentionally displace the object, and measurements limited to the areas of contact may not sufficiently disambiguate the pose of the object. By measuring at a range beyond the contact surface, proximity sensors detect the object before making contact [Navarro et al., 2021]. Once contact has been made, they can measure regions of the object that don't necessarily conform to the surface of the manipulator. These sensing capabilities facilitate robust object pose estimation both before and during manipulation of the object.

In this thesis, we design optical proximity sensors for robot fingertips and then apply them to pose estimation during manipulation. Generally, the robot will not have a perfect model of how its actions affect the pose of the object during manipulation. This imperfect model produces some error between the robot's perception and the object's true pose after each action, and uncorrected error can compound and eventually cause failure during a series of manipulations. We study how fingertip embedded optical proximity sensors can correct this error in the context of robot Rubik's cube solving, an example of highly sequential manipulation (Chapter 3). In order to accurately estimate the pose of an object, the proximity sensor must detect discriminative features of the object's geometry. To extend proximity sensing for pose estimation beyond cube shaped objects, we demonstrate the use of a learned model to detect regions of objects that will yield discriminative features when scanned by fingertip embedded proximity sensors (Chapter 4). While these sensors are useful prior to making contact with the object, they have a minimum sensing distance that limits their capabilities during contact. Therefore we develop an end effector mounted sensor for proximity, contact, and force sensing, and discuss how the design of its compliant contact surface optimizes performance (Chapter 5).

1.2 Hybrid Motor-Brake Actuation for Dexterous Manipulation

The range of motion required for dexterous manipulation often demands that the robot be highly articulated. However, actuating a large number of joints can be challenging. Fully actuated robots that use at least one motor to actuate each joint have a high degree of

control, but employing a large number of motors may violate weight, volume, and power consumption constraints [Billard and Kragic, 2019]. To satisfy these constraints, robots can be designed to use fewer motors and instead couple multiple joints together using tendons or other transmission mechanisms [He et al., 2019]. The downside of this underactuated strategy is that the robot becomes less dexterous; in particular, it can no longer control each joint independently of all the others.

We can augment conventional underactuated robots with alternative forms of actuation in order to regain the ability to independently control each of its joints without adding significant weight, volume, and power consumption. Specifically, we propose to install an electrostatic brake into each joint of the robot. When a joint's electronically controlled brake is engaged, it prevents motion of the joint by counteracting any torques exerted on the joint. Even if multiple joints are mechanically coupled together by a motor-actuated tendon, their motion can be decoupled as desired by controlling the state of the brakes. We refer to this control strategy as *hybrid motor-brake actuation* because the mechanism uses both electromechanical motors and electrostatic brakes for control. This thesis presents the design of an electrostatic brake equipped joint that maximizes braking strength via optimization of electrode conformance. By creating a joint that supports the stacking of thin brake electrodes, the braking strength can be scaled according to application requirements while remaining lightweight, compact, and power efficient (Chapter 6).

1.3 Application to In-Hand Manipulation

The ability to re-position and re-orient an object within the hand's grasp is a critical component of many higher level tasks that are desirable for robots to perform. In-hand manipulation scenarios exacerbate many of the perception and actuation challenges that we have previously discussed [Billard and Kragic, 2019]. The contact rich nature of in-hand manipulation makes it highly likely that the view of a camera will at least be partially occluded by the robot's fingers. Also, the robot's actuators must empower the robot to exert forces on the object that both move the object towards the desired pose and prevent it from being

ejected from the robot's grasp.

In this thesis, we build a two-fingered robot hand with integrated fingertip-embedded proximity sensors and electrostatic brake equipped joints in order to examine how alternative forms of both sensing and actuation can contribute to dexterous in-hand manipulation (Chapter 7). Here, the in-hand manipulation task consists of the robot moving a cylindrical object from one side of its workspace to the other. The six degree-of-freedom robot has only two motors, but the use of brakes enhances the speed and precision with which the hand can manipulate the object within its grasp. With respect to perception, fingertip embedded proximity sensors measurements allow the robot to accurately estimate the pose of the object throughout the manipulation without the need for occlusion-prone cameras.

1.4 Contributions

In this thesis, we design new sensors and actuators that serve as building blocks for alternative forms of robot perception and control. We make the following contributions:

- Design a fingertip embedded sensor that not only detects contacts and measures applied forces, but also maximizes contrast-to-noise ratio in the proximity regime via optimization of the contact surface geometry

- Develop a learning framework to guide proximity sensor positioning towards object regions with discriminitive geometric features during pose estimation

- Design an electrostatic brake equipped joint that maximizes braking strength by optimizing the conformance between brake electrodes

- Develop control policies for continuous, hybrid motor-brake actuation of highly underactuated mechanisms

We evaluate all of our contributions on real robotic systems. Our initial experiments focus on characterizing our devices' intrinsic capabilities according to relevant metrics, such

as distance measurement accuracy for our proximity sensor and maximum holding torque for our brake-equipped joint. We study how these individual technologies facilitate different forms of dexterous manipulation, and then examine their combined effect by incorporating both into a new robot hand. Using this common platform, we measure the performance of in-hand manipulation with proximity sensor based bayesian estimation and hybrid motor-brake actuated model predictive control.

1.5 Thesis Organization

The outline of this thesis' chapters and the interdependenices between them is shown in Fig. 1.1. Chapters 1 and 2 provide context for the entire thesis. Chapters 4 and 5 build upon Chapter 3. Chapter 6 is self-contained. Chapter 7 incorporates all of the work of the previous chapters, and Chapter 8 concludes the entire thesis.

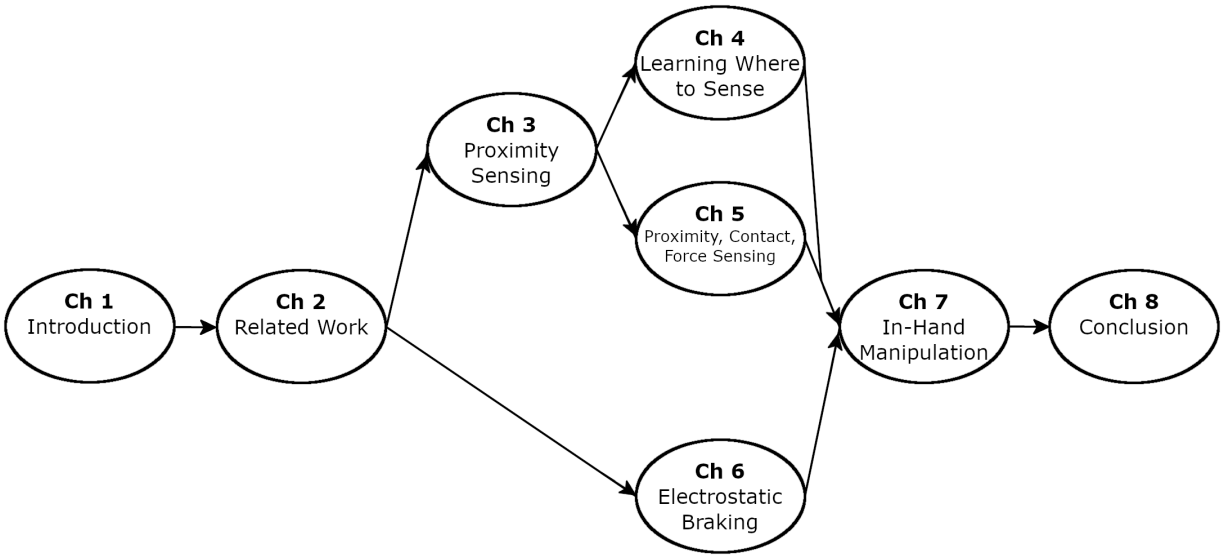


Figure 1.1: The chapters of this thesis and their interdependencies.

1.6 Publication Note

Most of Chapter 3 appeared in Yang et al. [2017]. This is joint work with Boling Yang, in which he implemented the task and motion planners and I created the sensor hardware. Most of Chapter 4 appeared in Lancaster et al. [2017]. Most of Chapter 5 appeared in Lancaster et al. [2019]. Most of Chapter 6 appeared in Lancaster et al. [2022b]. Chapter 7 was adapted from both Lancaster et al. [2022b] and Lancaster et al. [2022a].

Chapter 2

RELATED WORK

In this chapter, we examine work related to dexterous manipulation. We particularly focus on in-hand manipulation because it encapsulates many of the challenges in perception and actuation that this thesis addresses. We highlight these challenges by emphasizing the sensing modalities used for object pose estimation, and the mechanism design and actuation architecture of various dexterous hands. In order to provide a foundation for the contributions of this thesis, we also discuss works related to the sensing and actuation technologies that we aim to apply to dexterous manipulation.

2.1 Pose Estimation for In-Hand Manipulation

Manipulation of an object inherently constrains its pose to some degree. Even without exteroceptive sensing, a robot can reason about how the configuration of its hand constrains the object in order to estimate its pose with some amount of uncertainty. Sequences of uncertainty-aware motion primitives allow robots to perform in-hand manipulation without any exteroceptive sensory feedback [Bhatt et al., 2022]. Such information can be combined with constraints imposed by the motions of the arm, gravity, and external contacts to estimate object pose [Dafle et al., 2014, Chavan-Dafle and Rodriguez, 2015]. While this approach reduces hardware requirements, relying on the executed actions themselves to maintain small pose uncertainty can make it difficult to achieve arbitrary manipulations. Leveraging the constraints on the object's pose imposed by the robot's hand is a useful principle, but the use of exteroceptive sensing helps decouple accurate pose estimation from the particular actions executed by the robot.

The use of image sensors for pose estimation have facilitated the execution of dexterous

in-hand manipulations [Bircher et al., 2021, Hang et al., 2021, Morgan et al., 2022, Higo et al., 2018]. Andrychowicz et al. [2020] initially use a PhaseSpace motion capture system to track both the pose of the object and the position of the fingertips. After developing a working system, they used the learned control policy to generate data for a convolutional neural network that estimates the pose of the cube within the hand. In their experiments, they evaluate a system that replaces the motion capture tracking of the object's pose with the output of this network. Using vision-based object pose estimation decreased the robot's average performance by up to 32 percent in simulation and up to 42 percent on the real robot. Andrychowicz et al. [2020] serves as an example of image sensor based approaches to perception that generally require object mounted visual fiducials and/or a spatially distributed array of cameras. Other environments may not be able to fulfill such requirements, resulting in occlusion and degraded performance.

A number of studies have achieved object pose estimation by combining touch sensors with bayesian filtering techniques, particularly particle filtering [Álvarez et al., 2017, Chalon et al., 2013, Corcoran and Platt, 2010, Platt et al., 2011]. Specifically, these methods constrain their filter's pose estimates by assigning higher weights to particles that are consistent with observed sensor measurements. Koval et al. [2015] present a method to address the issue of particle starvation, in which a large portion of the particles have zero or negligible probability. This issue can become particularly prevalent when the space from which particles are sampled is of higher dimension than that of the space containing object poses consistent with the sensor observations. By sampling from this latter space, called the contact manifold, Koval et al. [2015] are able to demonstrate their method's superiority over more vanilla particle filtering methods.

Liang et al. [2020] leverage tens of separate physics simulations as an ensemble of forward models for estimating the motion of objects within the hand. Their (simulated and real) robot platform is an Allegro Hand equipped with SynTouch BioTac sensors on its fingertips. The parameters of the simulation are tuned using derivative free-optimization methods in order to reduce discrepancies from the real-world. The discrepancy between the real-world and

a simulation are measured by a cost-function, where multiple terms penalize disagreement between the simulation's and real-world's sensor measurements. During evaluation, the robot estimates the real-world pose of the object to be the pose of the object in the least costly simulation. The ablation experiments performed by Liang et al. [2020] indicate that touch sensor feedback more dramatically improved both the average and standard deviation of the final object pose error as the amount of initial pose uncertainty increased.

Other works do not perform explicit state estimation of the manipulation object, and instead learn model-free control policies for in-hand manipulation [Kumar et al., 2016, Falco et al., 2018]. Van Hoof et al. [2015] directly learn a Gaussian process control policy that maps from robot-object state to actions. The task is to use two fingers to laterally shift a cylindrical object resting on a table from side to side. The state space consists of the joint values of their robot hand, and measurements from touch sensors located at the tips of the fingers. Their trained controller is able to outperform a hand-tuned controller, and successfully manipulates unseen cylindrical objects of different sizes.

The sensing ranges of image sensors and tactile sensors place them at opposite ends of the effective object detection spectrum. The views of long range image sensors are easily occluded, while short range tactile sensors require contact to detect objects. They both fail to detect objects in a critical range regime; that is, right before contact is made. When first making contact with an object, the robot's hand may occlude the view of a head-mounted camera, and imprecise initial contact with the object could irreversibly displace the object and render any obtained tactile readings useless. In the context of in-hand manipulation, tactile based pose estimation may struggle to localize objects that do not conform to the surface of the fingertip.

2.2 Proximity Sensing for Robot Manipulation

Pre-touch sensing refers to the use of sensors embedded in the hand to detect objects at a range of millimeters and/or centimeters. These sensors have a range intermediate to that of image and tactile sensors. While we describe many related works here, please see Navarro

et al. [2021] for a more comprehensive survey of proximity sensing in robotics.

Jiang and Smith [2012] develop acoustic pretouch sensors embedded in the fingertips of a PR2 robot. They measure proximity to objects by detecting changes in the ambient noise spectrum inside of the sensor's acoustic cavity. One disadvantage is that the sensor fails to detect extermely soft and light materials. Fang et al. [2019] propose a bi-modal acoustic-optical sensor using the optoacoustic effect.

Capacitive sensors extract information from the environment by measuring changes in capacitance between two or more sensor electrodes. Such sensors can operate in both tactile and proximity modes [Göger et al., 2013]. Wistort and Smith [2008] use electric field pretouch as the feedback signal for closed loop control in robotic manipulation tasks. Mayton et al. [2010] extend this principle for co-manipulation of objects between humans and robots. Faller et al. [2019] retrofit an industrial gripper system with capacitive sensors for automated grasping of logs in a forestry robot. Muhlbacher-Karrer et al. [2015] integrate capacitive sensing for active object categorization in robot manipulation tasks. However, the performance of capacitive sensors degrades when sensing objects with a low dielectric constant such as fabrics, foam, and plastics.

Optical pretouch sensing methods are attractive because of their precision and ability to detect a wide range of materials. Guo et al. [2015] propose to perceive objects using a transmissive optical proximity sensor composed of an emitter and a receiver in a parallel gripper. Hsiao et al. [2009] use optical sensors for pose estimation based on a probabilistic model. Maldonado et al. [2012] augment long-range vision with optical sensors to obtain measurements of areas that are occluded from the image sensor. Yang et al. [2017] demonstrate continuous manipulation tasks using a time-of-flight (ToF) proximity sensor mounted on a parallel gripper.

Other works have developed optical sensors encased in transparent, compliant materials that facilitate both proximity and tactile sensing during manipulation [Patel and Correll, 2016, Lancaster et al., 2019]. Sasaki et al. [2018] successfully exploit ToF sensors to increase the robustness of the positioning of a robot hand during a grasping task. Optical sensors

may fail to detect objects that are transparent or highly specular. Methods that have been proposed to compensate for variations in reflectance include color calibration information obtained by a vision sensor [Konstantinova et al., 2016] and using a light-emitting diode with emission phase that is robust to varying reflectance [Koyama et al., 2019].

2.3 Design and Actuation of Dexterous Hands

A fair number of previous works have used the Shadow Hand as a platform for robot in-hand manipulation [Andrychowicz et al., 2020, Kumar et al., 2016, Nagabandi et al., 2020]. Twenty DC motors antagonistically actuate the tendons routed throughout the Shadow Hand's palm and phalanges (though there are also pneumatically actuated versions of the hand). The hand contains a total of 24 degrees of freedom, where each of the four pairs of the most distal joints in the non-thumb phalanges are coupled, and the remaining 16 joints are independently controlled. Andrychowicz et al. [2020] note that the hand incurred breakages throughout their experiments, particularly at the wrist-pitch joint. The Shadow Hand is a product offered by the Shadow Robot company, and is sold for many tens of thousands of dollars [Xu, 2019].

The Allegro hand is a commercially available alternative to the Shadow Hand. It has also been used in a number of in-hand manipulation studies [Liang et al., 2020, Hang et al., 2016, Or et al., 2015], but it is less kinematically similar to the human hand than the Shadow Hand. In particular, it only has four fingers instead of five, and does not have direct adduction/abduction of the fingers. Instead, each of the fingers is able to roll along its primary axis. Each of its fingers contains four independently controlled joints for a total of sixteen degrees of freedom. Unlike the tendon-actuated Shadow Hand, each of the Allegro Hand's joints is driven by a co-located DC motor. This hand is sold by the Wonik Robotics company for approximately $15,000 [Zhu et al., 2019]. Zhu et al. [2018] observed that experiments with the hand caused it to heat up, requiring a rest period to avoid permanent damage.

The Twendy-One Hand is the primary end-effector for a home-care robot research plat-

form [Funabashi et al., 2015]. It is kinematically similar to the Allegro hand with 16 degrees of freedom evenly distributed across four fingers. The most distal pairs of joints in the three non-thumb fingers are coupled, and the remaining ten joints are independently controlled. Actuation is achieved through thirteen DC motors, each of which is located near the joint that it actuates. While there is only limited documentation of this hand, it has been shown to be capable of re-orienting objects within its grasp.

The DLR/HIT II Hand is a five fingered gripper with 20 degrees of freedom [Liu et al., 2008]. Each of the five fingers are identical, with the two most proximal joints being independently controlled, and the two most distal joints being coupled. Similar to the Allegro and Twendy-One hands, DC motors are built into the fingers to actuate nearby joints. While the flexion/extension joints of the fingers have a rotation range of 90 degrees, the abduction/adduction joint is limited to 40 degrees. Despite the highly actuated nature of the hand, we were unable to find any instances of it being used for in-hand manipulation.

Xu and Todorov [2016] present a hand that aims to mimic the biological structure and function of the human hand as closely as possible. The links of the hand are 3D printed replicas of a cadaver's hand bones. To form joints, bones are connected by crocheted ligaments that resemble those present in biological human joints. These ligaments passively regulate the abduction/adduction of the fingers. The hand is actuated by tendons connected to ten servo motors; a pair of motors are each dedicated to the extension and flexion of the index and middle fingers, an additional motor controls abduction/adduction of both the index and middle fingers, another pair of motors actuate the flexion and extension of both the ring and pinky fingers, and the remaining three motors control the motion of the thumb. Xu and Todorov [2016] demonstrate in-hand manipulation by teleoperating the hand to rotate and reposition a white board eraser held by the hand.

Although the immense articulation of these hands allows them to attain a dexterous range of motion, their actuation strategies produce undesirable effects. The use of 10 or more motors to actuate each hand adds significant weight. This could disqualify their use in a number of applications, such as prosthetics in which light weight mechanisms are vital.

These motors also consume a significant amount of power, which limits the application of these hands to power constrained environments.

2.4 Electrostatic Actuation

There are a number of applications in which the need for lighter, more compact, power-efficient, and inexpensive actuation has prompted the use of electrostatic brakes. Such applications include communications equipment, tactile displays, virtual reality, and exoskeletons. Furthermore, electrostatic force generation technologies (including but not limited to electrostatic braking) have been applied to robotic grasping, locomotion, and modular robotics. In this section, we review a broad range of applications of electrostatic braking, and discuss the use of electrostatic force technologies in robotics.

2.4.1 Electrostatic Brake Applications

Johnsen and Rahbek [1923] perform some of the earliest investigations into electrostatic braking. They provide empirical observations of the effects of applied voltage and conductor separation distance on the resulting electrostatic force, and find that the amount of current required to achieve significant electrostatic forces is small (i.e. on the order of a few microamperes). Furthermore, they note that using a dielectric to separate the conductors of the brake with high voltage difference can lead to a buildup of charges in the dielectric that is detrimental to the observed braking effect. They instead separate the conductors with a rigid semi-conductive layer and model the effective separation distance between the conductors as the air gap(s) between the semi-conductive layer and the conductors. Such air gaps are the result of the conductive and semi-conductive layers' inability to perfectly conform to one another due to rigidity and imperfect surface smoothness. Johnsen and Rahbek [1923] also describe the zippering effect in which the points at which the conductors are closest act to pull nearby points closer to each other. Finally, they use electrostatic brakes to create multiple types of equipment for use in telecommunications, namely an electric relay, telegraph recorder, and a loud-speaking telephone.

More recently, electrostatic braking has been used to create various types of haptic interfaces. Zhang and Follmer [2018] use electrostatic brakes to control the pins of a 2.5D tactile display. Given a desired tactile pattern, a linear actuator extends in order to push all of the pins up and out of the base of the display. As the linear actuator contracts and allows the pins to fall, each individual pin's electrostatic brake will engage in order to form the overall desired tactile pattern. Each pin of the 4-by-2 array can withstand at least 28 gram-force of static loading force applied by the user. Hinchet et al. [2018] create a virtual reality glove that can restrict the movement of the thumb and forefinger via electrostatic brakes. Activating the brakes when grasping a virtual object provided the users with effective haptic feedback and improved grasp precision.

Diller et al. [2016] use electroadhesive clutches to design an ankle exoskeleton. Activation of each individual clutch causes a corresponding spring to engage with the exoskeleton. By activating different combinations of clutches, Diller et al. [2016] are able to produce six different levels of stiffness throughout the user's walking gait. Their electroadhesive clutch achieves three times higher torque density while consuming two orders of magnitude less power per unit torque relative to other electronically controlled clutches.

2.4.2 Electrostatic Technologies in Robotics

The underlying principles of electrostatic braking have been applied to a number of different tasks in robotics. Dieletric elastomer actuators (DEA) consist of two conductors separated by a compressible dielectric [Pelrine et al., 2000]. The attractive electrostatic force between the conductors resulting from applying a large voltage across them produces significant strains in the dielectric. Ji et al. [2019] integrate three DEAs into an insect sized robot. Each DEA actuates a leg at 450 hz in order to steer the robot and move at speeds of up to 3 cm/s. Prahlad et al. [2008] use electroadhesive technology to create robots that can locomote along walls. Given a sequence of highly compliant electrodes embedded in the robot, adjacent electrodes are oppositely charged in order to induce reciprocal charges in the wall substrate. The resulting attractive electrostatic forces allow both tracked and biomemetic robots to

climb walls of many different surface types. Similar electroadhesive methods have been applied to robot grasping, particularly to manipulate objects that are irregularly shaped or delicate [Guo et al., 2018, Savioli et al., 2014, Shintake et al., 2016]. Karagozler et al. [2007] use electrostatic force generation as a latching mechanism for modular robots. They also explore inter-modular power transfer and communication via the capacitive coupling that the electrostatic latch provides. Aukes et al. [2014] create an underactuated robotic gripper that uses electrostatic brakes to achieve hand configurations useful for grasping, as well as to increase the maximum pullout force of power grasps. Although their brakes allowed the underactuated hand to reach discrete configurations that would not otherwise be possible, the use of large DC voltages to engage the brakes caused dielectric charge injection that makes engagement and disengagement of the brakes relatively slow. This limits the use of braking to discrete changes in hand configuration rather than continuous control. However, a number of works have shown that the effects of charge injection can be mitigated by using AC voltages to actuate the brakes [Hinchet et al., 2018, Hinchet and Shea, 2020, Zhang and Follmer, 2018, Diller et al., 2016, Karagozler et al., 2007]

Chapter 3

PRE-TOUCH SENSING FOR SEQUENTIAL MANIPULATION

In this chapter, we establish an initial manipulation framework rooted in the use of fingertip embedded proximity sensors for object pose estimation. We formulate this framework in the context of sequential manipulation tasks, in which each executed action can introduce additional error into the robot's perception of the object's pose. While open-loop systems will generally fail to prevent accumulation of such error, our framework uses sensor feedback to correct the robot's estimate of the object's pose and thereby more robustly perform manipulation. Here, we apply the framework to the task of Rubik's cube solving, which requires the robot to translate, rotate, and re-grasp the object repeatedly. The initial manipulation framework presented here serves as a foundation for later chapters in which we extend pose estimation to a much a wider range of objects and enhance the capabilities of our proximity sensors.

3.1 Reducing Pose Error with Pre-touch Sensing

As robots continue to transition from operating in controlled, carefully designed environments towards human-centric, unstructured ones, they will have to make more sophisticated use of sensing to cope with the inherent uncertainty in our real world. In particular, robust robot manipulation is difficult to achieve because of the uncertainty involved in manipulating an object [Kemp et al., 2007]. For tasks that require sequential manipulations, the robot's belief about the pose of the object at any point in time can be corrupted by a poor characterization of the initial pose, or through the accumulation of error caused by controller noise, previous imperfect manipulations, and perceptual errors. Depending on the precision required, such errors can cause the robot to fail at the task of interest.

We choose to tackle the task of solving the Rubik's cube because the challenges that general purpose robots face when attempting to achieve sequential manipulation are well represented by this task. Solving a Rubik's cube can require up to 20 rotations [Rokicki et al., 2014]. For each rotation, the robot may have to alter its grasp on the cube by transferring it from one hand to the other and/or shifting the grasp point(s). Because each face of a Rubik's cube contains sub-cubes of a dimension of only 1.9 cm, each manipulation may only be able to tolerate error on the order of millimeters. Even if the robot is able to complete a rotation at a given point in time, an imprecise grasp could cause future rotations to fail by invalidating the robot's belief of where the cube is with respect to the robot's gripper.

In this chapter, we show that accumulating manipulation errors can be controlled by proximity sensors mounted to the robot's end-effectors, what we call *pre-touch* sensors. To this end, we introduce a new type of pre-touch sensing based on optical time-of-flight measurements, and fully integrate it with the PR2 system. We hypothesize that pre-touch scanning enables a robot to gain the geometric information necessary to estimate the pose of an object of interest, and thereby perform actions that are common in sequential manipulation, such as re-grasping. We demonstrate that pre-touch sensing allows a robot to much more effectively complete a particular sequential manipulation task, i.e. solving a Rubik's cube. Whereas previous work on pre-touch sensing has mostly focused on single-shot grasping, we believe that solving the Rubik's cube is the most complex, sequential task to which pre-touch sensing has ever been applied.

3.2 Sensor Hardware

Our optical pre-touch sensor measures the distance to a surface using the VL6180x optical time-of-flight sensor module created by ST MicroElectronics [ST , 2014]. It is capable of measuring the distance to an object with millimeter accuracy at a range of 1cm to 10cm (Fig. 3.1). Furthermore, some objects (particularly those with light colors) can be detected out to a range of 25.5cm.

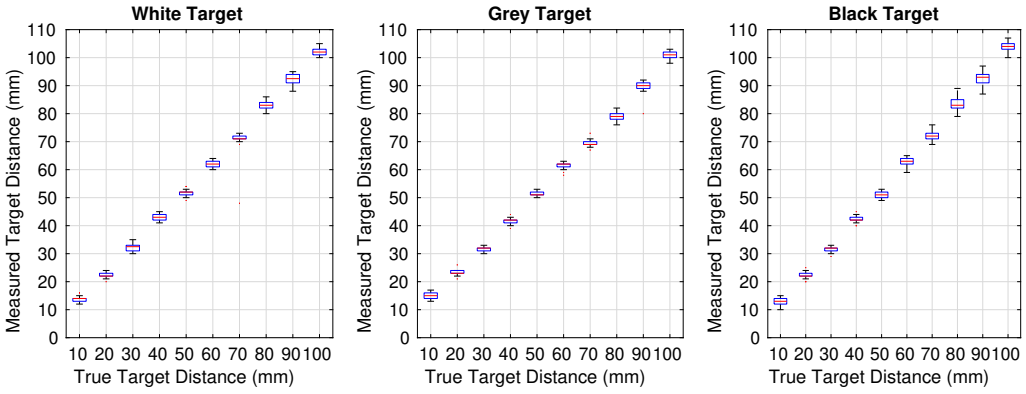


Figure 3.1: Boxplots of sensor measurements over the specified range for white, grey, and black target objects. Each box consists of 30 measurements.

The sensor supports up to six VL6180x sensing modules, where one can be placed at the tip of the finger, two on each side, and one in the pad of the finger. The location of each sensing module is labeled in Fig. 3.2, and will be referenced as such when referring to a specific sensor module throughout the rest of this work. The sensor's design consists of a main board (29x16.5mm) and a secondary board (8.25x5.75mm). The main board hosts an ATMega168PA microcontroller that communicates with each of the VL6180x sensing modules over I2C. The sensor module on the pad of the fingertip sensor is directly soldered to the main board. Every other module is connected to the main board through 1mm pitch headers that interface between the main board and secondary boards. The sensor's casing was designed to match the form factor of the PR2's fingertip.

The robot is able to collect data from each of the six sensing modules at a rate of 30 Hz. Measurements are passed from the sensor's microcontroller to the robot through an SPI interface built into the gripper and then published to a ROS topic. Two metal screw inserts are pressed into the sensor's casing so that it can be fastened to the robot's fingertip.

The components required to build one of these optical sensors cost less than $100. The circuit schematics, PCB files, sensor casing CAD files, and firmware are publicly available at https://bitbucket.org/planc509/optical-distance.

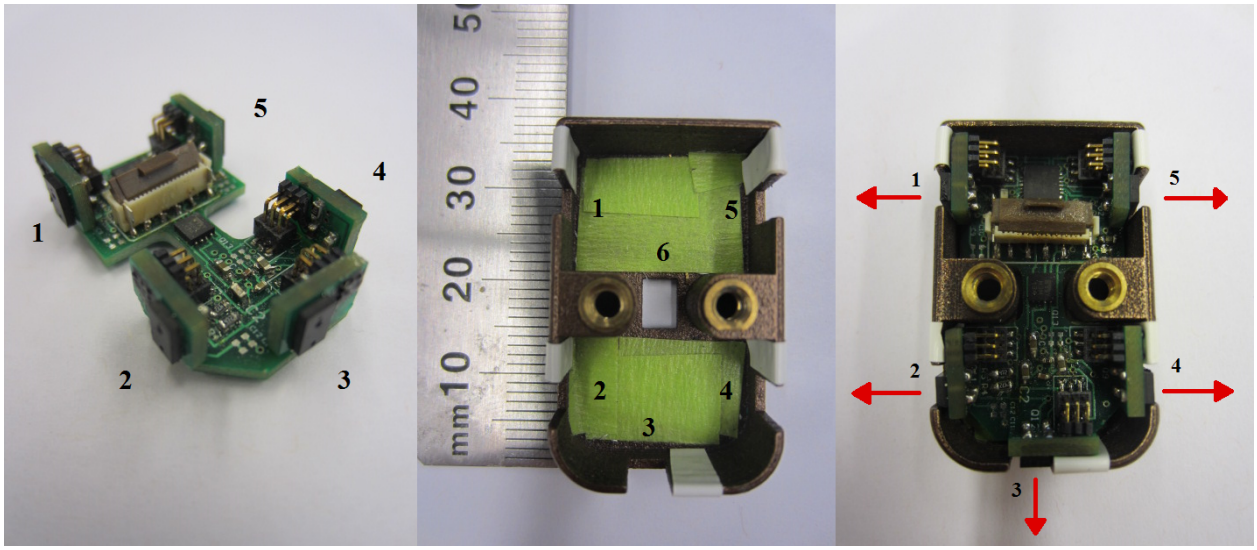


Figure 3.2: Left: The printed circuit boards that compose the sensor. Middle: The 3D printed sensor casing. The hole in the middle of the case is for a sensing module soldered to the bottom of the main PCB. Right: The assembled sensor with arrows denoting the directions of five out of six sensor modules' infrared beams.

3.3 Optical Pre-touch Scanning for Rubik's Cube

In this work, pre-touch sensing is used to estimate the pose of the Rubik's cube not only to marginalize positional error in the next grasp, but also to correct any error that does occur from one manipulation to the next. Note that when describing our method, any coordinate references are with respect to the coordinate frame of whichever gripper is currently holding the object. That is, the y-axis extends along the direction that the gripper opens and closes, the x-axis extends out along the direction that the robot's fingertips point, and the z-axis is orthogonal to both the x and y-axes using a right-handed coordinate system.

As the robot solves the Rubik's cube, it will need to transfer it from one hand to the other, as well as change how it is grasping it. Before each re-grasp, the robot uses pre-touch scanning to refine its estimate of the pose of the cube with respect to the coordinate frame

of the gripper that is currently holding it. The robot assumes that the cube is oriented such that its upper and lower faces are approximately parallel to the ground. This assumption is not always true, but works well in practice. Furthermore, the robot already has a good approximation of the center of the cube's position in the y direction and its rotation around the y-axis because the cube is held between the robot's fingers. However, due to errors in previous re-grasps, the cube could have shifted unexpectedly along the x and/or z directions. There are many pre-touch scanning strategies that could be used to estimate the position along these two directions, but our method used the following strategy in order to minimize the amount of actuation required:

1. The gripper that is not holding the cube is opened if it is not already open.

2. The optical pre-touch sensor on that gripper's fingertip is then oriented such that the beam of sensing module 3 is normal to one of the faces of the cube that is parallel to the yz-plane. This gripper is positioned such that the beam is not yet broken by the cube.

3. The gripper begins to close, causing the sensor to move in the y-direction. As the gripper closes, any significant change in the sensed distance indicates the position of the edge of the cube, allowing the robot to infer the cube's position along one of the uncertain axes.

4. Once the gripper has finished closing, the robot uses the sensor's distance measurements at the current position to estimate the position of the cube along the remaining uncertain axis.

This pre-touch scanning strategy was integrated into a baseline system that uses a computer vision module to recognize the colors of the cube faces, an iterative deepening A* search [Rokicki et al., 2014, Riano, 2011] to determine the necessary cube rotations, and a

finite-state machine based motion planner to execute the trajectories necessary to solve the cube.

3.4 Rubik's Cube Manipulation Evaluation

In order to determine the effectiveness of pre-touch scanning in the context of Rubik's cube solving, a system for manipulating the cube was created in which pre-touch sensing could be enabled or disabled. When pre-touch is enabled, the system uses the sensing strategy described in Section 3.3. When pre-touch is disabled, the system serves as a baseline for what is achievable without pre-touch sensing. Instead of scanning the cube after each re-grasp, the baseline system just assumes that the robot re-grasped the cube in the exact desired location.

Setup

We generated 10 random cube configurations that required between 20 to 23 rotations for the system to solve, and had both the baseline and pre-touch enabled versions of the system attempt them. In addition to reporting the success/failure rate, the robot's estimate of the cube's position throughout each trial is examined for both methods. Prior to each re-grasp, the robot's estimate of the cube position was recorded. All pose estimates were transformed into the frame of the gripper currently grasping the cube. In order to get ground truth measurements of the cube's position, an AR tag was attached to each face of the cube. We then used a Kinect mounted to the robot's head and cameras external to the robot to detect and estimate the position of the cube when appropriate. An AR tag was also added to each of the robot's grippers at a fixed distance away from that gripper's coordinate frame. This allowed us to compute the pre-touch enabled pose estimate (and corresponding ground-truth) of the cube without using the robot's coordinate transforms. Although our robot was re-calibrated prior to beginning this work, there was still significant error in the coordinate transforms (as there would be for any calibration of a high degree of freedom robot). Despite the use of AR tag detection as a 'ground truth' estimate of the pose of the cube, we are not

Table 3.1: End-to-end Rubik's Cube Solving

	Success	Fail	Avg. Rotations Completed
Baseline	0	10	9.6
Pre-touch	8	2	20.1

implying that this method is better than pre-touch sensing for pose estimation. This method will have its own errors depending on how well the tag is detected, and has the disadvantage of requiring one or more tags to be placed on any object whose pose is to be estimated.

End-to-end Results

The experiment demonstrated that the robot's ability to solve the Rubik's cube was significantly enhanced by pre-touch sensing, as shown in Table 3.1. Using the pre-touch enabled method, the robot successfully solved 8 out of 10 cube configurations. As for the two failure trials, the robot finished 14 and 19 rotations out of 21 and 20 total required rotations before failing to complete a rotation. On the other hand, the baseline method did not successfully solve any of the puzzles. The maximum number of successful rotations for any of the 10 trials was 17, the minimum was 3, and the average was 9.6. All of the unsuccessful rotations occurred when the robot failed to re-grasp the cube; either as it tried to transfer the cube from one gripper to the other, or as it attempted to grasp the cube in order to rotate a face. These results demonstrate that although re-grasping motions are very sensitive to positional error, pre-touch sensing allows the robot to effectively compensate for them.

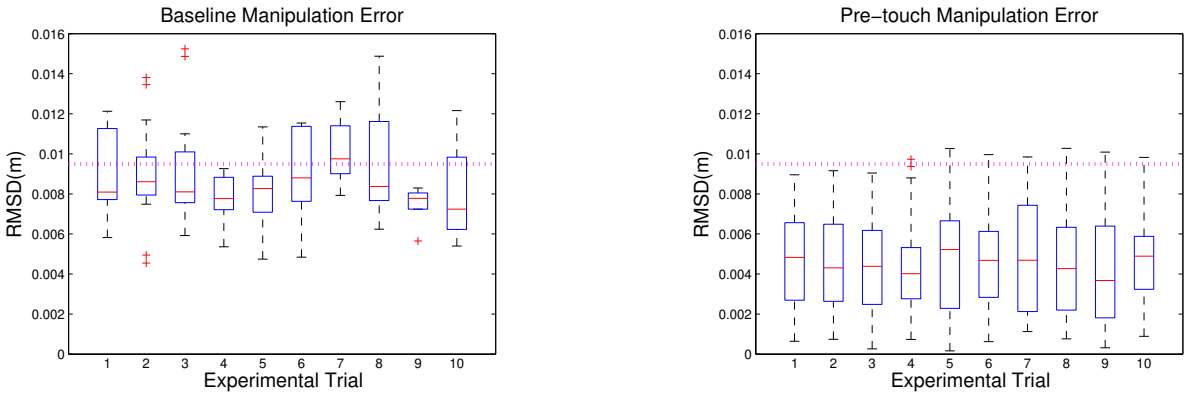


Figure 3.3: Box plots plots of positional error for the baseline (left), and corrected pre-touch (right) methods. Each box corresponds to one of the 10 trials and consists of all cube pose RMSD errors observed during that trial. The RMSD error is recorded prior to each re-grasp. The horizontal line across each plot denotes half of the dimension of a sub-cube, demonstrating that the increased dexterity provided by pre-touch sensing is significant for this task.

Intermediate Pose Estimation

Comparing the robot's positional estimate of the Rubik's cube for both methods to the ground truth values also yields interesting results. The robot re-estimated the pose of the cube prior to each re-grasp. For each pose estimate, the error was calculated as the root-mean-square deviation (RMSD) between the x and z axes of the estimate and the recorded ground-truth. The RMSD throughout each trial for both methods are summarized as box plots in Fig. 3.3.

The observed errors demonstrate that pre-touch sensing allowed the robot to significantly reduce the amount of error in the robot's estimate of the pose of the Rubik's cube. The robot has a finger width that is approximately equal to the size of a sub-cube of the Rubik's cube. Given that the desired grasp point is exactly in-between two sub-cubes, the margin of acceptable error is approximately half the width of a sub-cube. More error than this could

cause the robot to unintentionally constrain one of the faces, causing a future manipulation to fail. Alternatively, error above this threshold could cause the robot to fail to even grasp the cube. The left plot of Fig. 3.3 demonstrates that for most trials, a large portion of the baseline method's pose estimates had an RMSD significantly larger than half of the size of a sub-cube. In contrast, the right plot shows that the RMSD of pre-touch aided estimates very rarely went above this threshold. In fact, for the pre-touch enabled method, most pose estimates had an RMSD of less than 0.8cm. Thus, through the use of pre-touch scanning, the robot has been able to limit the error in its estimate of the cube's pose and thereby solve it more robustly.

3.5 Discussion

This work presented methods for using pre-touch scanning to help robots perform sequential manipulation. We first developed a new optical time-of-flight pre-touch sensor that is composed of inexpensive components. We then showed that this pre-touch sensor allows the robot to precisely re-estimate the pose of the Rubik's cube, endowing the robot with the dexterity necessary to robustly solve the cube.

Now that we have demonstrated the utility of pre-touch scanning in achieving sequential manipulation, there are several directions in which we would like continue our work. One area is to explore how the robot can determine where to scan and how to generate a trajectory that has a high probability of capturing useful geometric information. Furthermore, we will aim to develop metrics for the quality of an executed scan. Such metrics will be important if the robot is trying to manipulate objects that have the potential to not be well detected by one or more of its sensors. Finally, we would like to examine how multiple modalities of pre-touch scanning can most effectively be fused together for the purpose of robot manipulation.

Chapter 4

IMPROVED OBJECT POSE ESTIMATION VIA DEEP PRE-TOUCH SENSING

In this chapter, we extend our work in the previous chapter to pose estimation of general objects. Here, we estimate the pose of an object by aligning a reference model to measurements from a pre-touch sensor. When the robot measures discriminative geometric features of the object, it is better able to make accurate correspondences between the measurements and reference model. It is important to recognize that the measurements made by fingertip embedded proximity sensors are highly dependent on the position and orientation of the robot's manipulators. Thus pose estimation accuracy is highly dependent on the areas of the object that the robot chooses to scan with its pre-touch sensor. However, for a general object, it is not clear how to explicitly specify which object regions contain discriminative features.

Here we leverage the image processing capabilities of convolutional neural networks to detect object regions that will likely yield discriminative geometric features when scanned by a pre-touch sensor. We develop a dataset of over a hundred common objects, and use it to train our region detection model. We demonstrate that the pose of unseen objects are more accurately estimated using this model relative to baseline methods.

4.1 Extending Pre-touch Sensor Based Pose Estimation to New Objects

In order to achieve object manipulation in unstructured environments, a robot must be able to estimate object pose with respect to its end-effectors at an accuracy that is determined by the task. Head-mounted sensors that are commonly used in robotics may not be able to provide sufficient accuracy because of our inability to perfectly calibrate the robot's joints

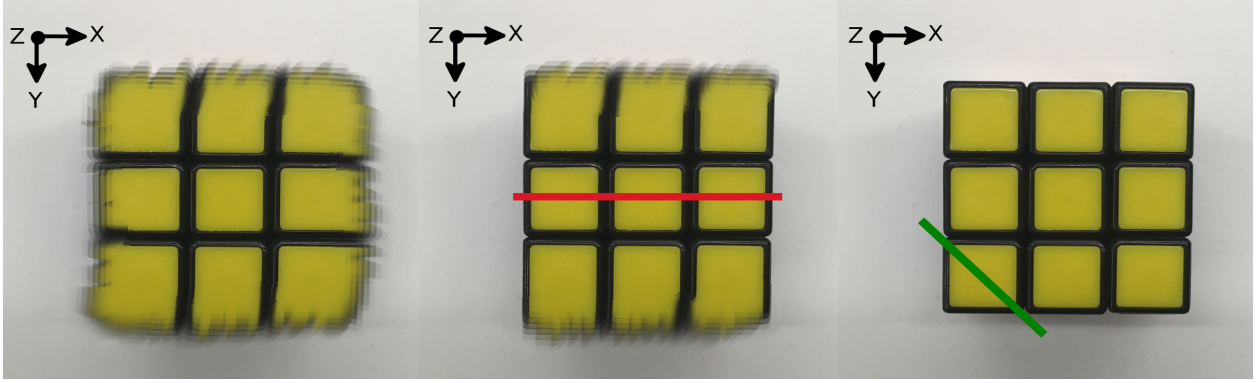


Figure 4.1: Left: The pose in the x and y directions is initially uncertain (as indicated by the blurriness). Middle: A horizontal pre-touch scan localizes the object in the x direction, but its position in the y direction is still uncertain. Right: Alternatively, a diagonal scan at the corner yields additional geometric features that eliminate uncertainty in both the x and y directions.

and cameras. Even slight mis-calibrations can be compounded by the high degree-of-freedom nature of modern robot arms, resulting in a significant positional and/or rotational error. Alternatively, proximity sensors mounted to the robot's end-effectors can avoid these calibration errors, and thereby provide the feedback that is necessary for highly precise tasks. We refer to this type of sensing as *pre-touch sensing* because it allows the robot to measure the object at close range but before making contact. However, these types of sensors typically only provide spatially sparse measurements, and therefore require more actuation to collect enough data to re-estimate an object's pose. In order to minimize actuation time, we propose to train a deep neural network to predict object regions that will yield discriminative information about the object's pose when scanned by a pre-touch sensor.

While specifying which regions are optimal for pre-touch sensing is not, in general, obvious, deep neural networks have the potential to learn feature hierarchies that encode them. An example of what such a deep network might learn is illustrated in Fig. 4.1. Suppose that the robot is uncertain about the pose of the object in the x and y directions, but is equipped

Figure 4.2: A subset of the items in the compiled dataset for learning to detect regions with discriminative geometric features.

with a pre-touch sensor that can measure depth along the z axis. Applying the pre-touch sensor across the horizontal region would provide much information about the pose in the x direction, but little information about the pose in the y direction. A more efficient strategy would be to scan diagonally at the corner of the object, allowing the robot to estimate the pose of the object in both the x and y directions well. While this is an example for a simple object, in general we encourage our network to predict regions that are relatively unique with respect to adjacent regions. We expect that such regions will be irregularly shaped, asymmetric and/or contain edges.

The application of machine learning to pre-touch sensing has not been well explored. To the best of the authors' knowledge, this is the first work to use deep neural networks to predict regions that are amenable to pre-touch sensing. We also created a publicly available dataset (https://bitbucket.org/planc509/pretouch_learn_dataset) of 136 common items with widely varying geometries. A subset of these items is shown in Fig. 4.2. This dataset is applicable to any pre-touch sensor that gathers geometric information. We are unaware of

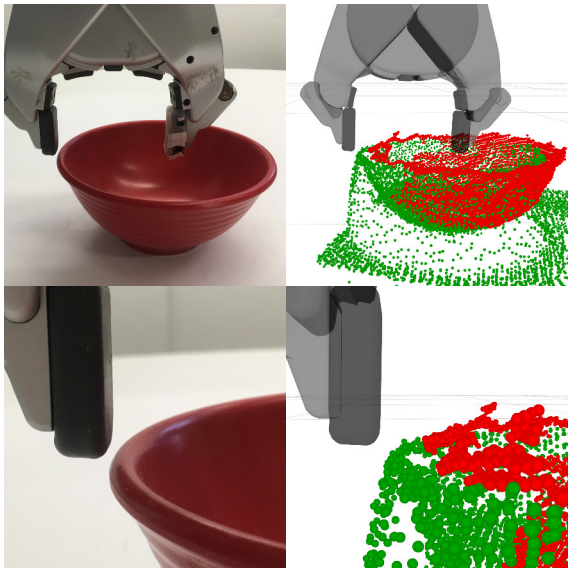


Figure 4.3: Left: A far and close view of the surface of the robot's fingertip aligned with the edge of the bowl. Right: Pre-touch measurements (green) and Kinect measurements (red) with respect to the fingertip.

any other databases for pre-touch sensing of comparable size and variety.

4.2 Pre-touch Sensing for Object Localization

For some robot manipulation tasks, accurate object pose estimation can be vital to task success. However, head-mounted sensors may not be able to provide sufficient accuracy due to imperfect calibration between the sensor and the robot's end effector. Here, we use an optical pre-touch sensor mounted to the robot's end effector to collect geometric information about the object, and thereby avoid the previous calibration error. We then apply the iterative closest point (ICP) algorithm to obtain a more accurate pose.

4.2.1 An Example of Calibration Error

One motivation for the use of pre-touch sensors is the inability to completely eliminate calibration error between the robot's head mounted sensors and its end effectors. Note that just

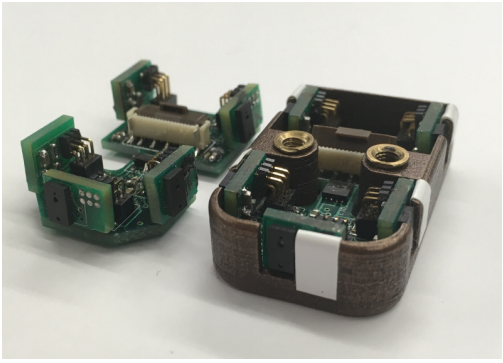


Figure 4.4: The optical time-of-flight sensor, with and without its casing.

prior to beginning this work, our robot was calibrated using standard procedures [Pradeep et al., 2014]. Fig. 4.3 demonstrates the typical severity of mis-calibration, as well as how pre-touch sensing can provide a better estimate of the pose of the object. While in reality the inner surface of the robot's fingertip is aligned with the edge of the bowl, data from the head-mounted Kinect indicates that they are not. In contrast, the data from the optical pre-touch sensor is significantly closer to reality.

4.2.2 Sensor Hardware

Optical pre-touch sensing can provide accurate distance measurements when applied to a wide range of objects. This work uses the optical time-of-flight pre-touch sensor from Yang et al. [2017] and is shown in Fig. 4.4. It is able to measure objects at a range of 1 to 10 cm with millimeter-scale accuracy. The sensor simultaneously measures from its six sensing modules, each at a rate of 30 Hz. However, only the ranging module at the tip of the finger is used in this work.

4.2.3 Data Registration

Given a point cloud of an object from a head-mounted Kinect and a point cloud resulting from a scan by the (more accurate) pre-touch sensor, correspondences between the two sets

of data must be found in order to estimate the spatial transformation between them. It is expected that the discrepancy between their centroids be on the order of a few centimeters, and that the two clouds have similar overall orientations. Aligning two point clouds that have a misalignment of this magnitude is well suited to the ICP algorithm, as opposed to a full registration approach that might be taken when the pair are more drastically misaligned. In aligning the two clouds using ICP, the algorithm yields a spatial transformation between the robot's old error prone belief (based on the Kinect cloud) and its new belief (based on the pre-touch sensor cloud).

While the above strategy allows the robot to re-estimate the pose of an object, obtaining a full pre-touch scan of the object can be expensive in terms of actuation time. It would be more efficient if the robot could re-estimate pose by only scanning a portion of the object and then matching it with the corresponding portion of the original Kinect cloud. The difficulty is that as less pre-touch data is gathered, the number of discriminative features in the scan decreases, making the matching more susceptible to error caused by noise and differing sensor characteristics. However, by choosing regions to scan that are rich in discriminative geometric features, the robot can collect data efficiently while still achieving high quality pose re-estimation. We describe how we learn to detect such regions in the next section.

In this work, we use the Point Cloud Library's (PCL) implementation of ICP [Rusu and Cousins, 2011, Holz et al., 2015]. Beyond PCL's default implementation of ICP, we also use a correspondence rejector that limits the influence of outliers [Phillips et al., 2007].

4.3 Learning Where to Sense

We propose to use a deep neural network to detect object regions that are amenable to pre-touch sensing. Such regions will contain discriminative geometric features that allow the robot to accurately re-estimate the pose of the object. We parameterize these regions as rectangles in the image plane, and re-purpose a state-of-the-art object detection neural network for our task. In order to generate examples from which the network can learn, we form a large set of random candidate regions, simulate pre-touch measurement in those

regions, and retain the regions that are most robust to random noise and offsets when being matched to the corresponding region of the Kinect cloud.

4.3.1 Region Specification

Rotated rectangles represent a flexible, yet simple way to specify regions. Jiang et al. [2011] first proposed using this representation for robot grasping, and many works have adopted it [Lenz et al., 2015, Redmon and Angelova, 2015, Pinto and Gupta, 2016]. In this work, a specified rectangle represents a region at which to apply pre-touch sensing. Specifically, we perform the pre-touch scan along the perimeter of the region. Each rectangle can be parameterized as follows:

$$[x, y, w, h, \theta]$$

where x and y are the center of the rectangle in pixels, w and h are the width and height of the rectangle, and θ is the rotation around the center of the rectangle in the image plane. Following Redmon and Angelova [2015], in practice we use cosine and sine of twice the angle to represent the rotation due to the symmetry of the rectangle.

4.3.2 Neural Network Architecture

As noted in Redmon and Angelova [2015], choosing rectangles to represent areas of interest essentially reduces the problem to object detection (albeit with an additional parameter for rotation), allowing us to take advantage of the powerful tools that have been developed in that area. A common strategy for detection is to classify sub-patches of the image, and then consider the most certain positive classifications to be detections. The naive approach is to individually classify all sub-patches of the image, but this is very inefficient. Lenz et al. [2015] address this issue by using a small neural network to evaluate a large pool of candidates before passing the most promising ones to a larger network. We instead use the Faster R-CNN framework [Ren et al., 2015], which uses a Region Proposal sub-network to indicate to the detection layers which sub-patches are likely to contain a region of interest. While the

original Faster R-CNN implementation [Girshick, 2015] was designed for object detection, we expanded it to be able to predict rotated rectangles (this required us to efficiently calculate the overlap between two rotated rectangles, which was achieved with the General Polygon Clipper library [Murta, 2000]). We initialized the convolutional layers of our network with the 'CNN_M' pre-trained network from Chatfield et al. [2014]. Given a depth image, our network outputs a list of detected regions, and each region is accompanied by a confidence score.

4.3.3 Data Collection

Generating examples of regions that are amenable to pre-touch sensing is not a trivial task. Ultimately we want to choose regions that when scanned by a pre-touch sensor, produce data that can be reliably matched with the corresponding regions of the original Kinect cloud in the presence of random noise and calibration error. Although a human may be able to label some of these areas, his or her intuition may not always coincide with what is optimal from an ICP perspective. Furthermore, if we can automate the data collection process, it will be easier to scale up the size of the dataset such that there is sufficient data to avoid overfitting. Here, we generate a large number of random candidate regions, and then build a set of labels based on which regions yield the most robust ICP matching results. We evaluate a large number of candidates by simulating pre-touch measurement of each region.

The label generation algorithm for a given object is summarized in Algorithm 1. A candidate set of of one thousand rectangles ($n_R = 1000$) is first randomly generated, where each rectangle has an intersection-over-union of at least fifteen percent with the rectangle bounding the object, and has an area that is ten to fifty percent of that of the bounding rectangle. In the *for* loop on line 5, the algorithm builds a target cloud for each rectangle by retaining the subset of points from K whose projections into the image plane are near or on the perimeter of the rectangle. The algorithm then generates ten random ($n_{trials} = 10$) offsets, where the translations in the x, y, and z directions are drawn from a uniform distribution with limits of -2.0 cm to 2.0 cm, and roll, pitch, and yaw rotations are drawn

Algorithm 1 Label Generation

procedure LABELGENERATION(K) ▷ Kinect cloud K
  $R \leftarrow n_R$ random rectangles
  $K2D \leftarrow$ projection of K into image plane
  $T \leftarrow n_R$ empty target clouds
  for `r in 1:`n_R **do**
    for `i in 1:`n_{K2D} **do**
      if $R^r.nearPerimeter(K2D_i)$ **then**
        $T^r.addPoint(K_i)$
  $\Delta K \leftarrow n_{trials}$ random translations & rotations
  for `j in 1:`n_{trials} **do**
    $K' = K + \Delta K^j$ ▷ Offset cloud K'
    $K'2D \leftarrow$ projection of K' into image plane
    $S \leftarrow n_R$ empty source clouds
    for `r in 1:`n_R **do**
      for `i in 1:`$n_{K'2D}$ **do**
        if $R^r.onPerimeter(K'2D_i)$ **then**
          $S^r.addPoint(K'_i)$
      $trialScores^{j,r} = Score(S^r + N(0, \sigma), T^r)$
  for `r in 1:`n_R **do**
    $scores^r = \max_j(trialScores^{j,r})$
  `return` $FilterRecs(R, scores)$
procedure SCORE(P, Q)
  $PAlign = icp(P, Q)$
  `return` $\frac{\sum |PAlign_i - PAlign_i^*|_2}{n_P}$

from a uniform distribution with limits of $-5.0°$ to $5.0°$. For each of these generated offsets, the *for* loop on line 10 applies it to K in order to generate K'. K' represents a simulation of what a pre-touch sensor would measure if it were applied to the whole object. Then the source cloud for each rectangle is built by retaining the subset of points from K' whose projections into the image plane are on the perimeter of the rectangle. This simulates a noiseless pre-touch scan along the perimeter of the region that corresponds to the rectangle. Before matching each source and target pair using ICP, the algorithm adds gaussian noise ($\sigma = 0.15$cm) to the source. The pair is scored by calculating the average distance between the predicted aligned cloud's and the ground truth aligned cloud's points (where the ground truth cloud is computed by applying the inverse of the original offset between K and K' to the noised source cloud). The final score for each rectangle (computed in the *for* loop on line 19) is the worst (i.e. largest) score for that rectangle across all trials. Therefore, in order to get a good (i.e. low) score, a rectangle has to perform well across multiple simulations, which adds robustness to the generated labels. The rectangle with the best score is retained as a label.

4.4 Results

In order to evaluate the proposed framework, we apply it to pose estimation of eight objects that are not in the dataset used to train the deep neural network. For each object, the PR2 robot receives an initial point cloud from the Kinect, which our detection network uses to propose a set of pre-touch scanning regions. For each region, the robot scans along the path that corresponds to the perimeter of the rectangle. After performing each scan with the pre-touch sensor, we match it with the corresponding region of the Kinect cloud, where the corresponding region is computed in a fashion similar to that of lines 6-8 of Algorithm 1. We obtain a ground-truth estimate of the pose by matching a separate pre-touch scan of the whole object with the Kinect cloud. We argue that this is reasonable because, again, the data from the pre-touch sensor is not affected by mis-calibration of the coordinate frames of the robot's arms. We have also observed, for example as in Fig. 4.3, that data from the

pre-touch sensor matches reality better than data from the Kinect. All scans were performed such that the sensing beam was normal to the supporting table for simplicity, but future systems could use more complex scanning strategies.

4.4.1 Individual Scan Comparison

We compare our region proposal network to two other baseline proposal methods. The first baseline simply proposes random rectangles that have the same properties as those used in the label generation process (Section 4.3.3). The second baseline, which is a variant of normal aligned radial features (NARF) [Steder et al., 2011], proposes regions in which there is a large change in the object's surface. This method first calculates standard NARF keypoints and corresponding descriptors. For each key point, the corresponding rectangle is defined by two perpendicular vectors whose origins are at the keypoint. The direction of the first vector corresponds to the element of the descriptor that has the greatest value, and the direction of the second is chosen according to which of the two elements perpendicular to the first has the greater descriptor value. The length of these vectors is chosen so that the rectangle's area is half that of the rectangle bounding the object. The rectangle's confidence score is the sum of the two elements that correspond to the chosen directions.

For each of the three methods, we collected scans corresponding to the first ten proposed rectangles for each of the objects, individually matched them to the corresponding portions of the Kinect cloud as previously described, and computed the average distance across corresponding points of the estimated aligned cloud and the ground truth aligned cloud. These pose estimate errors' means and standard deviations are shown in Table 4.1. The standard deviations can be interpreted as a measure of each method's consistency, where a lower value indicates greater consistency. For all but one of the items, our region detection network had the lowest mean, typically by a significant margin. Also, the network had the lowest standard deviation for most of the items, while being outperformed by a small margin on two of the items.

Table 4.1: Mean and standard deviation pose estimate error across ten individual scans for each of the eight different objects and each of the three methods. All values have centimeter units, and the best value across each of the three methods is bolded.

	Mean Error (cm)			Std. Dev. of Error (cm)		
	Random	NARF	Deep Pre-touch	Random	NARF	Deep Pre-touch
Air Freshener	1.31	2.21	**1.26**	0.52	1.50	**0.50**
Clock	2.21	2.97	**0.80**	1.40	2.16	**0.32**
Cleanser	1.93	**0.71**	1.16	1.46	**0.58**	0.60
Controller	2.77	2.91	**1.49**	1.33	1.96	**0.76**
Fruit Bowl	1.69	1.87	**0.83**	1.50	1.46	**0.29**
Gripper	2.24	1.48	**1.11**	1.39	0.95	**0.41**
Toy	2.55	1.94	**1.72**	1.13	0.66	**0.43**
Wallet	1.16	1.70	**0.89**	**0.63**	0.83	0.66

4.4.2 Deep Pre-touch Sequential Scanning

While the previous section demonstrated the performance of individual scans proposed by the region detection network, it is possible that a smaller margin of error is desired. We hypothesize that even better pose estimation can be achieved by concatenating multiple scans together. Specifically, we create a sequence of regions to be scanned by ordering them by their confidence scores. After performing each scan with the pre-touch sensor, we concatenate it with the previous scans before matching it with the corresponding regions of the Kinect cloud.

The results of the pose estimation for each object are shown in Fig. 4.5 and Fig. 4.6. As seen in Fig. 4.6, the pose estimation for each object typically improves as additional area is scanned. Fig. 4.6 also shows that most of the objects could be localized to within

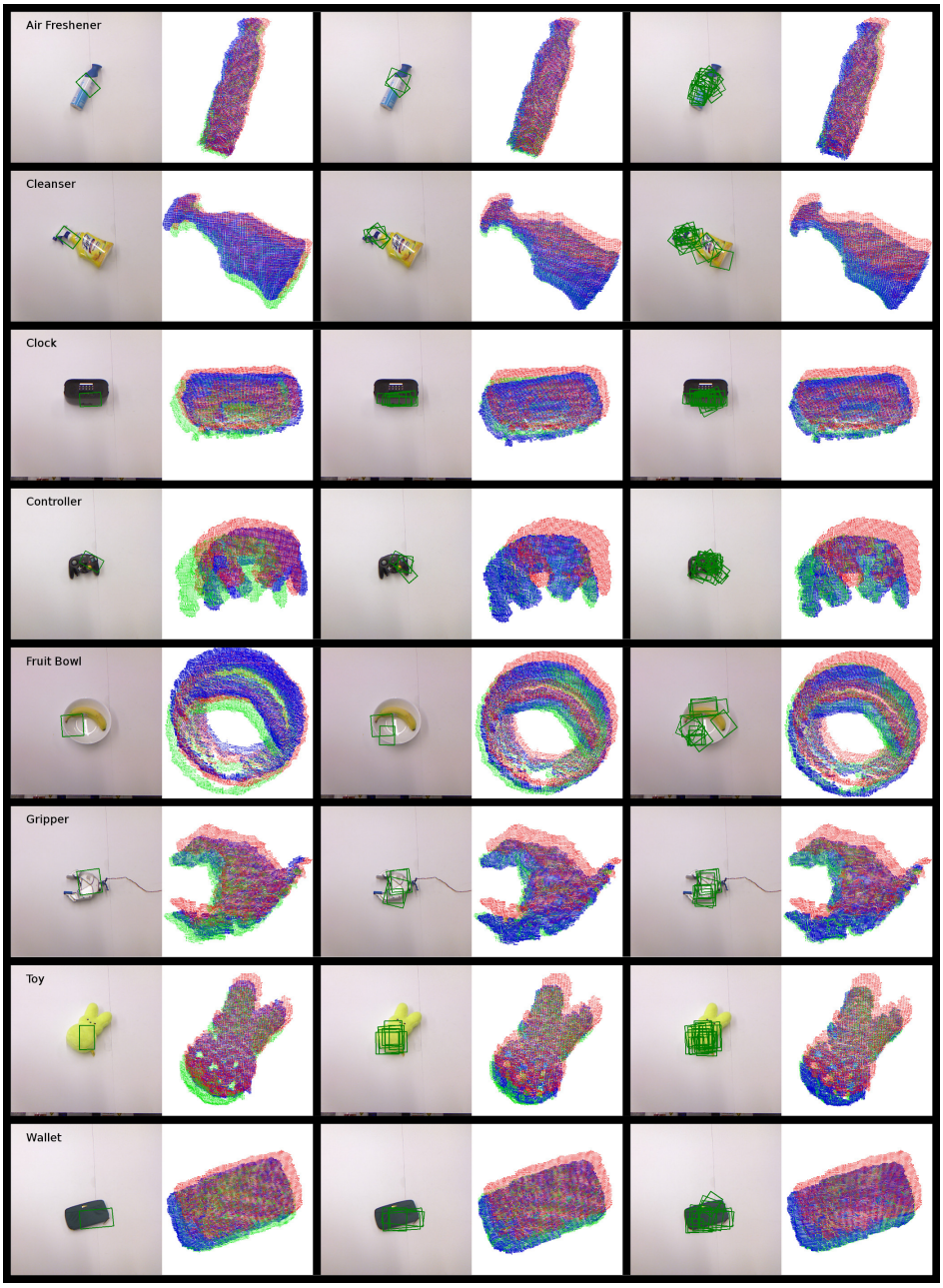


Figure 4.5: The matching results throughout the sequence of regions scanned by the pre-touch sensor for each object. For each pair, the left image indicates the regions to be scanned with green rectangles, while the right image displays the result of performing the scans. Three clouds are shown. Red represents the original Kinect data, blue represents the alignment estimated using the scans up to that point, and green represents the ground-truth alignment. The left-most pair corresponds to a single scan, the middle pair corresponds to a few scans, and the right-most pair corresponds to the point at which further scans provided no significant improvement in pose accuracy.

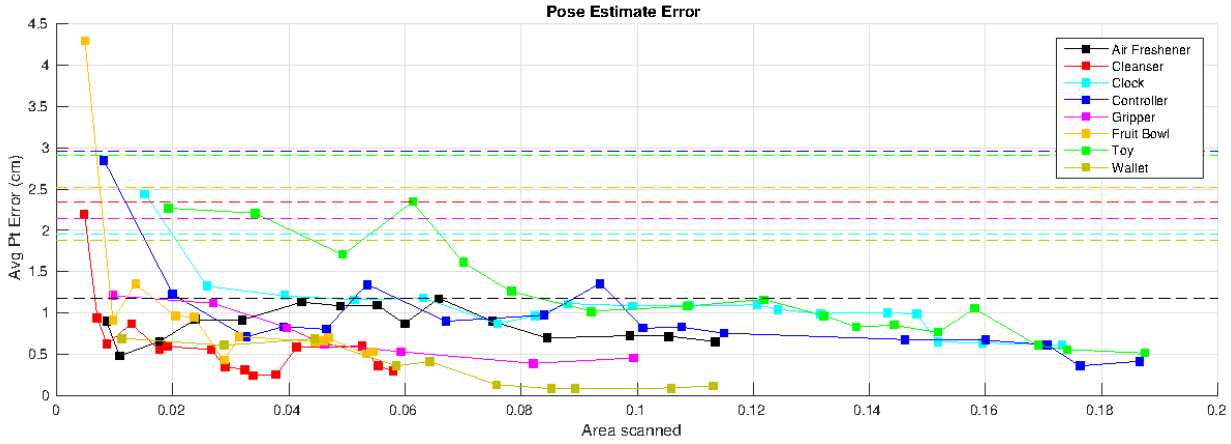


Figure 4.6: The pose estimate error after each pre-touch scan. The y-axis is the average distance between the ground truth alignment's points and the estimated alignment's points, and the x-axis denotes the percentage of the object that has been scanned. Each object corresponds to a different color, and each square represents a scan. The correspondingly colored dashed lines for each object represent the average distance between the points of the entire ground truth alignment and the points of the whole original Kinect point cloud.

approximately 0.5 cm after scanning less than twenty percent of the object. Some objects, such as the gripper, wallet, and cleanser achieved this accuracy after scanning less than ten percent of the object. Finally, for most of the objects, the pose estimation improves from the Kinect-based pose estimate after only a single scan. The sequence of region detections and pose estimations for each object is shown in Fig. 4.5. Most of the regions proposed by the network seem reasonable. It is especially promising that when examining the cleanser, the first region proposed by the network was the geometrically unique dispersal end of the object. Fig. 4.5 further illustrates how a robot could strategically choose how much time it spends scanning. If the robot can only afford a small amount of actuation time, a single scan could be used to most likely improve the pose estimate. If more time is available, the robot can likely achieve a highly accurate pose with just a few scans. On the other hand, if a high amount of accuracy is absolutely required, the robot can do many small scans while

avoiding an exhaustive scan of the whole object.

4.5 Discussion

We presented a framework for combining pre-touch sensing with deep learning to improve object pose estimation. By using a deep neural network to detect regions that are rich in geometric features, we can obtain enough information to re-estimate the pose without scanning the whole object. In fact, we found that a single scan was often sufficient to improve the estimate. Our region detection network outperformed two baseline methods, and can typically estimate the pose of an object to within 0.5 cm after a sequence of scans. The network is trained using a new dataset that contains minimal human bias and is applicable to any pre-touch sensor that gathers geometric information. This dataset has been made publicly available.

In future work, we would like to improve the accuracy of our method and integrate it into manipulation systems. With respect to accuracy, in this work, our network predicted regions that were individually well suited for pre-touch scanning. It is possible that a model that has memory or explicitly considers a sequence of regions could have better performance, such as a recurrent neural network. We also intend to examine the features learned by our model in more detail, as well as compare them to the features learned in other robot tasks, such as object detection and recognition.

Chapter 5

IMPROVED PROXIMITY, CONTACT, AND FORCE SENSING VIA OPTIMIZATION OF ELASTOMER-AIR INTERFACE GEOMETRY

In this chapter, we focus on improving the capabilities of the fingertip embedded sensors used in Chapters 3 and 4. While these previous sensors allowed the robot to estimate the pose of a manipulation object before making contact, their ability to measure objects once contact has been made is limited. Specifically, measurements based on the transmission and reception of infrared light from two separate sites imposes a minimum sensing distance, which prevents the sensor from accurately measuring the distance of an object within this distance threshold. To prevent the object from getting closer than the minimum sensing distance, we explore the placement of a transparent elastomer on top of the sensing module. The transparency of the elastomer allows our sensor to measure objects in the pre-contact regime, and the compressive properties of the elastomer establish a relationship between the measured distance and applied force in the post-contact regime.

Combining optical sensors with transparent elastomers provides a mechanism for robot manipulators to measure proximity, contact, and force. We propose an elastomer geometry that prevents undesired, internally reflected light from reaching the sensor's receiver in order to optimize contrast-to-noise ratio. We evaluate and compare this configuration's performance to that of two other elastomer geometries. We then demonstrate that this sensor endows the robot with the precise manipulation skills necessary to unstack a tower of blocks.

5.1 Proximity, Contact, and Force Sensing for Robot Manipulators

Endowing robots with fingertips that are sensitive to proximity, contact, and force (PCF) provides them with sensor feedback during critical moments of manipulation. Proximity data helps the robot localize an object, while contact and force data allows the robot to adjust how it is influencing an object, such as loosening its grip, pushing forward harder, or detecting slip. PCF sensors avoid the pitfalls of occlusion, unintentional object displacement, and non-zero sensing range that characterize head-mounted sensors, tactile-only sensors, and proximity-only sensors respectively. Recognizing combined PCF sensing as a relatively new concept, we focus on developing methods to further improve its capabilities and thereby its utility in robot manipulation.

We choose to focus on the optical sensing modality because it produces accurate measurements over a wide range. In particular, Patel et al. [2017] describe an innovative design for an optical PCF sensor. They place a clear elastomer over an optical proximity module. The ranging module can measure the distance to a nearby object because the elastomer is clear. To infer the force applied by an object in contact with the sensor, the elastomer is modelled as a spring that has been compressed by the measured distance.

While Patel et al. [2017] consider a number of design parameters, we explore other directions that further optimize the performance of this type of PCF sensor. First, our sensor uses time-of-flight (in addition to intensity) of reflected infrared light to measure distance. This allows our sensor's proximity measurements to be invariant to object surface reflectivity, which is a drawback that Patel et al. [2017] suffered from. Another area in which we improve upon Patel et al. [2017] is that their implementation operates in two separate modes. When measuring force, an electrical current configuration that optimizes sensitivity is used. However, when measuring distance, the infrared light reflecting from the elastomer-air boundary reaches the receiver and decreases the signal to noise ratio. We hypothesize that the second configuration that uses a greater amount of current is required to achieve an acceptable signal to noise ratio. We avoid having two different configurations by optimizing

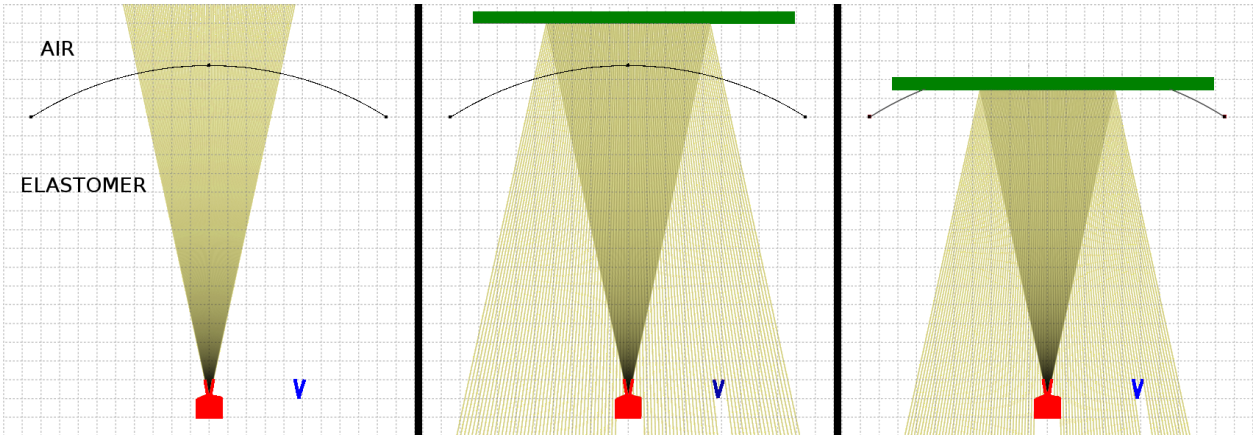


Figure 5.1: The path of infrared light emitted by the sensor when using a rounded elastomer-air boundary. The emitter is shown in red, the receiver in blue, the boundary in black, and a target object in green. All relevant metrics are to scale including the angle of light emission, transmitter and receiver aperture, curvature of the boundary, and spatial relationship between the transmitter, receiver and boundary. Each grid unit is equivalent to one millimeter [Tu, 2018]. Left: No object is present. While most of the light transitions through the boundary, some of it reflects off of the boundary and back into the transmitter. Middle: Object is in proximity. Right: Object makes contact and deforms the boundary.

the elastomer-air boundary to control the path of reflected light (Fig. 5.1).

In this work, we first propose the design of an improved PCF sensor that seamlessly transitions between measuring distances of up to 50 mm and contact forces of up to 10 newtons. We then demonstrate how the sensor can help the robot perform the delicate task of unstacking blocks. We provide publicly available hardware sources, software sources, and thorough instructions for fabrication of the sensor from inexpensive, commercially available components - which can all be found at https://bitbucket.org/opticalpcf/.

5.2 Improved Proximity and Force Sensing

Conventional reflective optical proximity sensors are severely affected by the surface properties of target objects because they directly measure the intensity of received light. Furthermore, while installing a clear elastomer over top of a proximity sensor establishes a mechanism for measuring force, infrared light that reflects from the elastomer-air boundary into the receiver reduces the signal-to-noise ratio. We employ a ranging module that can measure the time-of-flight of infrared light to remove the effect of surface reflectivity. We also design the geometry of the elastomer-air boundary in order to control the path of internally reflected light. In this section, we first characterize both the time-of-flight and the intensity outputs of the ranging module. Then we detail the theory and practice of how the elastomer-air boundary can be optimized to guide internally reflected light away from the receiver.

5.2.1 Time-of-Flight Sensor Operation

Our sensor uses the STMicroelectronics VL6180X Proximity Sensing Module to measure the distance travelled and intensity of emitted infrared light. Note that for all collected data, the calibration procedure suggested by the sensor module manufacturer was followed. The target distance for the offset calibration was at the top surface of the sensor, and the target distance of the cross-talk calibration was 50mm away from the top surface of the sensor. Fig. 5.2 illustrates the cone of emitted photons and the portion of them that are reflected from the object at distance d and back into the receiver. The half field-of-view angle θ is the same value for both the emission and reception cones, causing them to have the same radius r. The emitter and receiver are separated by a distance s.

Distance is estimated by measuring the average time that it takes for emitted photons to reflect off of a target object and then return to the receiver. Fig. 5.3 shows distance measurements for different colored targets - dark, light and in-between shades of red, green, and blue. The measurements are fairly linear with respect to the true distance, and typically

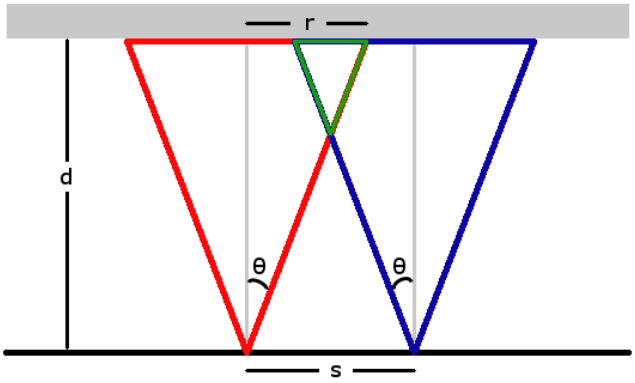


Figure 5.2: A side view of the sensor. The emitter produces a cone of photons as illustrated in red. The receiver cone is illustrated in blue. Their intersection forms the green viewing cone of the sensor, which defines the portion of photons that reflect off of the gray target object and into the receiver.

have an error on the order of a few millimeters.

The sensor simultaneously provides an estimate of the intensity of the reflected infrared light by measuring the rate at which photons are received. This phenomenon can be modelled by considering the geometry of the interaction between the sensor and target object. Intensity will be directly proportional to the ratio of the number of photons received to the number of photons emitted. This ratio is proportional to the ratio of the area of the object within the viewing cone A_{view} (i.e. the intersection of the emitter and receiver cones) to the area of the object within the emitter cone A_{emit}:

$$A_{view} = \tfrac{s}{2}\sqrt{4r^2 - s^2}$$

Given that $A_{emit} = \pi r^2$ and $r = d\tan(\theta)$, we obtain:

$$\hat{I} \propto \frac{A_{view}}{A_{emit}} \propto \frac{s}{2}\frac{\sqrt{4d^2\tan^2\theta - s^2}}{\pi d^2\tan^2\theta}$$

By combining constants and adding an offset term, we obtain the following expression:

$$\hat{I} = \kappa\frac{\sqrt{\alpha(d-\beta)^2 - \zeta^2}}{(d-\beta)^2} + \chi$$

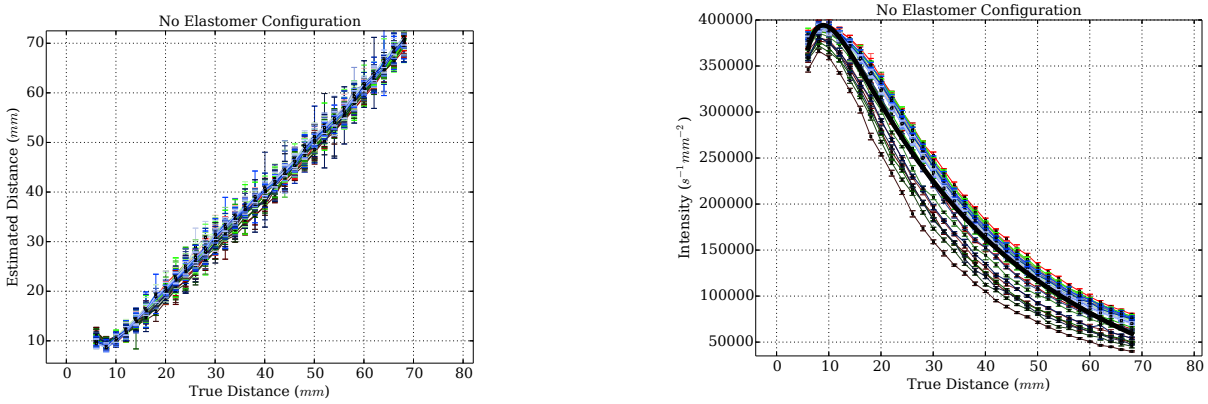


Figure 5.3: Measurements from the VL6180x optical sensing module. The color of each series corresponds to the color of the target. Left: The range output based on optical time-of-flight measurements. Right: The intensity (i.e. photon flux) output. The overlaid thick, black series is the result of fitting the derived model to the data.

Using the same targets previously mentioned, intensity measurements and a fitted model are shown in Fig. 5.3. We observe that the data fits the derived functional form above.

5.2.2 Elastomer-Air Boundary Optimization

If not treated carefully, light reflecting from the elastomer-air boundary will become a source of interference. Fig. 5.4 illustrates this effect when a naiive flat geometry is used. One solution to prevent infrared light from reaching the receiver is to simply make the distance between the sensor and the boundary smaller. However, doing so would lead to insensitivity in the force regime because no light would be able to reach the receiver. Fig. 5.4 presents two other possible solutions. First, one could place a physical blocker between the emitter and receiver. Alternatively, the geometry of the boundary could be shaped into a circular arc to reflect light back into the transmitter, and therefore away from the receiver. In order to best focus the reflected light into the transmitter, the transmitter should be placed at the focus of the circular arc, i.e. one radius away from the boundary. This distance d defines

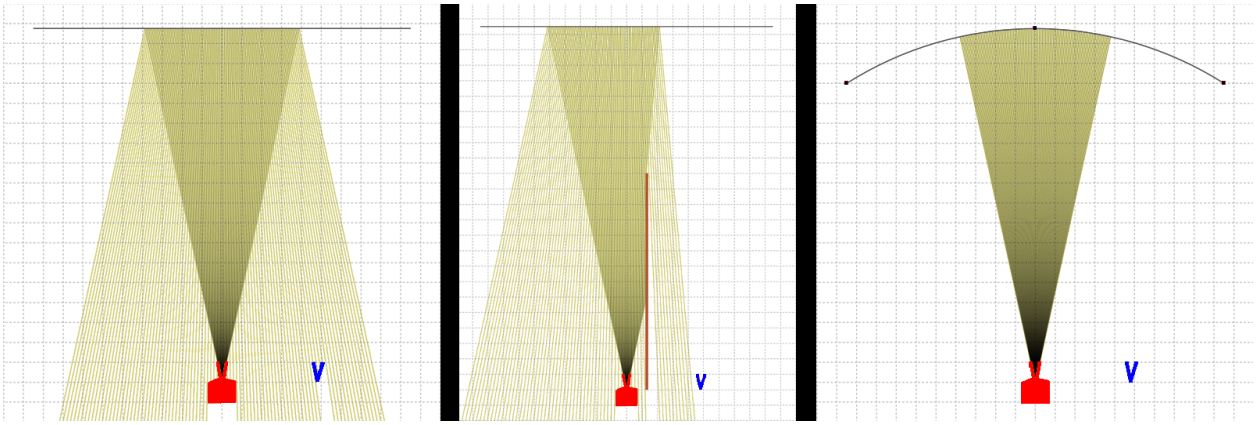


Figure 5.4: The path of emitted light rays for each of the examined elastomer-air boundary configurations. Note that only reflected rays are shown, refracted rays are omitted for clarity. The emitter is shown in red, the receiver in blue, the boundary in black, and the blocker in brown. All relevant metrics are to scale including the angle of light emission, transmitter and receiver aperture, curvature of the boundary, and spatial relationship between the transmitter, receiver, and boundary. Each grid unit is equivalent to one millimeter [Tu, 2018]. Left: Flat configuration. Middle: Blocker configuration Right: Circular arc configuration.

the thickness of the elastomer. The remainder of this section will detail the fabrication of the circular arc configuration.

Our sensor is fabricated by first 3D printing a case that can be mounted to the WSG-50 Weiss gripper. We place the case and electronics into a mold as shown in Fig. 5.5. Note that we print the mold such that its main axis is vertical so that the circular geometry is not limited by the layer height. The thickness d of the elastomer should be chosen to correspond to the operation region in which the ranging module is most sensitive. We found that a thickness of 17.75 millimeters works well for both the flat and circular arc geometries. We increase the thickness for the blocker configuration to approximately 23.5 milimeters so that light can still reach the receiver when the boundary is deformed. The elastomer is formed by pouring polydimethylsiloxane (PDMS) into the top of the mold. We use a mixing ratio

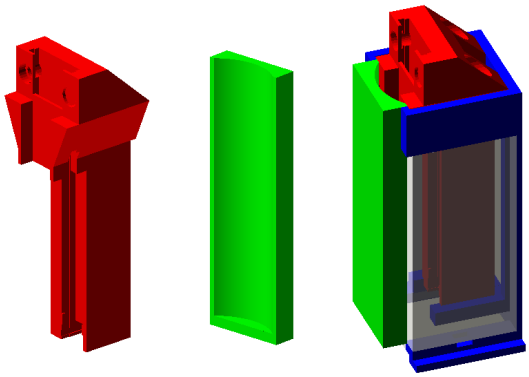


Figure 5.5: Visualizations of the sensor casing and mold. Left: Case for mounting sensor to Weiss gripper. Middle: Portion of mold that forms the circular arc elastomer-air boundary. Right: The assembled mold. Transparent planes correspond to pieces of acrylic.

of 30:1 to achieve a low Young's modulus, i.e. an elastomer that is more easily compressed by applied force.

Achieving an elastomer-air boundary that is optically clear is paramount to maintaining a high signal-to-noise ratio in the proximity ranging regime. This requires that the part of the mold that determines the boundary be smooth. The flat configuration, blocker configuration, and the implementation from [Patel et al., 2017] use a flat piece of acrylic for this part of the mold to achieve an optically clear boundary. However, when the elastomer-air boundary requires a curved geometry, obtaining optical clarity in an inexpensive way is more difficult. Simply 3D printing this section of the mold will produce unsatisfactory results because its layer-wise nature will cause unsmoothness. This is demonstrated in Patel et al. [2018], in which the work of Patel et al. [2017] is adapted for a Kinova gripper. The resulting elastomer-air boundary is opaque compared to the original implementation.

To manufacture an optically clear rounded elastomer-air boundary, we apply a number of post-processing steps after 3D printing the mold. We apply Smooth-On XTC3D Epoxy to this portion of the mold. Specifically, we apply two coats of the epoxy to the mold, sand it with 400 grit sandpaper, and then apply one more coat of epoxy. The resulting surface is

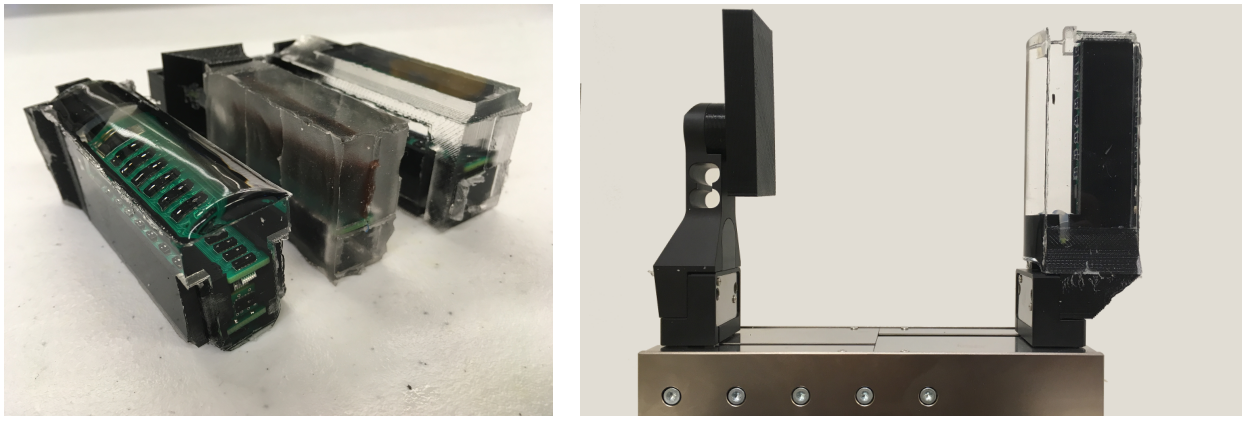


Figure 5.6: The evaluated sensors and measurement setup. Left: Three proximity-contact-force sensors designed for use with the Weiss Hand. Each has a different elastomer geometry, with the rounded configuration on the left, blocker configuration in the middle, and flat configuration on the right. Right: The Weiss Force Measurement Finger attached to the left finger of the Weiss hand and the sensor with rounded configuration attached to the right finger. The sensor's outputs are measured as the gripper width is varied.

sufficiently smooth to produce an optically clear boundary.

5.3 Experiments

We compare the performance of the flat, blocker, and circular-arc configurations in both the proximity and force measurement regimes. As shown in Fig. 5.6, the experimental setup consists of the sensor mounted to one finger of the Weiss hand, while the Weiss Force Measurement Finger (FMF) is mounted to the other. Different colored targets are attached to the plane of the FMF finger in order to measure the sensor's sensitivity to surface reflectance. The targets include dark, light (17% and 85% lightness in the HLS color space respectively), and in between shades of red, green, and blue. Ground truth distance is provided by the hand's motor encoder, which has a resolution of 0.1 mm. The FMF finger provides ground truth forces with plus or minus three percent accuracy.

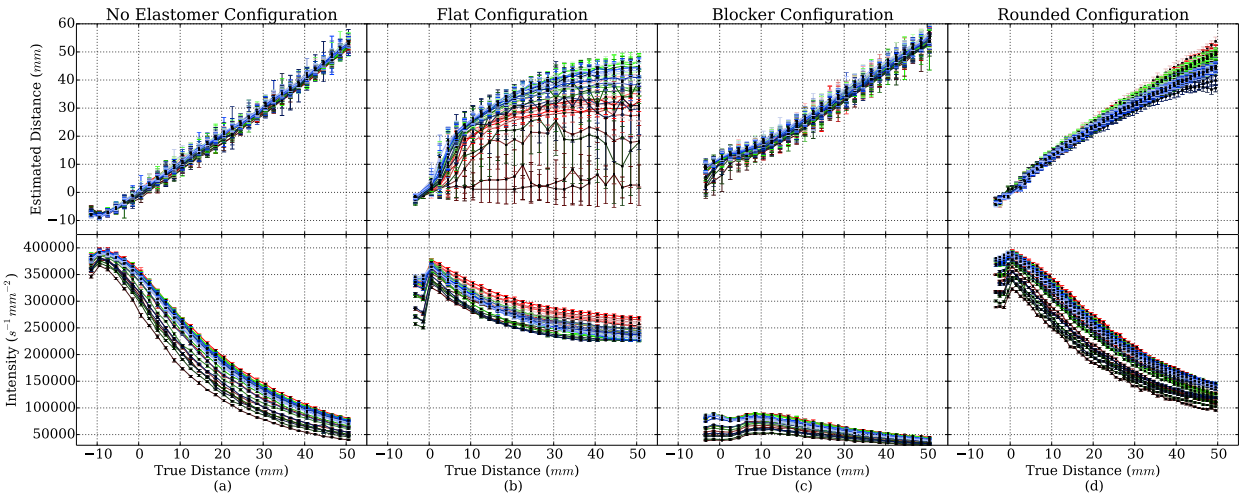


Figure 5.7: Range (upper row) and intensity (lower row) outputs for each configuration in the proximity regime. The zero point corresponds to the point at which an object makes contact with the sensor. The color of each series corresponds to the color of the target. (a) Output of VL6180X sensor (no elastomer). (b) Output of flat configuration. (c) Output of blocker configuration. (d) Output of rounded configuration.

5.3.1 Proximity Sensing

We first analyze the performance of all three configurations in the proximity regime. Fig 5.7(a) illustrates the performance of the sensor when it is not covered by elastomer. This represents the best performance that we could hope to achieve because placing elastomer on top of the sensor will only degrade the emitted signal and cause interference. In Fig. 5.7(b), we can see that the flat configuration sensor's range output is highly non-linear and influenced by surface reflectivity, and its intensity output has less dynamic range. In Fig. 5.7(c), the blocker configuration's range output has accurate, linear behavior throughout most if its range, but becomes non-linear in the region approximately one centimeter before contact. Also, its intensity output is relatively flat. The rounded configuration in Fig. 5.7(d) is also linear and accurate up until the four to five centimeter range at which darker colors

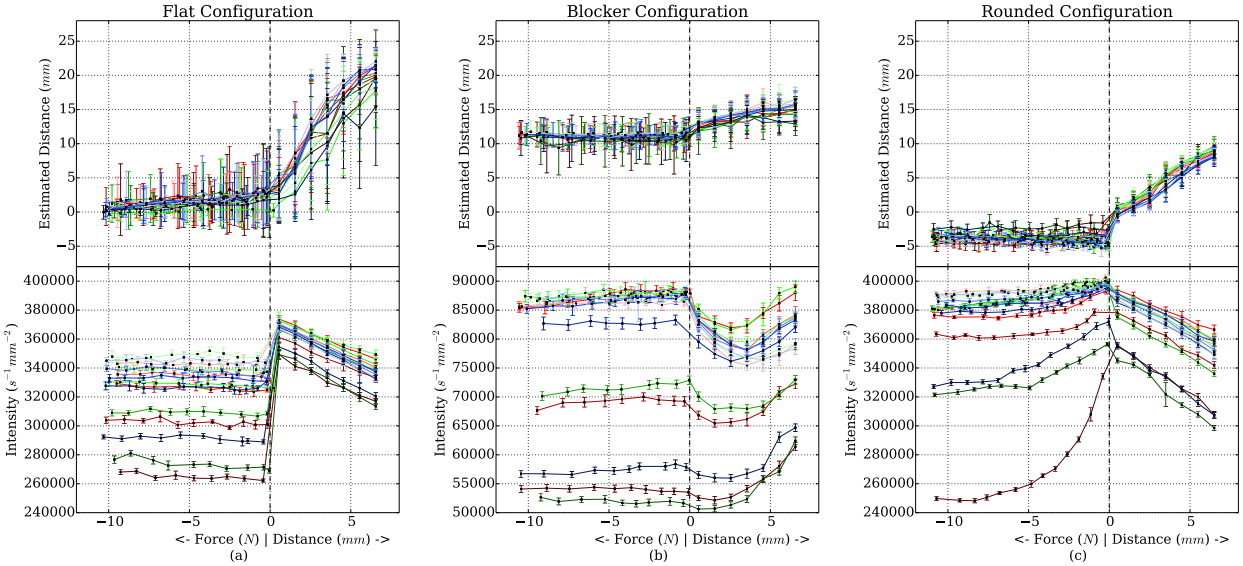


Figure 5.8: Range (upper row) and intensity (lower row) outputs for each configuration in the force regime. The zero point (denoted by a vertical dashed line) corresponds to the point at which an object makes contact with the sensor. The color of each series corresponds to the color of the target. (a) Output of flat configuration. (b) Output of blocker configuration. (c) Output of rounded configuration.

result in less than linear behavior. Furthermore, its intensity measurements are comparable to that of the elastomerless configuration. The data suggest that of the three configurations, the rounded configuration performs best. In particular, this configuration is preferable to the blocker configuration because it is more likely that accurate range measurements will be required just prior to contact rather than when objects are far away.

5.3.2 Contact Detection

To detect contact, we focus on the range output shown in Fig. 5.8. All three configurations generally decrease as the sensor transitions from the pre-contact regime to the post-contact regime. This decrease in the range output is most dramatic and the least noisy for the

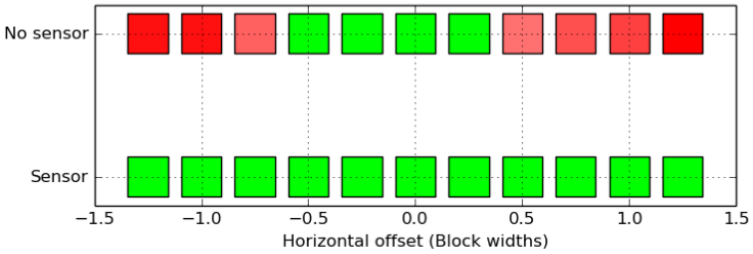


Figure 5.9: Green indicates that the robot successfully unstacked all of the blocks at the corresponding position with the specified method. Red indicates failure, where darker shades indicate that fewer blocks were successfully unstacked. One block width is one inch. Each stack consists of eight blocks.

circular-arc configuration. The flat configuration also has the potential to detect contact well based on the intensity data shown in Fig. 5.8(a). However, it may be limited by the intensity output's sensitivity to target reflectivity.

5.3.3 Force Sensing

As shown in Fig. 5.8, the range output for all of the configurations is relatively flat and/or noisy throughout the force regime. Therefore, force measurements will primarily be derived from the intensity output. The intensity values for the flat (Fig. 5.8(a)) and blocker (Fig. 5.8(b)) configurations are both flat. In contrast, the circular arc configuration (Fig. 5.8(c)) monotonically decreases as more force is applied. We also observe that its sensitivity depends on the reflectivity of the target, where less reflective targets result in greater sensation.

5.3.4 Grasping Evaluation

After finding that the sensor with a rounded elastomer-air boundary is the best performing configuration in both the proximity and force regimes, we demonstrate its utility for robots operating in uncertain environments. Here, the robot must deconstruct a stack of eight one-inch cubes that has been offset from its expected location. As shown in Fig. 5.9, we examine

how the robot performs both with and without using sensor feedback with the stack placed at various horizontal offsets. Here, it is assumed that the robot knows the size of the cube. When not using sensor feedback, the robot's performance degrades as the offset grows, with the most typical failure being one finger pushing a block off of the stack before the other finger can make contact. By using the sensors mounted to its gripper, the robot can servo its gripper to be centered around the block and robustly unstack it. For each cube, the robot executes a pregrasp in which the stack is between its fingers, and then adjusts the pregrasp based on the range outputs of the two fingers. Specifically, the smaller of the two fingers' measurements is used to adjust the gripper so that the corresponding finger is a target distance away from the cube (where the target distance is the distance that would center the gripper around the cube). The robot then adjusts the width of its gripper to be slightly less than the dimension of the cube in order to grasp and remove it from the stack.

5.4 Discussion

One limitation of the sensor is that some measure of the reflectivity of the target must be known in order to correlate an intensity measurement with an observed force. In future work, we intend to characterize the surface reflectivity of the target in the proximity regime, and then use this information in the force regime to make accurate force measurements. The robot could first measure the object at some distance to obtain a range measurement and an intensity measurement. Based on the data in Fig. 5.7, we believe that this pair of data is sufficiently discriminative to characterize the target's surface properties. Using this information, the sensor can lookup the force that corresponds to a measured intensity value once the target has made contact. With this capability, future work will focus on how this sensor allows the robot to carefully position its end effector before using force control to manipulate an object.

Chapter 6

ELECTROSTATIC BRAKES ENABLE INDIVIDUAL JOINT CONTROL OF UNDERACTUATED, HIGHLY ARTICULATED ROBOTS

In this chapter we begin examining this thesis' contributions to alternative forms of actuation for dexterous manipulation. Most forms of robot dexterous manipulation warrant the use of high degree-of-freedom mechanisms in order to facilitate considerable range of motion, but actuating a large number of joints with electromechanical motors can cause the robot to be unacceptably heavy, bulky, or power consumptive. We discuss designs for equipping underactuated robots with electrostatic brakes. Such an actuation strategy provides a foundation for hybrid actuated robots that have a weight, size, and power consumption similar to underactuated robots, but have significantly more dexterity in that they can reach any desired joint configuration.

Here, we propose a joint design that optimizes brake electrode conformance. We demonstrate that this brake-equipped joint design is able to achieve the significant braking strength predicted by its theoretical model. We compare our joint to those powered by electromechanical motors, and highlight the relative reductions in mass, volume, and power consumption that electrostatic brakes can achieve. We then use this joint design to control a ten degree-of-freedom, single servo motor robot and demonstrate its ability to cage and manipulate multiple objects simultaneously.

6.1 Hybrid Brake-Motor Actuation

Many organisms rely on the highly articulated nature of their bodies to perform vital functions. For example, human hands leverage twenty-one degrees-of-freedom (DoF) in order

to nimbly manipulate objects [Jones and Lederman, 2006]. Snake spinal columns composed of up to three hundred vertebrae facilitate versatile locomotion through their environments [Ma, 1999]. Lacking a rigid skeleton, octopus tentacles can use a seemingly unlimited number of degrees-of-freedom to both manipulate and locomote [Mazzolai et al., 2007]. While immense articulation affords flexibility of movement, it is the networks of incredibly high strength to weight and volume ratio muscle that generate the forces necessary for such maneuvers [Rospars and Meyer-Vernet, 2016]. Roboticists have attempted to solve highly dexterous tasks by building biologically inspired mechanisms with many degrees-of-freedom [Tsai, 1995, He et al., 2019, Billard and Kragic, 2019, Transeth et al., 2009], but their success has been limited by the inability of modern electromechanical actuators to simultaneously match both the strength and light weight of biological muscle. By designing an electrostatic brake with holding force that can be scaled to that of muscles yet remain similarly lightweight, we aim to build highly articulated, hybrid-actuated robot mechanisms with dexterity that approaches that of their biological counterparts.

When limited to conventional options for electronically controlled actuation, the most common architecture is to employ one or more motors per joint in order to achieve full actuation [Tsai, 1995, He et al., 2019]. However, applying this actuation scheme to mechanisms with a high number of degrees of freedom often violates volume, mass, power consumption, and cost constraints [Billard and Kragic, 2019, Transeth et al., 2009]. To satisfy these constraints, roboticists tend to reduce the number of motors by coupling multiple joints together through the use of tendons (Fig. 6.1). However, this severely degrades the space of joint trajectories that the mechanism can execute because direct control of individual joints is no longer possible [Ma and Dollar, 2017, Gosselin et al., 2008, Grioli et al., 2012, Xu et al., 2012].

Positional control of each individual joint can be restored to such tendon-driven systems by installing a brake into each joint. This allows torque exerted on a joint by a motor-actuated tendon to be counteracted by engagement of the joint's brake. Here, it is assumed that each brake can only be in one of two states (on or off), and that, when engaged, the applied

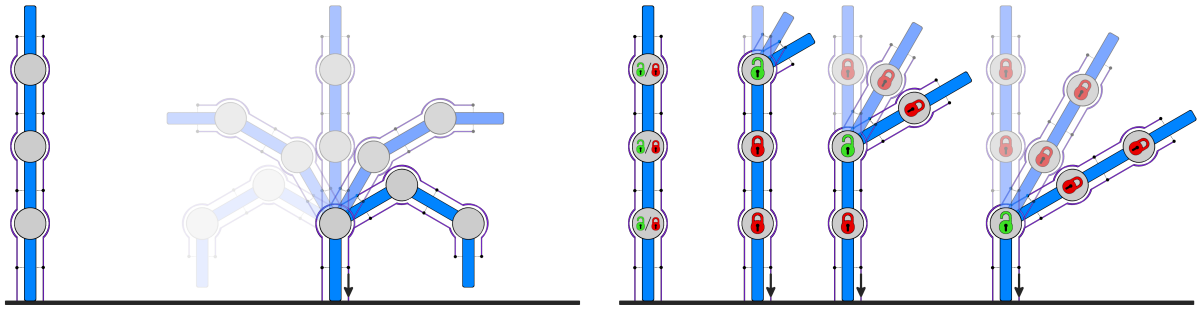


Figure 6.1: Installation of brakes into the joints of underactuated robots enables positional control of individual joints. Tension applied to the tendons exerts torque on all of the coupled joints but engaging a joint's brake prevents it from moving. (Left) The range of motion of conventional underactuated systems is limited to the trajectories that occur when the tendons are pulled or released. (Right) By engaging any combination of brakes, a much wider range of trajectories can be achieved.

braking torque entirely negates the torque applied by the tendon on the corresponding joint. Thus, by selectively choosing which brakes are engaged, joints that are connected to the same tendon can have their motion decoupled (Fig. 6.1). This actuation strategy can be applied to any kinematic structure in which the joints are coupled together by one or more tendons [Jacobsen et al., 1989, Koganezawa and Yamazaki, 1999]. Although the mechanism is still underactuated, its dexterity has been significantly improved because the restoration of control over individual joints allows it to reach arbitrary joint configurations.

Electrostatic brakes offer a promising pathway towards the implementation of dexterous brake-aided systems. Although the muscles that power dexterous movement in biological organisms use different physical mechanisms for force generation than electrostatic brakes, the metric of specific tension (i.e., force per cross-sectional area) that has classically been used to measure muscle strength is also naturally applied to electrostatic brakes. Muscles typically have maximum specific tension exertion capability on the order of two hundred

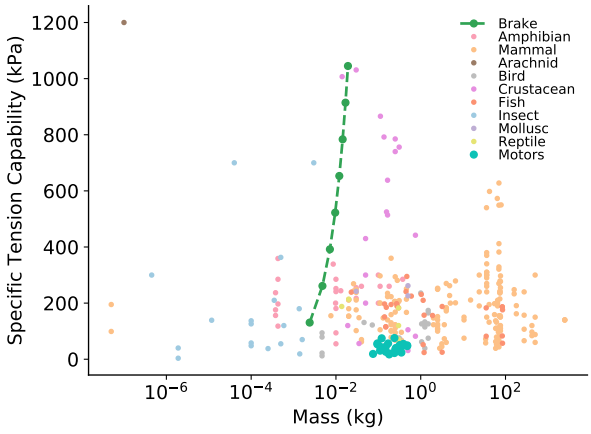


Figure 6.2: The expected frictional force per cross-sectional area (i.e. specific tension) of our electrostatic brakes is compared to that of measured biological muscle [Rospars and Meyer-Vernet, 2016] across multiple taxonomic groups. Braking force can be scaled by stacking sets of thin electrodes on top of each other (up to 8 stacked sets shown). Please see the beginning of Section 6.4 for brake parameter values.

kilopascals across a wide range of taxonomic groups [Rospars and Meyer-Vernet, 2016], but electrostatic brakes can achieve similar or higher specific tension while having a lower mass than most types of muscle (Fig. 6.2). This is particularly evident for brakes that stack two or more sets of electrodes on top of each other. Electromechanical motors are significantly heavier and, although they may be able to exert comparable amounts of raw force/torque, their relatively large cross-sectional areas result in worse specific tensions in comparison to electrostatic brakes and most types of muscle.

Relative to other braking methods, electrostatic brakes are more compact, lightweight, and power-efficient [Plooij et al., 2015, Guo et al., 2019]. Additionally, a common limitation to braking mechanisms is the amount of time required to engage or disengage the brake [Kontoudis et al., 2015, Do et al., 2020, Firouzeh et al., 2017, Aukes et al., 2014]. While electrostatic brakes can also suffer from hysteretic effects when driven by a direct cur-

rent source, alternating current methods allow them to engage and disengage on the order of milliseconds [Hinchet and Shea, 2020]. While the need to generate sufficient electrostatic force has required a number of previous electrostatic brake implementations to have large size or material costs [Hinchet and Shea, 2020, Hinchet et al., 2018, Karagozler et al., 2007, Diller et al., 2016], stacking multiple thin brakes on top of each other facilitates compact, inexpensive, and effective braking.

Conformance between electrodes is necessary for achieving the full strength of electrostatic brakes. Thick electrodes lack the compliance necessary to properly conform to one another. Thin electrodes will buckle if they are not tensioned along the direction of an applied force, or incidentally unzip if out-of-plane forces are applied at the electrodes' edges. Our insight is that we can optimize thin electrode conformance by employing a rack-and-pinion transmission to convert rotational motion of the joint into linear sliding between the electrodes. Controlling the motion of the electrodes to be along a single linear axis eliminates the out-of-plane forces that cause unzipping, and makes it easy to tension the electrodes along the direction of motion in order to prevent buckling. Carefully choosing the thickness and spacing of the electrodes allows this optimal conformance to be extended to multiple stacks of electrodes. By optimizing electrode conformance and stacking thin electrodes, we can achieve significant braking strength without increasing brake area or employing expensive, specialized dielectrics.

We transform this insight for optimal brake conformance into a novel joint design. With this joint design, we aim to demonstrate that hybrid motor-brake actuation enables robot operation in scenarios that not only require dexterity, but also place constraints on the robot's weight, size, and/or power consumption. To that end, we first measure the braking capabilities of this design in order to verify that it achieves the significant braking capability predicted by its theoretical, ideal-electrode-conformance model. Upon comparing our joint design to robot joints driven by electromechanical motors, we find that our brake equipped joint exerts similar holding torques yet is significantly more compact, lightweight, and power efficient. We then build a brake-equipped robot that demonstrates how our brake

design significantly enhances the dexterity of underactuated robots by enabling control of individual joints. This robot uses only a single motor and its brakes to actuate its ten degrees-of-freedom. Yet it is able to manipulate multiple objects simultaneously, a task that a conventionally underactuated robot counterpart would not be able to complete. A playlist of supplementary videos is available here: https://youtube.com/playlist?list=PLDzz4IzZazscc1xAZ5nRtCqaR8yYbkrMs

6.2 Electrostatic Force Generation

An electrostatic brake can be formed by bringing two conductors (or generally two bodies with conductors attached to them) in contact with a dielectric that separates them. When voltage is applied across the conductors, the resulting attractive force induces a frictional force that resists tangential motion between the conductors. Thus, electrostatic brakes may oppose relative motion between two bodies by inducing frictional forces between them.

The maximum braking force is primarily determined by the area of overlap between the conductors, applied voltage, separation distance, dielectric permittivity, and the conductor-dielectric coefficient of friction. However, achieving a desired braking force is limited or hindered by a number of factors such as the voltage tolerance of control electronics, reduced effective area of overlap due to poor electrode conformance, and cost of high permittivity dielectrics. In the following subsections, we formalize the modeling of electrostatic brake strength, discuss the design considerations necessary for achieving a desired braking capability, and provide motivation for a brake design that facilitates the stacking of brakes.

6.2.1 Parallel Plate Capacitor Model

The most basic brake structure consists of two conductive plates separated by an insulator. A controlled voltage applied to the brake creates an attractive force F_a between the plates, which can be modelled as a parallel plate capacitor (Fig. 6.3):

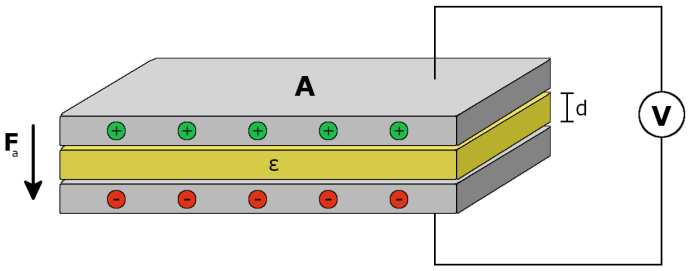


Figure 6.3: Electrostatic phenomena between two parallel conductors can induce a braking effect. In the absence of applied voltage, the two parallel conductors can exhibit planar sliding motion relative to each other. Once voltage is applied, an attractive force between the oppositely charged conductors induces a frictional force between them that prevents sliding.

$$F_a = \frac{\epsilon A V^2}{2d^2}, \tag{6.1}$$

where ϵ is the permittivity of the dielectric, A is the area of overlap, V is the applied voltage, and d is the distance between the plates. Along with the coefficient of friction between the plates and dielectric μ, F_a induces a frictional force $F_{max\,brake}$ that opposes sliding motion of the plates relative to each other:

$$F_{max\,brake} = \mu F_a. \tag{6.2}$$

6.2.2 Scalable Electrostatic Force

The quadratic relationship between braking force and voltage in eq. (6.1) encourages the use of high voltages in order to maximize braking capability. In practice, the applied voltage is limited either by the voltage tolerance of the control electronics, or the dielectric breakdown of the insulator. Generally, the voltage tolerance of the electronics increases with their size,

and the insulator's breakdown voltage is proportional to its thickness (which itself affects the maximum braking force). The amplitude of the applied voltage has practical limits, but there are other avenues of brake design that can further increase braking capability.

First, the proportional relationship between dielectric constant and braking force motivates the use of high relative permittivity insulators [Hinchet and Shea, 2020, Diller et al., 2016]. These materials are expensive and difficult to source, particularly in comparison to lower permittivity insulators such as polyimide and polyethyleneterephthalate (PET) films. Second, the braking force's inverse quadratic relationship with plate distance encourages the use of thinner dielectrics, but practical limitations include material availability, fabrication capability, mechanical robustness, and the resulting decrease in breakdown voltage threshold. Third, increasing the area of the electrodes (and their area of overlap) will yield greater braking capability, but will generally make the brake less compact.

Finally, electrode non-idealities can hinder braking strength. In particular, the effective area of overlap between the brake's electrodes is heavily affected by the degree to which they conform to each other during engagement. Conformance of thicker electrodes is limited by their lack of compliance. On the other hand, thinner electrodes with greater compliance are more susceptible to both unzipping and buckling. Unzipping occurs when out-of-plane forces peel the electrodes away from each other at the edges of contact, and buckling occurs when in-plane forces cause one or more electrodes to cave in on themselves [Karagozler et al., 2007, Diller et al., 2016]. Buckling can be prevented by applying tension to the electrodes when mounted to the brake substrate, and peeling can be mitigated by ensuring that each electrode is limited to motion along its plane. While it is possible to produce a braking effect between electrodes with relative rotational motion [Aukes et al., 2014, Johnsen and Rahbek, 1923], our own experience with alternative joint prototypes suggests that linear sliding best avoids electrode conformance issues. We found that limiting motion between the electrodes to be along a single linear axis eliminates the out-of-plane forces that cause unzipping, and buckling can be prevented by tensioning the electrodes along the direction of motion.

In formulating the design of our brake, we aim to maximize braking strength while

minimizing size and cost. While it has a low dielectric constant relative to other materials, we use PET films because they are thin, mechanically robust, and widely available. To minimize the area of our brake's electrodes, we stack multiple brake electrodes on top of each other according to the required maximum braking force. Since electrostatic brakes are constructed from thin materials, stacking them does not significantly increase volume nor weight. Finally, our brake design maximizes effective area of overlap by optimizing electrode conformance. As previously discussed, we achieve this by limiting motion of the electrodes to be along a single linear axis. The design of our brake and corresponding joint is further detailed in Section 6.3.

6.3 Electrostatic Braking for Robot Joints

Designing a robotic joint that incorporates an electrostatic brake presents a number of challenges. On top of the requirements for significant braking force generation, it is crucial that a brake-aided joint has consistent braking capabilities throughout its joint range. Achieving such consistency requires the spatial relationship between the conductors (particularly area of overlap, conformance, and separation distance) to remain constant throughout all possible positions of the joint. The design should also be amenable to stacking brakes so that the braking force can be scaled as necessary. Finally, the use of high voltages (on the order of 1 kV) to actuate the brake can cause time-dependent, undesirable effects that are not captured by the parallel plate capacitor model.

In the following subsections, we describe the design and operation of our brake-aided joint. Our novel electrostatic brake equipped joint converts rotational motion of the joint into linear sliding motion between the electrodes in order to maintain the optimal and consistent electrode conformance that is assumed by our electrostatic braking model (Sec. 6.2.1). We detail the materials and methods used to construct the joint. Finally, we discuss how injection of charge into the dielectric over time is detrimental to brake strength, disengagement, and re-engagement, and we describe our implementation of control circuitry that severely mitigates the effects of charge injection.

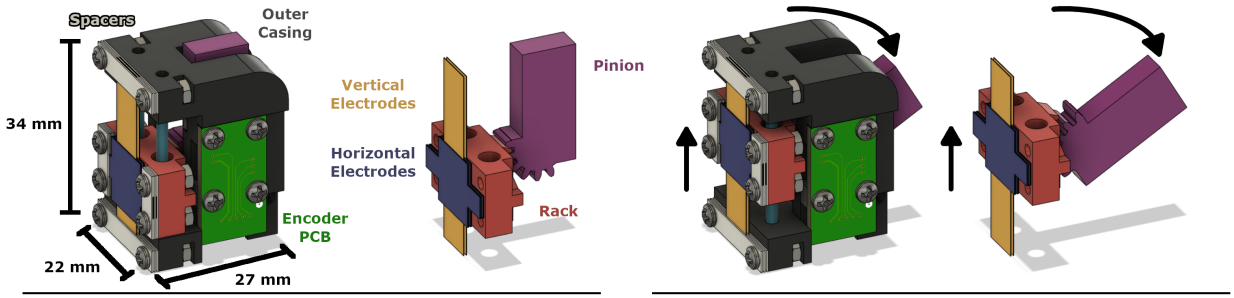


Figure 6.4: Rotational motion of the joint is transformed into linear sliding between the two sets of electrodes by a rack-pinion transmission system. (Left) The major subcomponents of the joint. Illustrations both with and without the outer casing are shown. (Right) Rotational motion of the pinion causes the two sets of brakes to linearly slide relative to each other. Conversely, blocking relative motion between the two sets of electrodes will prevent rotation of the joint.

6.3.1 Brake-Aided Joint Design

We first design a modular robot joint that demonstrates the potential of electrostatic brake-aided systems. Our joint uses a rack and pinion transmission system to convert the rotational motion of the joint into linear sliding motion between the brake's electrodes (Fig. 6.4). In our joint design, a set of electrodes mounted to the rack (i.e., the horizontal electrodes) are interleaved with another set of electrodes mounted to the joint's outer casing (i.e., the vertical electrodes). When the brake is not engaged, rotation of the pinion pushes the rack along stainless-steel rails that have been mounted to the outer casing of the joint. Here, the two sets of electrodes slide along each other with negligible frictional force. Once the brake is engaged, the resulting electrostatic force between the two sets of electrodes prevents them from sliding relative to each other. The rack-mounted set of electrodes thereby prohibits the rack from sliding along the rails, which in turn blocks rotation of the pinion.

We define a *stack* to be a vertical electrode sandwiched between two horizontal electrodes (Fig. 6.5). The vertical electrode is composed of a stainless-steel strip that uses

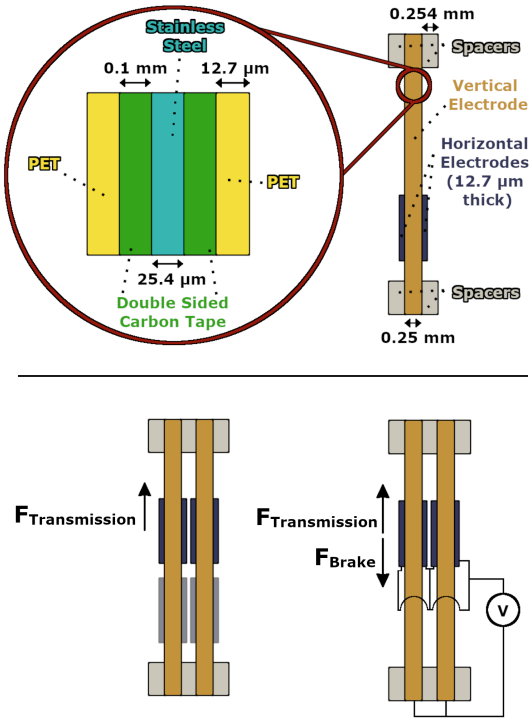


Figure 6.5: Construction of electrostatic brakes from thin, lightweight materials facilitates stacking in order to increase braking capability. (Top) Side view of a single stack of electrodes consisting of a vertical electrode sandwiched between two horizontal electrodes. (Bottom) Two stacks of electrodes. Spacers separate adjacent stacks of electrodes. (Bottom Left) When the joint rotates, a linear force causes the horizontal electrodes to slide along the vertical electrodes. (Bottom Right) Application of voltage induces an equal and opposite frictional force that prevents sliding between the electrodes and therefore rotation of the joint.

double-sided carbon tape to attach a dielectric sheet to each of its faces. Carbon tape is an adhesive solution that does not increase the effective thickness of the dielectric due to its conductivity [Hinchet and Shea, 2020, Hinchet et al., 2018]. Each of the horizontal electrodes is simply a strip of stainless steel. When stacking electrodes, we use spacers to control the spatial offset between adjacent stacks. Stacking electrodes is an effective method for scaling

Table 6.1: The volume, mass, and cost of the proposed electrostatic brake equipped joints. Row one corresponds to the prototype joint used when measuring holding torque capability. Row two corresponds to the updated joint used in the ten degree-of-freedom robot. Metrics for these two rows do not include the brake (electrodes and spacers) because the brake's volume, mass, and cost are dependent on the degree of stacking. Instead, the volume, mass, and cost of a single stack of electrodes and spacers is reported in the third row.

	Volume (cm^3)	Mass (g)	Cost (USD)
Prototype Joint	20.0	20.2	\$15.42
Updated Joint	25.4	24.9	\$16.18
Brake (per stack)	0.30	2.4	\$0.61

braking strength without adding significant volume, mass, and cost (Table 6.1).

6.3.2 Joint Construction

We construct a vertical electrode by first cutting out a 76.2 mm by 8 mm strip of 25.4 µm thick stainless-steel foil with scissors (Fig. 6.5). Both faces of the same end of this strip are then each covered by a 35 mm by 8 mm strip of double-sided conductive carbon tape. Films of 12.7 µm thick PET dielectric are then laid onto both faces of the vertical electrode such that they cover the carbon tape. Once adhered to the vertical electrodes, a hand-held rotary cutter is used to trim the PET films such that they each have a width of 12 mm. Any film that protrudes beyond the end of the vertical electrode should also be removed using the rotary cutter.

Horizontal electrodes consist of a 76.2 mm by 10 mm strip of 12.7 µm thick stainless-steel foil. Four square holes that align with mounting holes on the gear rack are cut out of the strip using an exacto knife. We also fold the horizontal edges of the electrode over itself to ensure that the edges are smooth. This smoothness prevents the edge of the horizontal

electrode from cutting into the dielectric layer of the vertical electrode during engagement.

The outer casing, pinion, and rack are 3D printed from the ABS filament of a Stratasys F120 3D printer. Both ends of the pinion are mounted to the joint's outer casing via two ball bearings. Four stainless steel washers are pressed into the plastic rack using a soldering iron. These washers have a nominal inner diameter of 0.094 inches. They facilitate linear sliding of the rack against two 0.09375 inch diameter dowel pins inserted into the outer casing of the joint.

Stacks of brakes are mounted to the back of the outer casing. Thin and rigid electrical wire is temporarily inserted into the four mounting holes of the rack and the four mounting holes at the top and bottom of the outer casing. These wires help hold the spacers and electrodes in place as they are being stacked on top of each other. A single stack of brakes is formed by first placing a pair of 11.5 mm by 4 mm strips of 0.254 mm thick stainless steel through the wires protruding from the rack and a pair of 21.7 mm by 4 mm strips of 0.254 mm thick stainless steel through the wires protruding from the top and bottom of the outer casing. These spacers are constructed using a hand-held sheet metal cutter and a 2.5 mm diameter metal hole punch. A horizontal electrode is then laid across the rack, followed by a vertical electrode across the top and bottom of the casing, and finally followed by another horizontal electrode across the rack. If this is the final stack of brakes, then the stack's final layer consists of another pair of spacers at the top and bottom of the outer casing and another pair of rack mounted spacers. If additional stacks of electrodes will eventually be added, then the stack's final layer consists of only a pair of rack mounted spacers. This process can then be repeated in order to add more stacks of brakes. Once any additional stacks of brakes are placed on top, 3D printed fasteners are slipped onto the protruding wires. The wires are then removed and replaced with M2 screws that are twisted into hex nuts embedded in the rack and outer casing in order to hold the stacks of electrodes in place.

Note that one layer of spacers separates adjacent vertical electrodes in order to ensure that there is space in between them for the horizontal electrodes to slide. The thickness of horizontal electrodes is negligible in comparison to that of vertical electrodes. Therefore,

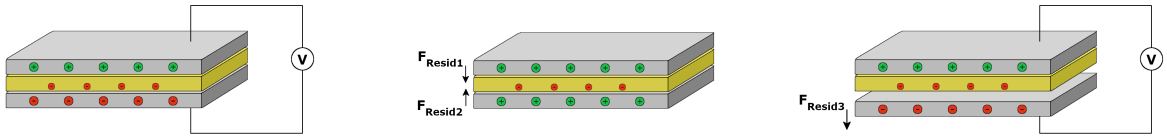


Figure 6.6: Charge injection is detrimental to the operation of an electrostatic brake. (Left) Application of high voltage for a sustained period across the brake's conductors causes electric charges to be injected into the brake's dielectric. (Middle) Injected charge remains trapped in the dielectric even once the voltage has been removed, preventing the brake from disengaging. (Right) Charges trapped in the dielectric also hinder re-engagement of the brake by generating a repulsive force.

by choosing stainless steel spacers with a thickness as close as possible to that of a vertical electrode, two layers of spacers between adjacent stacks of horizontal electrodes will result in an offset that is approximately equal to the offset between adjacent vertical electrodes. The closer these offsets are, the flatter the contact between the vertical electrode and horizontal electrodes within a given stack, which is conducive to electrode conformance and effective area of overlap.

6.3.3 Charge Injection

The application of high voltage to the brake during engagement significantly increases the amount of time required to subsequently disengage the brake [Aukes et al., 2014]. As shown in Fig. 6.6, when a high voltage is applied across the brake's plates, electrical charges are injected into the brake's dielectric and become trapped inside [Montanari, 2000, Dissado et al., 1997]. These stored charges induce a residual attractive force between the plates that prevents brake disengagement after the removal of voltage and hinders brake re-engagement once voltage is again applied. The magnitude and persistence of this residual force increases with the duration of the preceding applied voltage [Aukes et al., 2014]. This observation suggests

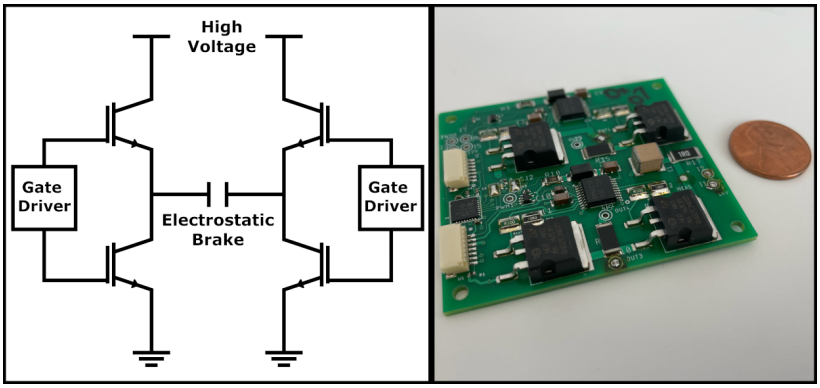


Figure 6.7: To mitigate charge injection, the polarity of the applied voltage must be periodically inverted during brake engagement. (Left) Schematic of the high voltage H-bridge circuit used to control the brake. (Right) The printed circuit board that implements the H-bridge circuit.

that the applied voltage signal during engagement take the form of a bipolar square wave. This waveform maintains the full voltage magnitude across the plates, but also periodically inverts the voltage polarity such that the effect of charge injection is made negligible [Hinchet and Shea, 2020, Hinchet et al., 2018].

6.3.4 Brake Driver Circuitry

In order to actuate the brakes, we designed a printed circuit board for generating high voltage bipolar square waves (Fig. 6.7). The circuit implements an H-bridge architecture to enable swapping of the voltage polarity across its two outputs. Our 1200V tolerant H-bridge is composed of four IGBT transistors controlled by two gate driver modules. Given a single joint, all of the joint's horizontal electrodes are connected to one output of the H-bridge, while all of its vertical electrodes are connected to the other output.

6.4 Brake-Aided Joint Performance

In this section, we evaluate the performance of our brake-aided joint and demonstrate how it can enhance the dexterity of highly articulated, underactuated systems without significantly increasing weight, volume, or power consumption. Here, we aim to quantify the conformance of the brake's electrodes throughout the range of the joint. Specifically, we measure the maximum braking torque that the brake-aided joint can exert, and compare it to the performance predicted by the ideal-conformance, parallel plate capacitor model. We expect that the observed maximum braking torque will increase linearly with the degree of stacking. We then compare these results to the holding torques of motors commonly used in highly articulated robots to establish that electrostatic brakes can serve as lighter, compact, and power efficient alternatives.

For the remainder of this section and those that follow, we use an applied voltage of 1000V, PET insulator with approximate permittivity of 3.35 [McMahon et al., 1959] and thickness 12.7 µm, and a stainless steel-PET coefficient of friction value of 0.71 [Mens and De Gee, 1991]. The only exception is that we used a nominal voltage of 950V when performing the multi-object manipulation task with our ten degree-of-freedom robot. We lowered the voltage because the output of our high voltage power supply is dependent on the load current. In the most extreme case of all brakes turning on or off at the same time, the output voltage could momentarily fluctuate by up to 100V. Lowering the nominal output voltage ensured that the voltage tolerance of our H-bridge circuit was not exceeded when these fluctuations occurred.

6.4.1 Electrostatic Brake Stacking

To verify the consistency and scalability of braking strength, we evaluate our joint's maximum holding torque throughout its joint range for an increasing number of stacked electrodes (Fig. 6.8). Measurements consist of applying increasing amounts of torque to the pinion until the engaged brake can no longer prevent relative motion between the two sets of electrodes,

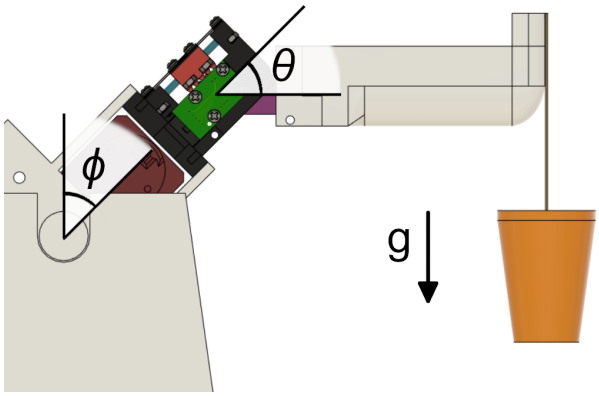


Figure 6.8: Visualization of the experiment platform that facilitates torque measurements throughout the joint's range. For a desired test setting of the joint angle θ, the base angle ϕ is chosen to orient the lever arm perpendicular to the direction of gravity. Torque applied to the joint is increased by adding known weights to the orange cup attached to the end of the lever arm.

causing the joint to rotate. The expected holding torques can be computed using the previous force equations and the rack-pinion equation:

$$T_{max\,brake} = \frac{1}{2} d_{Pinion} F_{max\,brake}, \tag{6.3}$$

where the pitch diameter of the pinion d_{Pinion} is 12 mm and each vertical electrode - horizontal electrode pair is expected to have an approximate area of overlap of 0.8 cm^2 throughout the joint's range of motion.

The applied torque was computed by applying a known weight to a fixed lever arm of total length 0.1 meters. The joint was oriented such that the length of the lever arm was perpendicular to the direction of gravity. The weight was increased in increments of approximately 5 grams by adding small metal pellets to the cup attached to the end of the lever arm. Our joint is equipped with an AEAT-8800 magnetic encoder, which we used to measure the joint angle throughout the experiment. The brake was engaged using a

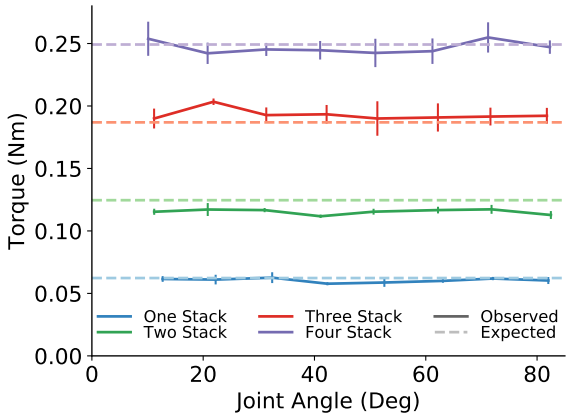


Figure 6.9: Observed and expected holding torque measurements throughout the joint's range of motion. Configurations of up to four stacks of electrodes are measured. Measurements are averaged over five trials.

15 Hz bipolar square wave with an amplitude of 1000 V. While this is a high voltage, we used a power supply limited to a maximum current output of 1 mA as a safety measure. We measured the maximum holding torque at intervals of ten degrees for joints with up to four stacks of electrodes. For each configuration of joint angle and stacks of electrodes, we performed five measurements of the holding torque.

As expected, the brake's holding torque increases proportionally to the number of electrode stacks (Fig. 6.9). Small differences between the observed and expected torque were likely caused by slight deviations in dielectric thickness, electrode fabrication, and applied voltage. Furthermore, the joint's braking capability is generally independent of joint angle, which suggests that electrode conformance is consistent throughout the joint's range of motion. We expect that perpetually increasing the degree of stacking will continue to yield linear increases in braking torque as long as the spacing between adjacent stacks can be kept constant.

For holding torques within operating ranges of robotics applications, electrostatic brakes

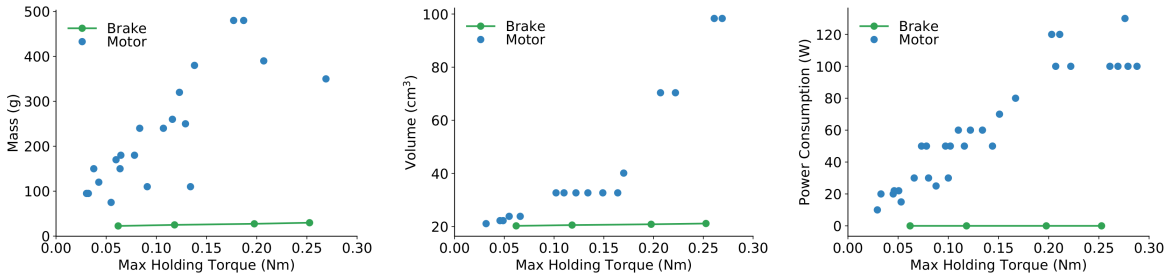


Figure 6.10: Comparison of the brake equipped joint with motors that have similar holding torques across multiple metrics. Our joint is at least four times lighter and one thousand times more power efficient than the compared motors. (Left) Mass comparison. (Middle) Volume comparison. (Right) Power consumption comparison.

have significantly less weight, size, and power consumption than motors, which serve as the primary mode of actuation in robotics (Fig. 6.10 and Table 6.1). Here, we compare our electrostatic brake equipped joint to Maxon motors because they are frequently used (either directly or as a subcomponent of a servo motor system) in high degree-of-freedom robots [Liu et al., 2008, Wright et al., 2012, Tuffield and Elias, 2003, Ha et al., 2011, Buckingham et al., 2007, Schulz, 2011, Liljebäck and Mills, 2017, Karakasiliotis, 2013]. We observe that our brake equipped joint is at least four times lighter and one thousand times more power efficient (approximately $0.01W$ power consumed per stack) than the compared motors. Furthermore, our joint exerts the same holding torques as the larger volume motors while remaining as compact as the smaller motors. Yet we also recognize that motors have capabilities that brakes do not; brakes can only resist force/torque, while motors can independently apply force/torque. However, if the use of motors in a particular application violates design constraints, one or more of them can be replaced by our brake in order to achieve the same amount of holding torque while significantly reducing weight, volume, and power consumption.

Our measurements for the mass, volume, and cost of the brake equipped joint include

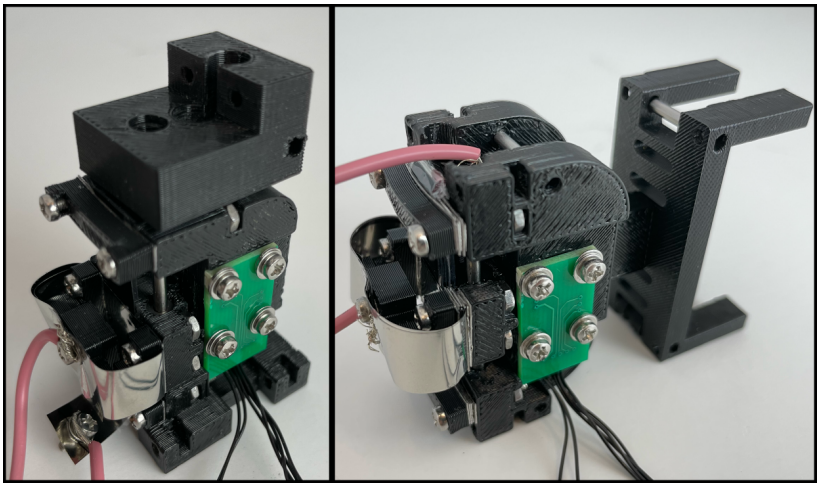


Figure 6.11: The joint design can be adapted according to the desired robot kinematics. (Left) Original joint design used to evaluate braking strength. (Right) A modified joint with symmetric joint limits used in the ten degree-of-freedom robot.

that of the outer casing, rack, pinion, and brakes, as well as that of the materials for coupling them together (ball bearings, dowel pins, spacers, hex nuts, and screws). By measuring the voltage drop across a current sense resistor in series with the output of the high voltage power supply and the H-bridge circuit, we computed the current consumption of the electrostatic brake with Ohm's law. Given the output voltage of the power supply, we used these current measurements to compute the brake's power consumption. The holding torque, mass, volume, cost, and power consumption of the motors are either directly given or derived from the Maxon catalog. For each individual metric, the best performing motors over the observed range of braking torques were reported.

6.5 Braking for Highly Articulated Mechanisms

Any series of two or more robotic joints can benefit from the reduced volume, weight, and power consumption associated with exchanging motors for electrostatic brakes. These benefits become particularly evident for mechanisms that leverage many degrees of freedom

to achieve their task, such as wrapping around objects. For example, reducing the power consumption of a deep-sea robotic tentacle will improve its battery life, or decreasing the weight of a snake robot will allow it to climb more effectively. To demonstrate the value of electrostatic braking for such mechanisms, we developed a robot composed of a serial chain of ten brake equipped joints, driven by only a single servo motor. Despite only having a single motor, these brakes facilitate the execution of highly dexterous maneuvers.

6.5.1 Robot Description

We modified the outer shell of each joint to provide routing for the robot's single pair of tendons, prevent inadvertent contact between the brake electrodes and other parts of the robot and its environment, and increase the overall joint range to have symmetric limits of –60° and 60° (Fig. 6.11 and Table 6.1). The robot's brake design did not change, and the description of joint construction in Subsection 6.3.2 still applies here. Each joint was equipped with two stacks of brake electrodes.

Our ten degree-of-freedom robot uses a single Dynamixel XM430 servo motor to exert tension on a single pair of tendons that are routed through the robot's joints. Each tendon is a segment of 0.45 mm diameter braided fishing line that can withstand up to 65 pounds of tension force. One tendon is routed through the left side of the robot, while the other is routed through the right side. The distal end of each tendon is secured to the final link of the robot by wrapping it around a screw that has been twisted into a screw insert. The proximal end of each tendon is wrapped around its own dedicated 3D-printed tendon spool. The single servo motor controls the motion of both spools; as one spool is reeled in by the motor, the movement of the robot unreels the other spool as necessary.

In order to reach a desired joint configuration, our ten degree-of-freedom robot controls the state of its ten brakes (on or off) and the velocity of the single motor. For this experiment, the magnitude of the commanded velocity of the motor is always 0.2 rad/sec. To decide the sign of the velocity, each joint votes for the sign corresponding to the motor velocity required to reach its desired joint value. The joints that are in the minority turn on their brakes, and

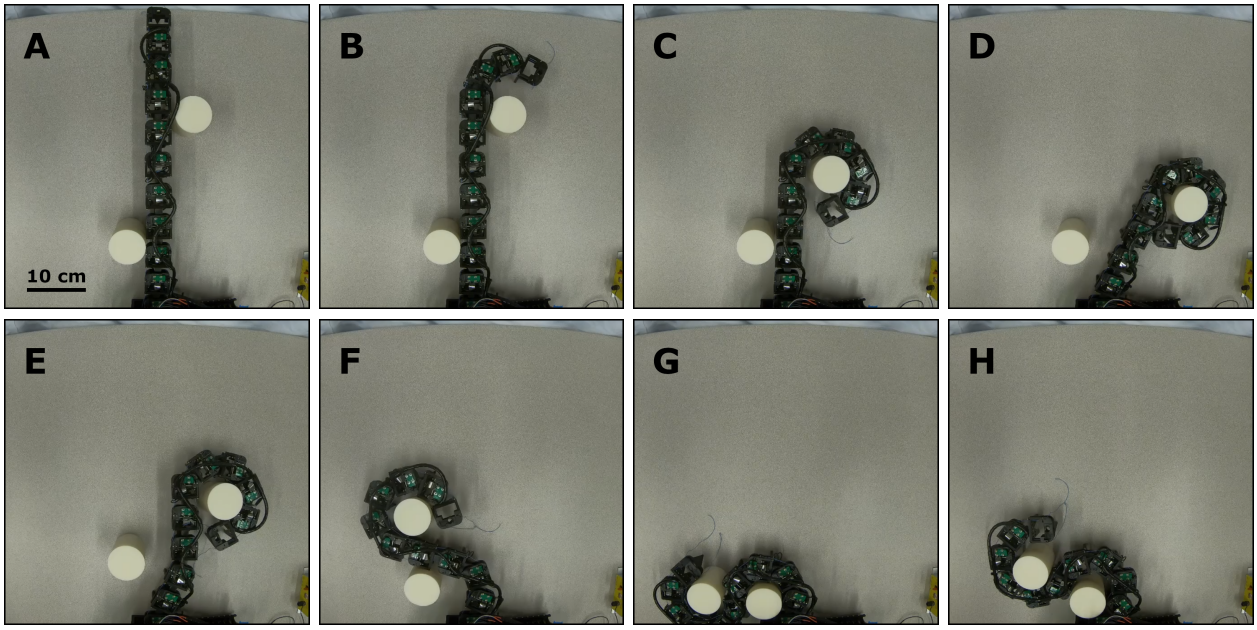


Figure 6.12: The ten degree-of-freedom, single servo motor robot wraps around two objects and then translates them along the tabletop. (A) The initial configuration of the robot and objects. (B) The robot begins wrapping around the upper cylinder. (C) The robot completes a cage around the top cylinder. (D) The robot re-positions in order to avoid a self-collision during its upcoming cage of the lower cylinder. (E) The robot translates the upper cylinder back towards the center of the table. (F) The robot maintains its cage around the upper cylinder as it begins to wrap around the lower cylinder. (G) The robot forms a cage around the lower cylinder. (H) The robot manipulates both objects by moving its most proximal joint while locking all other joints in order to maintain both cages.

the motor's commanded velocity is set according to the votes of the majority. As each joint in the majority reaches its desired joint value, it turns on its brake. Once all of the joints in the majority have reached their desired values (and therefore have turned their brakes on), the joints in the minority turn off their brakes and the motor's commanded velocity is set according to the wishes of the brakes in the minority. As each joint in the minority reaches its desired joint value, it turns on its brake. Once the final joint in the minority reaches its desired value, the overall joint configuration has been achieved.

6.5.2 Manipulation of Multiple Objects

As an example, we consider the challenging task of simultaneously manipulating two plastic cylinders (Fig. 6.12). The robot uses its distal links to wrap around the top cylinder. Once the cylinder has been caged, the robot curls its more proximal links around the lower cylinder while maintaining the distal links' cage. After creating a second cage around the lower cylinder, the robot locks all but the single most proximal of its joints. This allows it to maintain both cages as it actuates the joint at its base in order to manipulate both cylinders.

6.6 Discussion

Our experiments demonstrate that electrostatic braking is an attractive actuation strategy whenever full actuation of a multi-jointed mechanism violates weight, volume, or power consumption constraints, but conventional underactuation does not provide sufficient control. Stacking thin brakes on top of each other allows for braking capability to be scaled as necessary regardless of whether the corresponding joint is large or small. We proposed a joint design that optimizes electrode conformance by converting rotational motion of the joint into linear sliding between the electrodes. We observed that even with multiple stacks of brakes, the joint design achieves optimal electrode conformance. Specifically, measurements of the joint's braking strength were as large as that predicted by its theoretical model. In comparison to motors, our brake-equipped joint is four times lighter and one thousand times more power efficient while exerting similar holding torques.

We use this joint design to develop an underactuated, highly articulated robot. We built a ten degree-of-freedom robot that uses a single motor in concert with its brakes to cage and manipulate multiple objects. A conventionally underactuated version of this robot would not be able to complete this task. We demonstrated brake-aided actuation for a particular highly articulated robot, but it is applicable to any mechanism that uses tendons to couple together a set of two or more joints. In order to gain even greater control over the motion of the joints, future electrostatic brakes could exert a continuum of braking forces (rather than only being on or off) by low-pass filtering a pulse-width modulated voltage.

6.6.1 Limitations

Unfortunately, the use of relatively high voltages can be an impediment to the wide spread adoption of electrostatic brakes due to the need for additional safety precautions, high voltage tolerant electronics, and electric spark resistant fabrication. While stacking of electrodes allows braking capability to be increased linearly, it will unlikely be able to compensate for the quadratic reduction in braking capability accompanied with lowering the applied voltage. Instead, the development of insulators that are increasingly thin yet characterized by low cost, mechanical ruggedness, and high permittivity will allow the voltage to be reduced and, more generally, encourage further adoption of electrostatic braking in robot joints.

6.6.2 Implications

The dexterity of conventional robots is typically dependent on the use of many motors. Replacing conventional motors with electrostatic brakes is a design direction that enables significant reductions in a robot's cost, weight, volume, and power consumption while maintaining the ability to reach arbitrary joint configurations. This approach will enable robots to achieve dexterous movement in scenarios that were not previously possible.

Practical mobile manipulation tasks often require the robot to be untethered from remote power sources. Instead, the robot will typically draw energy from on board batteries that implicitly dictate the amount of time that the robot can operate. Weight will also

influence operation time as heavier robots will need to expend more energy to move and/or manipulate. Autonomous aerial vehicles (AAV) equipped with manipulators exemplify such a scenario [Baizid et al., 2017, Orsag et al., 2014]. While AAVs can be deployed to areas that are inaccessible to land-based vehicles, their battery life is typically on the order of tens of minutes. The use of manipulators further decreases the battery life of an AAV, but the development of light weight, power efficient brake-aided manipulators will mitigate the amount of energy spent on manipulation. Mobile manipulation is just one area in which electrostatic braking can address the need for robots to perform dexterous movement in weight, volume, or power constrainted settings; others include prosthetics, climbing robots, and undersea robots.

Although the robots from this work have rigid structures, electrostatic brakes are also well suited for installation into soft robots due to the flexibility of their underlying materials [Polygerinos et al., 2017, Manti et al., 2016]. In particular, thin electrostatic brakes could be embedded in the skin of biologically inspired robots to either help actuate the joints of an internal skeleton or dynamically create joints in completely soft robots. Skin-embedded electrostatic brakes could serve a dual purpose; not only providing a means of actuation, but also leveraging the capacitive nature of the brake to sense external forces applied to the skin.

Chapter 7
IN-HAND MANIPULATION

While the previous chapters have separately discussed either sensing or actuation for dexterous manipulation, this chapter aims to examine both areas within a single, common application: in-hand manipulation. To this end, we build a six degree-of-freedom, two-fingered robot hand for planar in-hand manipulation. The hand is composed of electrostatic brake equipped joints from Chapter 6, and its fingertips are adapted versions of the optical proximity, contact, and force sensors presented in Chapter 5. This hand demonstrates the feasibility of incorporating alternative forms of both sensing and actuation into robot manipulators, and serves as a platform for studying how they enhance dexterous manipulation. Here, we examine how optical proximity sensor based pose estimation and hybrid motor-brake control affect the speed and precision of in-hand manipulation.

7.1 Robot In-Hand Manipulation

In-hand manipulation skills are critical to many of the tasks that are desirable for robots to perform. Such skills are defined by their ability to reorient an object within the hand, and can be decomposed into action primitives according to the direction of translation, axis of rotation, and intended motion of the object with respect to contact point(s) [Bullock et al., 2012]. For example, one would likely pick a pencil off of a table with a pinch-like grasp, but then rotate and shift the pencil up between the thumb and index finger in order to grasp near the head of the pencil so as to to have fine control over how the tip marks the paper when writing. More generally, tool use, object articulation, and pick and place tasks will significantly benefit from the ability to reposition, reorient, and exert force on the object of interest within the robot's hand. Achieving such tasks will require the robot to robustly

perceive the object, as well as exert sufficient control of its limbs to apply desired forces on the object.

7.1.1 Electrostatic Braking for In-Hand Manipulation

Many in-hand manipulation tasks consist of smaller sub-skills, each of which requires the robot manipulator to execute different ranges of motion. Grasping or pinching an object in the palm may warrant substantially different movement than rotating or twisting it. Accommodating such wide variations in movement requires the hand to be highly articulated with up to twenty or more degrees of freedom. The most dexterous robot hands actuate these large number of joints with ten or motors [Andrychowicz et al., 2020, Zhu et al., 2019, Funabashi et al., 2015, Xu and Todorov, 2016], which may disqualify them from being used in applications that are sensitive to weight, size, or power consumption.

At the other end of the dexterity spectrum are highly underactuated hands that have significantly less weight and power consumption [Kontoudis et al., 2015, Gosselin et al., 2008, Krausz et al., 2015, Chamara and Gopura, 2019, Tavakoli and de Almeida, 2014]. These hands are designed for grasping common objects rather than manipulating them within the hand. Many such designs route tendons through multiple phalanges in order to couple a large number of degrees of freedom to a smaller amount of actuators. This static coupling limits the dexterity of the hand because, in the absence of external forces, the fingers are only able to move along a small number of fixed trajectories.

We intend to improve the range of motion of these types of underactuated designs by incorporating electrostatic brakes into the joints of the hand. We hypothesize that using these inexpensive, low-power, and compact brakes to vary the stiffness of the joints will allow the dexterity of the hand to approach that of humans. Aukes et al. [2014] previously employed electrostatic brakes in a robot hand to lock the joints in order to reach discrete grasping configurations, but was unable to perform dynamic variation of joint torque due to hysteresis caused by charge injection. However, by leveraging more recent methods for driving electrostatic brakes [Hinchet and Shea, 2020], we believe that these hysteretic affects

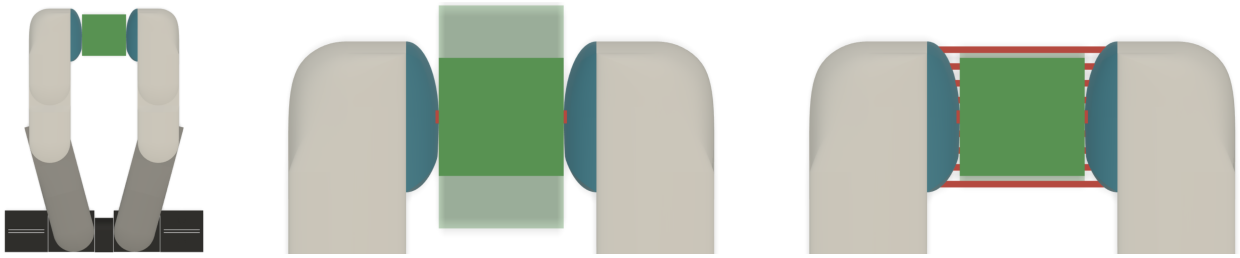
can be mitigated to an extent that continuous control can be achieved. In this work, we combine hybrid motor-brake actuation with a model predictive control method to perform in-hand manipulation.

7.1.2 Optical Proximity Sensing for In-Hand Manipulation

As a robot attempts to execute a particular in-hand manipulation skill, the appropriate control action at any point in time will depend heavily on the pose of the object. Pose estimation algorithms fuse the robot's control actions and sensor observations across time in order to continuously track the pose of the object. These algorithms must be provided with sensor measurements that disambiguate the pose of the object in order to perform well, but sensing modalities currently used for in-hand manipulation struggle to continuously provide such measurements. Here, we analyze the application of a previously unused sensing modality - fingertip embedded proximity sensors - for pose estimation during in-hand manipulation.

The effectiveness of a sensor used for object localization is determined by its ability to measure discriminative features of the object during manipulation. For RGB(D) image sensors, these features are image pixels that correspond to keypoints of the object or a fiducial attached to the object. When these features are occluded from view, pose estimation performance suffers. This is particularly an issue for in-hand manipulation as the fingers will often block environment- or robot-mounted image sensors' view of the object.

Tactile sensing is the other most commonly used sensing modality applied to in-hand manipulation. Its features consist of contact measurements that are the result of an object's surface exerting force on the sensorized portion of the finger. Tactile sensors can be made robust to the occlusion that image sensors experience by being embedded into the contact surface of the fingers. On the other hand, the ability of tactile sensors to detect discriminative features is limited by the fact that they require contact with the object in order to make measurements. Even if the sensor has a high spatial resolution [Yuan et al., 2017, Donlon et al., 2018, Lambeta et al., 2020], objects that do not conform to the sensor's surface (e.g. objects with high amounts of curvature) may be difficult to localize. A sensing modality



(a) (b) (c)

Figure 7.1: Optical proximity sensors provide measurements beyond the local areas of contact that reduce uncertainty in pose estimation. a) Top down view of a two fingered hand with blue fingertips manipulating a green object. b) Zoomed in view in which the red regions indicate object-fingertip contacts that would be detected by tactile sensors. While detection of these contacts results in the object being well localized in the horizontal direction, the translucent green area illustrates the large pose uncertainty in the vertical direction (assuming the geometry of the object is known apriori). c) Optical distance measurements illustrated by red rays emanating from the fingertips detect a greater portion of the object and thereby significantly reduce pose uncertainty in the vertical direction.

that improves upon the two described here would be robust to finger occlusions and also be able to measure object geometry beyond the local areas of contact.

Proximity sensors successfully capture discriminative geometric features for pose estimation where previous sensing modalities fail. The robot's fingers can heavily occlude the object from view of image sensors mounted to the environment or the robot itself, but proximity sensors are robust to such occlusion by virtue of being embedded in the fingers themselves. Tactile sensors embedded in the fingers are also robust to occlusion, but require contact in order to detect the object. Therefore, it can be difficult for tactile sensors to localize objects that do not conform to the sensorized portion of the finger surface. Consider a scenario in which a two fingered hand with slightly rounded fingertips attempts to manipulate an object

with flat sides, such as in Fig. 7.1. In such a scenario, contact detection indicates that the object is within the robot's grasp, but does not elucidate whether the contacts are towards the top corners of the object, bottom corners of the object, or somewhere in between. Proximity sensors can resolve this ambiguity by sensing beyond the local areas of contact, resulting in improved pose estimation. In this work, we perform Bayesian pose estimation based on measurements from fingertip embedded optical proximity sensors.

7.1.3 Contributions

In this chapter, we evaluate the performance of hybrid motor-brake control and proximity sensor based pose estimation in the context of in-hand manipulation. Specifically, we measure the speed and precision with which a two-fingered hand can move a cylindrical object from one side of its workspace to a specified position on the other side of its workspace. Our primary insight is that alternative forms of actuation and sensing can address challenges to in-hand manipulation that conventional modalities cannot. Specifically, hybrid motor-(electrostatic) brake actuated hands serve as an effective compromise between dexterous fully actuated hands and low cost, light weight, and power efficient underactuated hands. Furthermore, proximity sensors are particularly well-suited to pose estimation for in-hand manipulation because they avoid occlusion by being embedded in robot fingertips while also being able to detect portions of the object that do not necessarily make contact with the fingertip surface.

Our work aims to quantify the degree to which alternative forms of sensing and actuation enhance in-hand manipulation relative to baseline mechanisms. To that end, we first describe a novel two-fingered robot hand for planar object manipulation with electrostatic-brake equipped joints and fingertip embedded optical proximity sensors. We also present the first demonstration (to the best of the authors' knowledge) of fingertip embedded proximity sensors being used for pose estimation during an in-hand manipulation task. Our results compare the hand's required execution time and final object positioning accuracy both with and without the use of brakes. We also obtain these same metrics (manipulation speed and

precision) when examining the accuracy of proximity sensing based pose estimation versus that of tactile based pose estimation. A supplementary video of this work is available here: https://youtube.com/watch?v=Ct9k0mSmAYs

7.2 Electrostatic Braking and Proximity Sensing for In-Hand Manipulation

To investigate the value of electrostatic braking and proximity sensing for in-hand manipulation, we deploy a complete system for object manipulation under pose uncertainty on our custom two-fingered robot hand. We first describe our robot's hardware architecture while focusing on its brake-equipped joints and compliant fingertips with embedded optical time-of-flight proximity sensors. We then describe our implementation of a particle filter that uses measurements from these sensors to estimate the object's pose, and conclude by detailing the model predictive control policy that our robot uses to perform in-hand manipulation.

7.2.1 Two-Fingered Robot for In-Hand Manipulation

We developed a two-fingered robot hand capable of in-hand manipulation as shown in Fig. 7.2. For each finger, a tendon (consisting of 65 lb. max tension fishing line) is anchored to the fingertip and then routed through the links of the finger in order to be attached to a XM430-W350-R Dynamixel Servo motor. Flexion of a finger occurs when its corresponding motor pulls on the tendon, and springs embedded in each joint extend the finger as the motor releases the tendon. Each finger is composed of three revolute joints. Each joint contains an AEAT-8800 magnetic encoder. The encoders measure the angular position of each joint at a rate of 30 Hz. The limits for each joint are from 0 to 90 degrees.

7.2.2 Electrostatic Brake Equipped Fingers

Although the fingers are underactuated, a unique feature of the robot hand is that the rotation of each individual joint can be blocked by a corresponding electrostatic brake, allowing the motion of the joint to be decoupled from the motor as desired. Electrostatic brakes

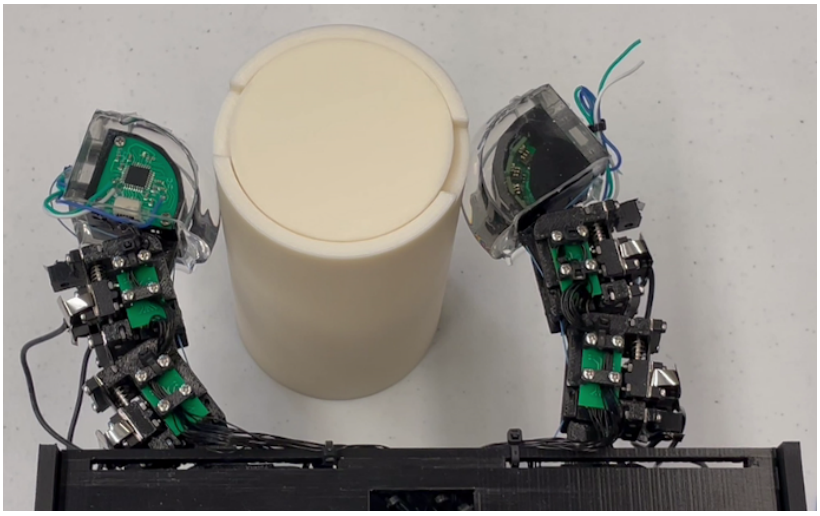


Figure 7.2: A two fingered hand for in-hand manipulation. Each of its joints is equipped with an electrostatic brake, and its compliant fingertips have embedded proximity sensors.

leverage the electrostatic attraction that occurs between two conductors at differing voltage potentials. This attraction induces a perpendicular frictional force that resists motion between the conductors. By integrating a rack and pinion transmission into our joints, we can transform rotational motion of the joint into linear sliding between the conductors. This optimizes conformance between the brake's electrodes in order to achieve significant braking capability. The robot can engage any combination of joints' brakes in order to block the motion of those joints. These brakes enable the robot to control the motion of individual joints, allowing the hand to reach any arbitrary joint configuration within the joint limits.

Here, we use the brake designs and joints described in Chapter 6 to form the robot's fingers. Each finger consists of an 'Updated Joint' at the base, and an intermediate 'Prototype Joint' and a distal 'Prototype Joint'. Each joint is equipped with two stacks of electrostatic brakes. We use the same brake control circuitry, where the control signal has an amplitude of 950 V.

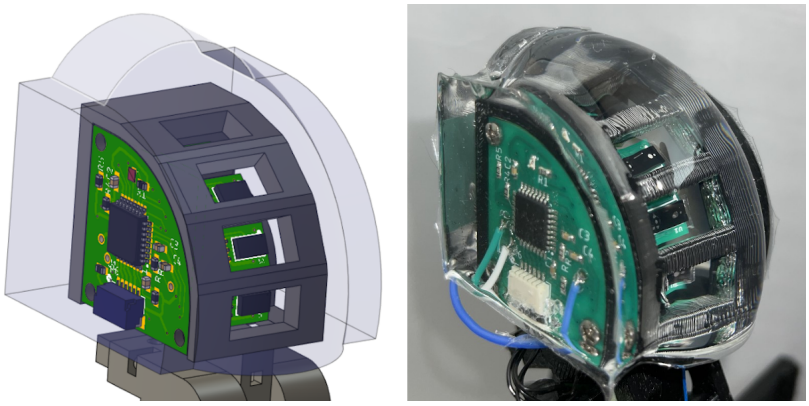


Figure 7.3: A compliant fingertip with four embedded optical time-of-flight proximity sensors. Left: A visualization of the fingertip design. PDMS (colored in translucent blue) forms a compliant contact surface. Right: The constructed fingertip.

7.2.3 Compliant Fingertips with Embedded Proximity Sensors

Optical time-of-flight sensors possess a number of qualities that facilitate the practical implementation of proximity sensor based pose estimation. This type of sensor makes easily interpretable (i.e. distance) measurements that are insensitive to most object compositions and surface properties. The ability to compactly manufacture such sensors allows them to be embedded in compliant regions of the finger, and this compliance aids the robot hand in maintaining a grasp on the object. This subsection describes the design and construction of our robot hand's sensorized fingertips.

Four STMicrolelectronics VL6180x optical time-of-flight modules are embedded into each of the robot's fingertips as shown in Fig. 7.3. The sensor modules of each fingertip are positioned along a circular arc of radius 15mm at intervals of 25 degrees, and oriented such that their optical transmitters point perpendicular to the arc's tangent. Each sensing module produces distance measurements at a rate of 30 Hz, which are sent to a fingertip embedded microcontroller that provides the measurements to the robot's main computer via the same communication protocol used to query the encoders.

After the fingertip's embedded electronics are mounted to a 3D printed substrate, this rigid skeleton is covered by transparent, compliant polydimethylsiloxane (PDMS). We design a mold for the fingertip that fulfills two critera:

- For each sensor module, the contact surface above the module's optical transmitter and receiver should be shaped as a circular arc centered around the transmitter in order to focus infrared light reflecting off of the inner contact surface back towards the transmitter (as opposed to towards the receiver). This results in better contrast-to-noise ratio performance of the sensor.

- The contact surface should be at a distance above the 10mm minimum sensing range of the module (here, we use a distance of 15mm).

Once a satisfactory mold has been 3D printed, the surface is covered with two coats of Smooth-On XTC3D Epoxy. This epoxy fills in the gaps between the individual 3D printed layers and is necessary to produce a contact surface that is optically transparent. The mold is then filled with a 10:1 silicone-catalyst PDMS mix, thoroughly degassed using a vaccum chamber and pump, and cured for 24 hours. For more details on this process, please see our previous work on the construction of optical proximity-contact-force sensors [Lancaster et al., 2019].

7.2.4 Proximity Sensor Based Pose Estimation

By virtue of using similar measurement mechanisms, many of the standard techniques applied to LIDAR based localization in the context of mobile robots can be adapted for proximity sensor based object pose estimation [Thrun et al., 2001]. In particular, we implement a particle filter for estimating the pose of the manipulated object. Our encoder measurements themselves provide a sufficiently accurate estimate of the robot's internal state, therefore our particle filter's estimated state x_t only consists of the cylindrical object's xy position.

Given N particles, our particle filter's sensor model generates particle weights $p(z_t|x_t^i)$ that represent the likelihood of sensor measurements z_t occurring given each particle hypothesis x_t^i, where $i \in \{1, \ldots, N\}$. Like other beam-based sensor models, we assume that each individual sensor beam measurement is conditionally independent of all of the others given the underlying state:

$$p(z_t|x_t^i) = \prod_{j=1}^{M} p(z_t^j|x_t^i) \tag{7.1}$$

with $M = 8$ due to the robot having two fingers, each with four sensing modules (see Fig. 7.3). The individual observation probabilities are a weighted mixture of a Gaussian distribution representing the expected distance measurement with noise and a uniform distribution representing the possibility of a sensor glitch, communication error, unexpected detection, or other random effects:

$$p(z_t^j|x_t^i) = \eta \cdot \left(w_1 \cdot \mathcal{N}(z_t^{j*}, \sigma^2) + w_2 \cdot \mathcal{U}(0, z_{max})\right) \tag{7.2}$$

where z_t^{j*} is the expected measurement of sensor j given the current encoder values, robot forward kinematics, and particle hypothesis x_t^i. Here, w_1 and w_2 are mixing weights chosen according to the importance of the corresponding term, and η serves as a normalizer to ensure that the sum over the distribution is one.

Each time the robot executes an action, our particle filter's motion model propagates its particles according to the expected change in position plus zero-mean gaussian noise. The expected change in position is computed by executing the action in a physics simulator for robotics. This simulator is a key component of our model predictive control policy, and is discussed in further detail in the following subsection.

7.2.5 Model Predictive Control for In-Hand Manipulation

We use model predictive control (MPC) to generate actions for in-hand manipulation. Starting with the current system state, MPC simulates a large number of action sequences in order

to find trajectories that will move the system towards the goal state over the trajectories' time horizon. By repeating this process at each timestep, MPC provides adaptive control that is robust to unmodeled dynamics.

Given the current state s_t of the robot-object system and the action a_t that the robot executes, our dynamics model predicts the resulting next state s_{t+1}. The state s_t consists of the six joint positions and velocities, as well as the xy position and velocity of the manipulation object. The action a_t is composed of the commanded positions of the robot's two motors, and the states of each of the six brakes (on or off). While in general any combination of brakes can be turned on, for this work we limit executed braking configurations to be those in which exactly one of the brakes in each finger is off. By reducing the number of possible braking configurations from 64 to 9, it is more tractable for the model predictive controller to explore the action space. We use NVIDIA's Isaac Gym [Makoviychuk et al., 2021], a GPU based physics simulator, as a highly parallelizable dynamics model.

Our in-hand manipulation controller is an adapted version of the model predictive path integral (MPPI) framework [Williams et al., 2017, Zhong et al., 2019]. The standard MPPI controller generates a large number of action sequences, simulates those sequences to obtain corresponding trajectories and costs, and then outputs a cost-weighted average of those action sequences for execution on the real robot. However, this averaging does not make sense for a hybrid action space for which the discrete variables lack a Euclidean distance measure. Instead, we require each action sequence to maintain a consistent brake configuration throughout the corresponding simulated trajectory, and then compute cost-weighted averaged action sequences for each possible brake configuration. Initially, of the 9 outputted action sequences, we execute the first action of whichever sequence has the lowest cost. For subsequent time steps, we only choose an action sequence corresponding to a different brake configuration if it has a significantly lower cost (we use a constant threshold percentage ϕ) than the cost of the action sequence corresponding to the previous brake configuration. Otherwise, we execute the first action of the action sequence corresponding to the previous brake configuration.

At each time step, our MPPI controller simulates many trajectories over a time horizon T. It computes a cost for each trajectory that penalizes the fingers not making contact with the object and object distance from the goal:

$$J(s_t, \ldots, s_{t+T}) = a_1 \cdot \sum_{\tau=t}^{t+T} \mathcal{I}(s_\tau) + a_2 \cdot |x_{goal} - x_{t+T}| \tag{7.3}$$

where $\mathcal{I}(s_\tau)$ is an indicator function that returns the number of fingertips *not* in contact with the object, and x_{goal} is the desired position of the object.

7.2.6 Implementation Details

The algorithms described in Sections 7.2.4 & 7.2.5 are simultaneously executed on a single desktop PC with an Intel i7 Quad-Core CPU, 64 GB of RAM, and a NVIDIA Titan XP GPU. Initial values for all of the following parameters were chosen based on our intuition and then hand-tuned until reasonable performance was achieved. Our particle filter used 1000 particles and produced pose estimates at a rate of 18 Hz. Its sensor model used parameter values $\sigma = 5mm$, $w1 = 0.95$, $w2 = 0.05$. Regardless of whether brakes are being used or not, our MPPI controller simulates 297 trajectories over a time-horizon $T = 10$ at each time step. Its cost function uses parameter values $a_1 = 0.1$, $a_2 = 200$, and the MPPI hyperparameter λ is set to 0.1. Switching between braking configurations is thresholded on a value of $\phi = 25\%$. Controls are generated at a rate of 5 Hz.

7.3 Evaluation

We consider a planar in-hand manipulation task in which the robot must translate an object from a known initial location to a goal location. These locations are chosen to be on the boundary of the robot's manipulation workspace. The object's initial pose is 4.5cm to the left of the geometric plane that symmetrically bisects the hand (see Fig. 7.6), and 4.5cm above the base of the fingers. The goal pose is the reflection of the initial pose across the bisecting plane. The object is a cylinder of radius 4cm and a height of 14cm.

When attempting the in-hand manipulation task, the robot begins at a position in which all joints have a value of zero (Fig. 7.6 on the left). It then uses a joint position controller to move to a preset pose that corresponds to the robot making initial contact with the object at the known initial pose. Once contact has been made, the MPPI controller attempts to move the object to the goal pose. A successful manipulation trial ends when the horizontal distance from the object's ground-truth location (provided by visual fiducial localization) to the goal location is less than 1mm.

The following subsections present two experiments. The first experiment measures how the use of electrostatic brakes affects in-hand manipulation performance. The second experiment compares in-hand manipulation performance when using proximity sensor based pose estimation versus that of tactile sensor based pose estimation.

7.3.1 Evaluation of Electrostatic Brake Aided In-Hand Manipulation

We evaluate the performance of our brake-enabled robot hand during in-hand manipulation. In order to measure the advantage of brake enabled mechanisms relative to their conventional underactuated counterparts, we quantify the robot's in-hand manipulation dexterity both with and without the use of brakes (Fig. 7.4). Specifically, we measure the execution time required to complete the manipulation and the distance between the object and the goal position at the end of the manipulation. We perform ten trials for each method (no braking and braking). The configuration of the fingers when initially making contact with the object was chosen so that the robot is able to complete the manipulation without the use of brakes. For this experiment, we use visual fiducials for object localization in order to focus on the effects of the brakes. Instead, the experiment in the next subsection focuses on the effect of alternative sensing techniques.

Results

We observe that the use of electrostatic brakes significantly improves the dexterity of the robot hand. Without the brakes, the system is limited to a space of trajectories that corre-

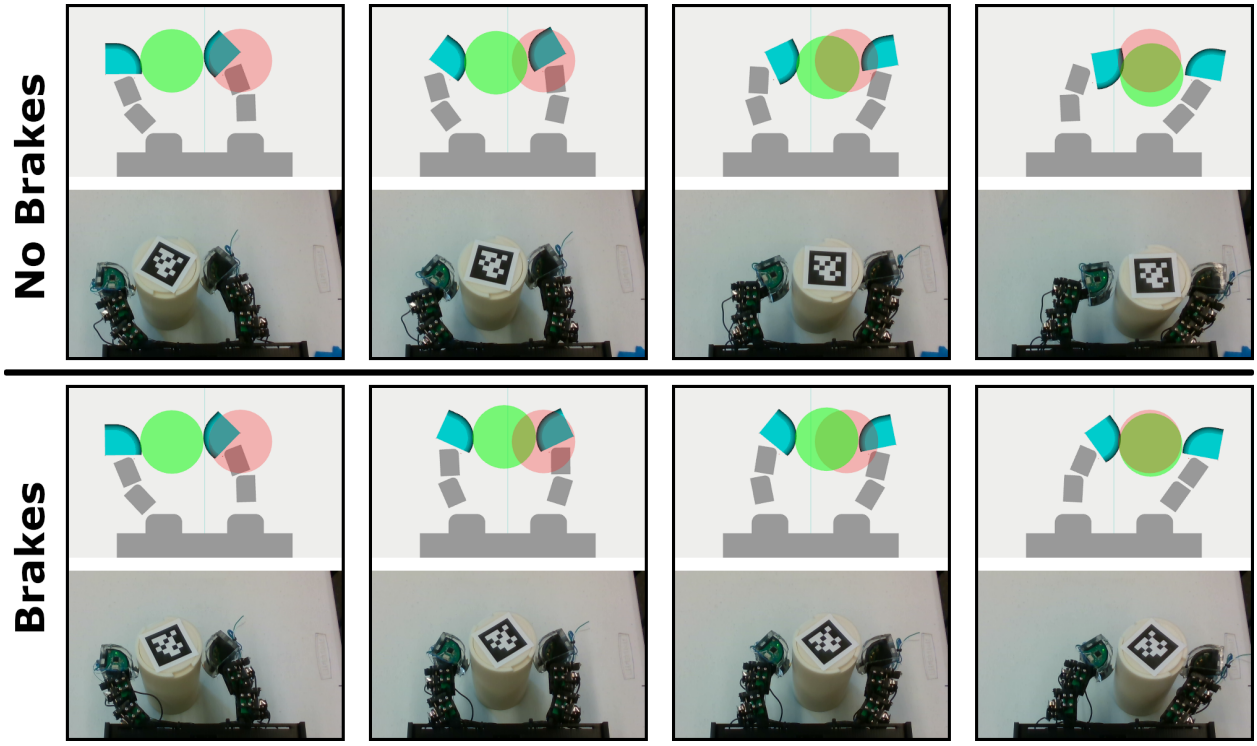


Figure 7.4: Representative in-hand manipulation trials both with and without the use of electrostatic brakes. For each pair of images, the upper image shows a visualization of the robot-object system, and the lower image shows the real object and robot. The green cylinder represents the pose of the object, and the red cylinder represents the desired goal pose of the object. The left-most snapshots show the pose at which initial contact is made for both trials, right most snapshots show the final pose of the object, and intermediate snapshots show the progression of the manipulations. Top: A manipulation trial in which the brakes are not used. Bottom: A brake-aided manipulation trial.

spond to just pulling and releasing the tendons. The flexibility to control individual joints via braking allows the controller to find trajectories that are both faster and terminate more closely to the goal pose.

The robot hand was only able to complete nine out of ten manipulation trials without the use of brakes, but succeeded in all ten manipulation trials that did use brakes. During

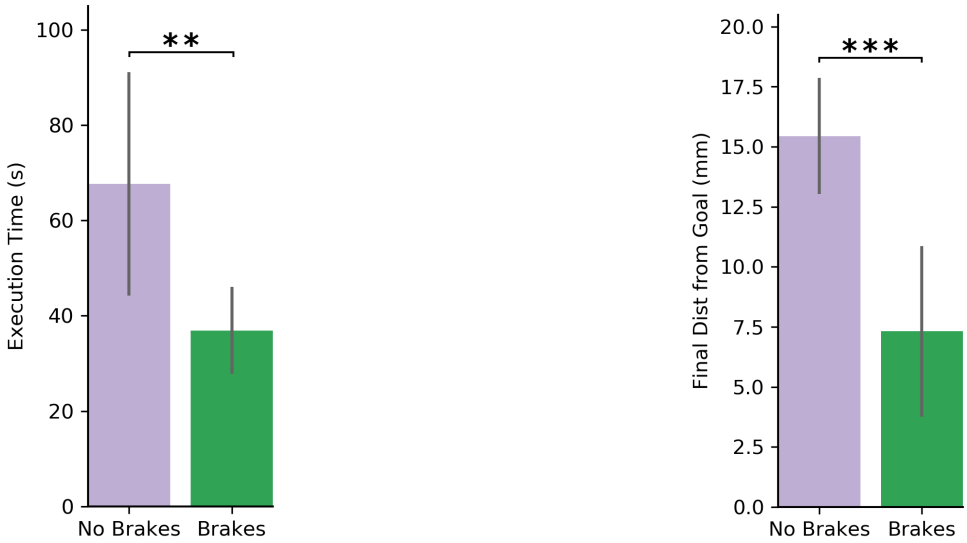


Figure 7.5: Comparison of in-hand manipulation performance when using or not using brakes. Error bars correspond to one standard deviation. Increasing number of stars indicates higher significance levels ($p < 0.05$, $p < 0.01$, $p < 0.001$) according to a paired U test. (Left) The amount of time required to complete the manipulation. (Right) The distance between the final pose of the object and the desired goal pose.

the brakeless manipulation trial that failed, the robot-object system reached a state in which the controller could not find actions that would bring the object closer to the goal position. We considered this trial to be a failure after the robot was unable to make any progress towards the goal over the course of several minutes. Note that all of the following statistics are computed using only the successful manipulation trials. We found that the use of brakes significantly reduces the amount of time required to complete the in-hand manipulation task ($p < 0.01$, Mann-Whitney U test) as shown in Fig. 7.5. The robot was able to complete the task in 36.9 seconds on average when using brakes, which is 45% faster than the 67.7 second average execution time observed when not using brakes. Upon completing the task, brake-aided manipulation had positioned the object significantly closer to the goal position compared to manipulation without brakes ($p < 0.001$, U test). Without brakes, the robot

achieved an average final positioning error of 15.4 mm. Brake-aided manipulation achieved 7.3 mm final positioning error on average, which represents a 53% reduction in error relative to manipulation without brakes.

7.3.2 Evaluation of Proximity Sensing for In-Hand Manipulation

We measure the robot's in-hand manipulation performance when using pose estimation based on three different sensing modalities. These three modalities are tactile sensing, proximity sensing, and visual fiducial localization. For each of the sensing modalities, we measure the distance between the goal pose and the object's pose at the end of the manipulation. Fiducial localization also serves as a ground-truth pose estimate during the tactile and proximity sensor trials in order to estimate pose estimation accuracy throughout the manipulation. Ten manipulation trials are undertaken for each of the three sensing modalities. A trial ends and is considered to have failed if the previous condition is not fulfilled within 60 seconds after first making contact with the object. Note that for all trials in this subsection, the robot is using hybrid motor-brake control.

We implement tactile sensing by artificially limiting the range of the embedded proximity sensors to be just beyond the contact surface of the fingertip. Although the sensors can detect objects up to 255mm away, a tactile sensor would only be able to detect objects at a distance less than or equal to the distance from the sensor to the finger surface d_{tact_max}. Therefore, all measurements greater than d_{tact_max} are truncated to d_{tact_max} when in tactile sensing mode. We perform this truncation for both the real observations z_t^j and expected observations z_t^{j*} of the particle filter. Although the distance from any sensor to the fingertip surface is 15mm, we set d_{tact_max} to a slightly larger 18mm to ensure that sensor noise does not result in false negatives.

Results

Relative to tactile sensor based pose estimation, we found that proximity sensor based pose estimation is more accurate throughout the trajectory and results in more precise positioning

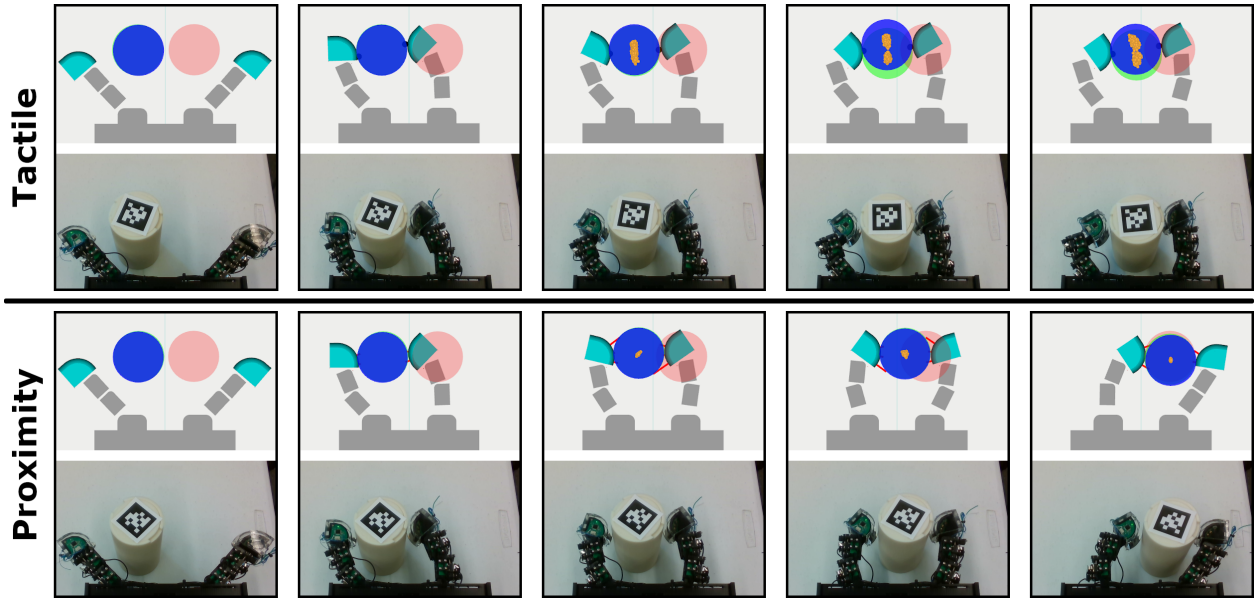


Figure 7.6: Representative in-hand manipulation trials using tactile sensor and proximity sensor based pose estimation. The blue cylinder represents the object pose estimated by the particle filter (particles shown in orange). The green cylinder (heavily overlapped by the blue cylinder, and most visible in images to the upper right) represents the ground-truth pose. The red cylinder represents the desired goal pose of the object. Note that the visual fiducials are only used to estimate the ground-truth pose; they do not affect the sensor based pose estimation nor the executed control policy. The left-most snapshots show the initial pose for both trials, and the snapshots second from the left show when the cyan fingertips initially make contact with the object. Subsequent snapshots show the progression of the manipulations. Top: A tactile sensor manipulation trial. In the upper images, black spheres along the fingertip surface illustrate detected contacts. Here, pose estimation error pushes the system into an area of the state space in which the controller can no longer make progress towards the goal. Bottom: A proximity sensor manipulation trial. Red lines emanating from the fingertips illustrate proximity sensor measurements.

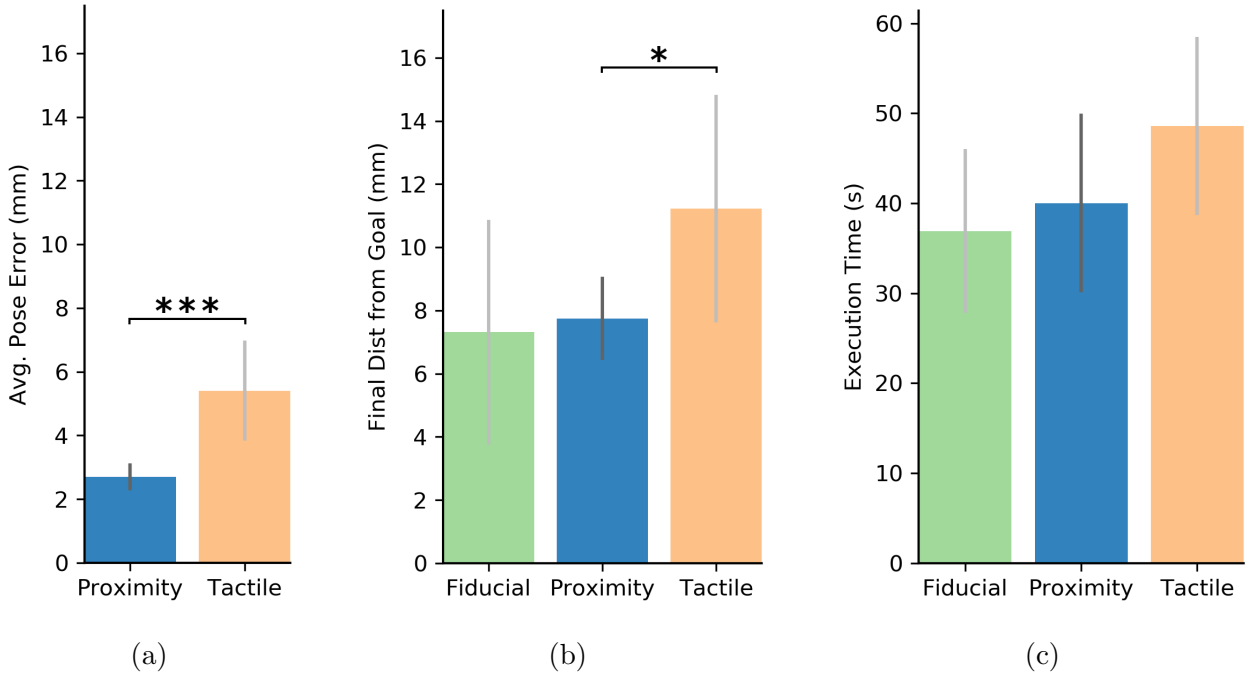


Figure 7.7: Comparison between pose estimation sensing modalities. Error bars correspond to one standard deviation. Increasing number of stars indicates higher significance levels ($p < 0.05$, $p < 0.01$, $p < 0.001$) according to a paired U test. a) The average pose error. b) The object's final distance from the goal. c) Time between initial contact and task completion.

of the object. We observed that the spread of the particle distribution in the vertical direction is larger for tactile sensor based pose estimation (Fig. 7.6). This is exemplified by the second-to-last snapshot of the tactile sensing sequence in which the distribution has bifurcated around two distinct hypotheses. This may be due to the sensor data not being sufficiently discriminative to determine whether the robot is grasping above or below the diameter of the cylinder.

Across our experiments, we found that proximity sensing enables better performance than tactile sensing. Proximity-sensor based pose estimation achieves significantly lower average

Table 7.1: The results of ten in-hand manipulation trials for each object perception method.

	Avg. Pose Err. (mm)	Success	Goal Dist (mm)	Exec. Time (s)
Fiducial	-	10/10	7.3 ± 3.6	36.9 ± 9.1
Proximity	2.7 ± 0.4	10/10	7.8 ± 1.3	40.0 ± 10.0
Tactile	5.5 ± 1.6	5/10	11.2 ± 3.6	48.6 ± 10.0

pose error than tactile-sensor based pose estimation ($p < 0.001$, Mann-Whitney U test). The average pose error throughout the manipulation for proximity sensing was 2.7mm, less than half of the 5.5mm error observed for tactile sensing (Fig. 7.7). The quality of pose estimation is also reflected on the overall task performance. When using visual fiducial or proximity sensor based pose estimation, the robot was able to successfully complete all ten trials. However, only five out of ten trials were successful when using tactile-based pose estimation. For the five trials that succeeded, tactile sensing based pose estimation trials resulted in 11.2mm average final position error. In contrast, proximity-sensor based pose estimation trials resulted in an average final positioning error of 7.8mm across all trials, 30% lower than that of tactile sensing, enabling also significantly lower final positioning error compared to tactile sensing ($p < 0.05$, U test). Furthermore, proximity-sensor based pose estimation resulted in an average task completion time of 40.0 seconds, 18% faster than the 48.6 second average completion time of (successful) tactile sensing trials. The complete statistics for all experiments are listed in Table 7.1.

It is also informative to examine how the pose error evolves throughout the in-hand manipulation task. With the exception of the very beginning of the task, proximity-sensor based pose estimation has lower error than tactile based pose estimation as shown in Fig. 7.8. We hypothesize that the tactile modality does better at the beginning due to a mismatch between the particle filter's sensor model and the actual sensor. The optical beams of the sensor model are represented by beams with an infinitesimally small width, but the real

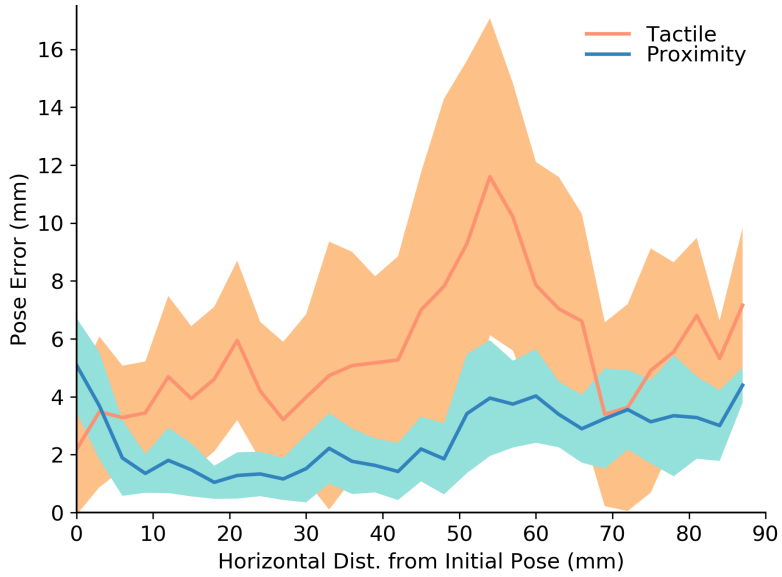


Figure 7.8: The error in pose estimation for both tactile and proximity sensing. Each series is averaged over ten trials, and the shaded errors represent one standard deviation.

sensor's detection zone takes the shape of a cone. For the particular configuration in which the robot initially makes contact with the object, the cylinder is just barely within the conic field-of-view of one of the sensors. This causes the sensor to return a measurement significantly less than its max range, but the corresponding sensor model expected measurement is at max range (because the beam has no width). Tactile-based pose estimation is better able to filter this out because its (truncated) max range is much closer to the measurement returned by the actual sensor. However, because the true sensor's detection cone is quite thin (with a radius on the scale of millimeters depending on the distance from the sensor), this configuration in which the object barely grazes one of the sensing cones does not occur often throughout the overall manipulation. Once the particle filter has coalesced around the true state by observing measurements outside of this configuration (i.e. the manipulation proceeds), it naturally blunts the effect of this type of rare configuration by conditioning upon the previous state distribution.

7.4 Discussion

In this chapter, we studied both proximity sensor based pose estimation and hybrid motor brake control in the context of in-hand manipulation. Equipping an underactuted hand with electrostatic brakes significantly enhances its dexterity without adding much weight or power consumption. The novel application of proximity sensing to in-hand manipulation is motivated by the fact that fingertip-embedded proximity sensors are robust to occlusion (unlike environment- or robot-mounted image sensors) and able to sense beyond the local areas of contact (unlike tactile sensors). We considered an in-hand manipulation task that involved moving an object from one side of its workspace to the other. We found that the use of electrostatic brakes allowed the two-fingered robot to perform in-hand re-positioning of an object 45% more quickly and with 53% lower positioning error than without brakes. We then measured the accuracy of tactile sensor and proximity sensor based pose estimation and its effect on in-hand manipulation performance. Our experiments demonstrate that proximity sensor based pose estimation has an average error of less than 3 millimeters, which is almost half of that of tactile sensor based pose estimation. In terms of task performance, proximity sensing resulted in significantly more robust task completion and final object positioning accuracy relative to tactile sensing.

7.4.1 Limitations

One direction for future work is to perform similar experiments for a wider range of in-hand manipulation tasks. In particular, these experiments could be expanded by using manipulation objects of different size, weight, and geometric shape. The task could be further generalized by using arbitrary initial and goal poses within the robot's workspace.

A limitation of our work is the way in which we implemented tactile sensing. Particularly, higher resolution tactile sensors will likely perform better than our spatially sparse tactile sensor baseline. Although we directly compare tactile sensing against proximity sensing for pose estimation in our experiments, there are other aspects of manipulation that require some

form of tactile sensing. Specifically, tactile sensors can detect applied forces, object slippage, vibrations, etc. that can be useful for manipulation but would be difficult or impossible to measure with purely proximity sensors. Ultimately, we believe that future work should focus on developing fingertip sensors that use either new sensing technologies or a mix of existing mechanisms to achieve high resolution proximity **and** tactile sensing.

Chapter 8

CONCLUSION

Current highly dexterous robots are dependent upon perception and control systems that will fail in unstructured, constrained environments. Rather than continuing to rely on the limited number of modes of sensing and actuation that underlie these systems, we propose to develop alternative and augmentative modalities as a foundation for more robust perception and control.

This thesis first develops a fingertip embedded optical proximity sensor with a range intermediate to that of cameras and tactile sensors. The ability to detect objects beyond the surface of the fingertip allows the sensor to make measurements without requiring contact, and being embedded in the finger makes the sensor robust to occlusion. We observe that the robot can use this sensor for pose estimation by detecting discriminative geometric features of the object of interest, and develop a learning based framework for detecting such features. We then extend the sensor's capabilities into the contact and force measurement regime by covering it with a transparent elastomer. We find that using a rounded air-elastomer boundary effectively directs infrared light relflecting off of the boundary back towards the transmitter, resulting in improved contrast-to-noise ratio.

The second part of this thesis explores alternative methods for actuating the many degrees-of-freedom that dexterous manipulation typically requires. Rather than fully actuating highly articulated robots with many motors, we can augment conventional underactuated robots with electrostatic brakes such that they maintain their low weight and power consumption and gain the ability to control the position of each individual joint. In designing an electrostatic brake equipped joint, we use a rack and pinion transmission to convert rotational motion of the joint into linear sliding between the brake's electrodes. This design

optimizes electrode performance as evidenced by the joint being able to achieve the braking strengths predicted by its theoretical model (which assumes perfect electrode conformance). When comparing to joints driven by electromechanical motors, we find that our joint has four times less weight and one thousand times less power consumption. Furthermore, we use these joints to build a ten degree of freedom robot with only a single motor that is capable of manipulating multiple objects simultaneously.

In the final part of this thesis, we combine electrostatic braking and optical proximity sensing into a single robot hand. We implement a particle filter for optical sensor based pose estimation and adapt model predictive path integral control for our hybrid motor-brake actuation architecture. Our robot uses these modules to perform an in-hand manipulation task in which it must move a cylindrical object from one side of its workspace to the other. Relative to conventional underactuation (i.e. no brakes), we find that the use of brakes significantly increases the speed and precision with which the robot moves the object to the desired position. Furthermore, proximity sensor based pose estimation more accurately tracks the object's location throughout the manipulation and facilitates lower final positioning error relative to tactile sensor based pose estimation.

We conclude this thesis by suggesting directions for future research based on this work.

High Density Proximity, Contact, and Force Sensors

This thesis presents various iterations of proximity sensors embedded in the fingertips of robots. Unfortunately, the spatial resolution of our sensors is limited by the package size of the individual sensing modules that we use to make optical measurements. This results in relatively low spatial resolutions on the order of centimeters at best. Although they require contact to make measurements, Gel Sight type sensors can achieve spatial resolution on the order of tens of microns [Yuan et al., 2017]. Rather than relying on individual sensing modules, these sensors use cameras equipped with dense pixel arrays to capture light reflected off of the inner surface of an encapsulating gel. Similarly, future fingertip embedded proximity sensors can leverage infrared emitters and infrared cameras to achieve

similar spatial resolutions. Such designs should carefully consider the spatial relationship between the emitter(s) and camera(s) in order to ensure that the minimum sensing range is near, or ideally below, the surface of the fingertip. While it may be difficult to use time-of-flight methods at such a short range, infrared stereo methods could be a feasible solution for estimating depth.

Sensor Fusion for Robust Perception

In this thesis, we develop new proximity sensors and compare their performance to cameras and tactile sensors. We emphasize the performance of individual sensing modalities, but the most robust perception systems will be those that leverage multiple forms of sensing. Throughout a manipulation task, each sub-task may call for a different type of sensor. For example, a robot that wants to manipulate an object placed on a table top may first use a head-mounted camera to roughly estimate the pose of the object and estimate its surface properties. If the robot then makes contact with the object and wants to use the proximity, contact, and force sensor presented in Chapter 5 to estimate the amount of force it is applying, the surface property information inferred from its initial camera-based observations would be useful. Even within just the proximity sensing regime, different modalities can complement each others strengths and weaknesses [Yang et al., 2017]. For more information on the advantages and disadvantages of individual proximity sensing modalities, please see Navarro et al. [2021].

Electrostatic Actuation of Highly Articulated Mechanisms

While we demonstrate that the use of electrostatic brakes can replace most of a fully actuated robot's motors and still maintain the ability to control the position of individual joints, at least one motor is still required to make the robot move. This is because electrostatic brakes cannot themselves exert force, they can only resist it. The remaining motor(s) will represent the vast majority of the robot's (actuation) power consumption. Rather than using hybrid motor-brake actuation, future robots could further reduce power consumption by multiple

orders of magnitude by replacing the remaining motor(s) with dielectric elastomer actuators (DEAs) [Pelrine et al., 2000]. The functionality of both DEAs and electrostatic brakes depend on the generation of electroattractive force; DEAs use electroattractive force to strain a compressible material and thereby perform physical work, while electrostatic brakes use electroattractive force to create a frictional force between two bodies. Accordingly, the same or similar control circuitry could potentially be used to operate both technologies. Beyond reductions in power consumption, soft robotic systems would particularly benefit from hybrid DEA-brake actuation. Unlike rigid electromechanical motors, both DEAs and electrostatic brakes are typically composed of thin, flexible materials suitable for integration into the body of a soft robot.

BIBLIOGRAPHY

D. Álvarez, M. A. Roa, and L. Moreno. Tactile-based in-hand object pose estimation. In *Iberian Robotics conference*, pages 716–728. Springer, 2017.

O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

D. M. Aukes, B. Heyneman, J. Ulmen, H. Stuart, M. R. Cutkosky, S. Kim, P. Garcia, and A. Edsinger. Design and testing of a selectively compliant underactuated hand. *The International Journal of Robotics Research*, 33(5):721–735, 2014.

P. Azad, T. Asfour, and R. Dillmann. Stereo-based 6d object localization for grasping with humanoid robot systems. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 919–924. IEEE, 2007.

K. Baizid, G. Giglio, F. Pierri, M. A. Trujillo, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero. Behavioral control of unmanned aerial vehicle manipulator systems. *Autonomous Robots*, 41(5):1203–1220, 2017.

A. Bhatt, A. Sieler, S. Puhlmann, and O. Brock. Surprisingly robust in-hand manipulation: An empirical study. *arXiv preprint arXiv:2201.11503*, 2022.

A. Billard and D. Kragic. Trends and challenges in robot manipulation. *Science*, 364(6446), 2019.

W. G. Bircher, A. S. Morgan, and A. M. Dollar. Complex manipulation with a simple robotic hand through contact breaking and caging. *Science Robotics*, 6(54):eabd2666, 2021.

R. Buckingham, V. Chitrakaran, R. Conkie, G. Ferguson, A. Graham, A. Lazell, M. Lichon, N. Parry, F. Pollard, A. Kayani, et al. Snake-arm robots: a new approach to aircraft assembly. Technical report, SAE Technical Paper, 2007.

I. M. Bullock, R. R. Ma, and A. M. Dollar. A hand-centric classification of human and robot dexterous manipulation. *IEEE transactions on Haptics*, 6(2):129–144, 2012.

M. Chalon, J. Reinecke, and M. Pfanne. Online in-hand object localization. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2977–2984. IEEE, 2013.

R. Chamara and R. Gopura. An under-actuated mechanism for anthropomorphic robotic prosthetic hand. In *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, pages 162–166. IEEE, 2019.

K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

N. Chavan-Dafle and A. Rodriguez. Prehensile pushing: In-hand manipulation with push-primitives. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6215–6222. IEEE, 2015.

C. Corcoran and R. Platt. A measurement model for tracking hand-object state during dexterous manipulation. In *2010 IEEE International Conference on Robotics and Automation*, pages 4302–4308. IEEE, 2010.

N. C. Dafle, A. Rodriguez, R. Paolini, B. Tang, S. S. Srinivasa, M. Erdmann, M. T. Mason, I. Lundberg, H. Staab, and T. Fuhlbrigge. Extrinsic dexterity: In-hand manipulation with external forces. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1578–1585. IEEE, 2014.

S. Diller, C. Majidi, and S. H. Collins. A lightweight, low-power electroadhesive clutch and spring for exoskeleton actuation. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 682–689. IEEE, 2016.

L. Dissado, G. Mazzanti, and G. Montanari. The role of trapped space charges in the electrical aging of insulating materials. *IEEE Transactions on Dielectrics and Electrical Insulation*, 4(5):496–506, 1997.

B. H. Do, V. Banashek, and A. M. Okamura. Dynamically reconfigurable discrete distributed stiffness for inflated beam robots. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9050–9056. IEEE, 2020.

E. Donlon, S. Dong, M. Liu, J. Li, E. Adelson, and A. Rodriguez. Gelslim: A high-resolution, compact, robust, and calibrated tactile-sensing finger. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1927–1934. IEEE, 2018.

P. Falco, A. Attawia, M. Saveriano, and D. Lee. On policy learning robust to irreversible events: An application to robotic in-hand manipulation. *IEEE Robotics and Automation Letters*, 3(3):1482–1489, 2018.

L.-M. Faller, C. Stetco, and H. Zangl. Design of a novel gripper system with 3d-and inkjet-printed multimodal sensors for automated grasping of a forestry robot. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5620–5627. IEEE, 2019.

C. Fang, D. Wang, D. Song, and J. Zou. Toward fingertip non-contact material recognition and near-distance ranging for robotic grasping. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4967–4974. IEEE, 2019.

A. Firouzeh, M. Salerno, and J. Paik. Stiffness control with shape memory polymer in underactuated robotic origamis. *IEEE Transactions on Robotics*, 33(4):765–777, 2017.

S. Funabashi, A. Schmitz, T. Sato, S. Somlor, and S. Sugano. Robust in-hand manipulation of variously sized and shaped objects. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 257–263. IEEE, 2015.

R. Girshick. py-faster-rcnn. https://github.com/rbgirshick/py-faster-rcnn, 2015.

D. Göger, H. Alagi, and H. Wörn. Tactile proximity sensors for robotic applications. In *2013 IEEE International Conference on Industrial Technology (ICIT)*, pages 978–983. IEEE, 2013.

C. Gosselin, F. Pelletier, and T. Laliberte. An anthropomorphic underactuated robotic hand with 15 dofs and a single actuator. In *2008 IEEE International Conference on Robotics and Automation*, pages 749–754. IEEE, 2008.

G. Grioli, M. Catalano, E. Silvestro, S. Tono, and A. Bicchi. Adaptive synergies: an approach to the design of under-actuated robotic hands. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1251–1256. IEEE, 2012.

D. Guo, P. Lancaster, L.-T. Jiang, F. Sun, and J. R. Smith. Transmissive optical pretouch sensing for robotic grasping. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 5891–5897. IEEE, 2015.

J. Guo, C. Xiang, and J. Rossiter. A soft and shape-adaptive electroadhesive composite gripper with proprioceptive and exteroceptive capabilities. *Materials & Design*, 156:586–587, 2018.

J. Guo, J. Leng, and J. Rossiter. Electroadhesion technologies for robotics: A comprehensive review. *IEEE Transactions on Robotics*, 36(2):313–327, 2019.

I. Ha, Y. Tamura, H. Asama, J. Han, and D. W. Hong. Development of open humanoid platform darwin-op. In *SICE Annual Conference 2011*, pages 2178–2181. IEEE, 2011.

K. Hang, M. Li, J. A. Stork, Y. Bekiroglu, F. T. Pokorny, A. Billard, and D. Kragic. Hierarchical fingertip space: A unified framework for grasp planning and in-hand grasp adaptation. *IEEE Transactions on robotics*, 32(4):960–972, 2016.

K. Hang, W. G. Bircher, A. S. Morgan, and A. M. Dollar. Manipulation for self-identification, and self-identification for better manipulation. *Science Robotics*, 6(54):eabe1321, 2021.

B. He, S. Wang, and Y. Liu. Underactuated robotics: a review. *International Journal of Advanced Robotic Systems*, 16(4):1729881419862164, 2019.

R. Higo, Y. Yamakawa, T. Senoo, and M. Ishikawa. Rubik's cube handling using a high-speed multi-fingered hand and a high-speed vision system. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6609–6614. IEEE, 2018.

R. Hinchet and H. Shea. High force density textile electrostatic clutch. *Advanced Materials Technologies*, 5(4):1900895, 2020.

R. Hinchet, V. Vechev, H. Shea, and O. Hilliges. Dextres: Wearable haptic feedback for grasping in vr via a thin form-factor electrostatic brake. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, pages 901–912, 2018.

D. Holz, A. E. Ichim, F. Tombari, R. B. Rusu, and S. Behnke. Registration with the point cloud library: A modular framework for aligning in 3-d. *IEEE Robotics & Automation Magazine*, 22(4):110–124, 2015.

K. Hsiao, P. Nangeroni, M. Huber, A. Saxena, and A. Y. Ng. Reactive grasping using optical proximity sensors. In *2009 IEEE International Conference on Robotics and Automation*, pages 2098–2105. IEEE, 2009.

S. C. Jacobsen, H. Ko, E. K. Iversen, and C. C. Davis. Antagonistic control of a tendon driven manipulator. In *1989 IEEE International Conference on Robotics and Automation*, pages 1334–1335. IEEE Computer Society, 1989.

X. Ji, X. Liu, V. Cacucciolo, M. Imboden, Y. Civet, A. El Haitami, S. Cantin, Y. Perriard, and H. Shea. An autonomous untethered fast soft robotic insect driven by low-voltage dielectric elastomer actuators. *Science Robotics*, 4(37):eaaz6451, 2019.

L.-T. Jiang and J. R. Smith. Seashell effect pretouch sensing for robotic grasping. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 2851–2858. IEEE, 2012.

Y. Jiang, S. Moseson, and A. Saxena. Efficient grasping from rgbd images: Learning using a new rectangle representation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3304–3311. IEEE, 2011.

A. Johnsen and K. Rahbek. A physical phenomenon and its applications to telegraphy, telephony, etc. *Journal of the Institution of Electrical Engineers*, 61(320):713–725, 1923.

L. A. Jones and S. J. Lederman. *Human hand function*. Oxford university press, 2006.

M. E. Karagozler, J. D. Campbell, G. K. Fedder, S. C. Goldstein, M. P. Weller, and B. W. Yoon. Electrostatic latching for inter-module adhesion, power transfer, and communication in modular robots. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2779–2786. IEEE, 2007.

K. Karakasiliotis. Legged locomotion with spinal undulations. Technical report, EPFL, 2013.

C. C. Kemp, A. Edsinger, and E. Torres-Jara. Challenges for robot manipulation in human environments. *IEEE Robotics and Automation Magazine*, 14(1):20, 2007.

K. Koganezawa and M. Yamazaki. Mechanical stiffness control of tendon-driven joints. In *Proceedings 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients (Cat. No. 99CH36289)*, volume 2, pages 818–825. IEEE, 1999.

J. Konstantinova, A. Stilli, A. Faragasso, and K. Althoefer. Fingertip proximity sensor with realtime visual-based calibration. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 170–175. IEEE, 2016.

G. P. Kontoudis, M. V. Liarokapis, A. G. Zisimatos, C. I. Mavrogiannis, and K. J. Kyriakopoulos. Open-source, anthropomorphic, underactuated robot hands with a selectively lockable differential mechanism: Towards affordable prostheses. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 5857–5862. IEEE, 2015.

M. C. Koval, N. S. Pollard, and S. S. Srinivasa. Pose estimation for planar contact manipulation with manifold particle filters. *The International Journal of Robotics Research*, 34 (7):922–945, 2015.

K. Koyama, K. Murakami, T. Senoo, M. Shimojo, and M. Ishikawa. High-speed, small-deformation catching of soft objects based on active vision and proximity sensing. *IEEE Robotics and Automation letters*, 4(2):578–585, 2019.

N. E. Krausz, R. A. Rorrer, et al. Design and fabrication of a six degree-of-freedom open source hand. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24 (5):562–572, 2015.

V. Kumar, A. Gupta, E. Todorov, and S. Levine. Learning dexterous manipulation policies from experience and imitation. *arXiv preprint arXiv:1611.05095*, 2016.

M. Lambeta, P.-W. Chou, S. Tian, B. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer, et al. Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation. *IEEE Robotics and Automation Letters*, 5(3):3838–3845, 2020.

P. Lancaster, B. Yang, and J. R. Smith. Improved object pose estimation via deep pre-

touch sensing. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 2448–2455. IEEE, 2017.

P. Lancaster, P. Gyawali, C. Mavrogiannis, S. S. Srinivasa, and J. R. Smith. Proximity sensing for pose estimation during in-hand manipulation. In *In Submission*, 2022a.

P. Lancaster, C. Mavrogiannis, S. S. Srinivasa, and J. R. Smith. Electrostatic brakes enable individual joint control of underactuated, highly articulated robots. *In Submission*, 2022b.

P. E. Lancaster, J. R. Smith, and S. S. Srinivasa. Improved proximity, contact, and force sensing via optimization of elastomer-air interface geometry. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3797–3803. IEEE, 2019.

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

R. Li, R. Platt, W. Yuan, A. ten Pas, N. Roscup, M. A. Srinivasan, and E. Adelson. Localization and manipulation of small parts using gelsight tactile sensing. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 3988–3993. IEEE, 2014.

J. Liang, A. Handa, K. Van Wyk, V. Makoviychuk, O. Kroemer, and D. Fox. In-hand object pose tracking via contact feedback and gpu-accelerated robotic simulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6203–6209. IEEE, 2020.

P. Liljebäck and R. Mills. Eelume: A flexible and subsea resident imr vehicle. In *Oceans 2017-Aberdeen*, pages 1–4. IEEE, 2017.

H. Liu, K. Wu, P. Meusel, N. Seitz, G. Hirzinger, M. Jin, Y. Liu, S. Fan, T. Lan, and Z. Chen. Multisensory five-finger dexterous hand: The dlr/hit hand ii. In *2008 IEEE/RSJ international conference on intelligent robots and systems*, pages 3692–3697. IEEE, 2008.

R. Ma and A. Dollar. Yale openhand project: Optimizing open-source hand designs for ease of fabrication and adoption. *IEEE Robotics & Automation Magazine*, 24(1):32–40, 2017.

S. Ma. Analysis of snake movement forms for realization of snake-like robots. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, volume 4, pages 3007–3013. Ieee, 1999.

V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

A. Maldonado, H. Alvarez, and M. Beetz. Improving robot manipulation through fingertip perception. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 2947–2954. IEEE, 2012.

M. Manti, V. Cacucciolo, and M. Cianchetti. Stiffening in soft robotics: A review of the state of the art. *IEEE Robotics & Automation Magazine*, 23(3):93–106, 2016.

B. Mayton, L. LeGrand, and J. R. Smith. An electric field pretouch system for grasping and co-manipulation. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 831–838. IEEE, 2010.

B. Mazzolai, C. Laschi, M. Cianchetti, F. Patane, L. Bassi-Luciani, I. Izzo, and P. Dario. Biorobotic investigation on the muscle structure of an octopus tentacle. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1471–1474. IEEE, 2007.

W. McMahon, H. Birdsall, G. Johnson, and C. Camilli. Degradation studies of polyethylene terephthalate. *Journal of Chemical and Engineering Data*, 4(1):57–79, 1959.

J. Mens and A. De Gee. Friction and wear behaviour of 18 polymers in contact with steel in environments of air and water. *Wear*, 149(1-2):255–268, 1991.

G. Montanari. The electrical degradation threshold of polyethylene investigated by space charge and conduction current measurements. *IEEE Transactions on Dielectrics and Electrical Insulation*, 7(3):309–315, 2000.

A. Morgan, K. Hang, B. Wen, K. E. Bekris, and A. Dollar. Complex in-hand manipulation via compliance-enabled finger gaiting and multi-modal planning. *IEEE Robotics and Automation Letters*, 2022.

S. Muhlbacher-Karrer, A. Gaschler, and H. Zangl. Responsive fingers—capacitive sensing during object manipulation. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 4394–4401. IEEE, 2015.

A. Murta. A general polygon clipping library. *Advanced Interfaces Group, Department of Computer Science, University of Manchester, Manchester, UK*, 2000.

A. Nagabandi, K. Konolige, S. Levine, and V. Kumar. Deep dynamics models for learning dexterous manipulation. In *Conference on Robot Learning*, pages 1101–1112. PMLR, 2020.

S. E. Navarro, S. Mühlbacher-Karrer, H. Alagi, H. Zangl, K. Koyama, B. Hein, C. Duriez, and J. R. Smith. Proximity perception in human-centered robotics: A survey on sensing systems and applications. *IEEE Transactions on Robotics*, 2021.

K. Or, M. Tomura, A. Schmitz, S. Funabashi, and S. Sugano. Interpolation control posture design for in-hand manipulation. In *2015 IEEE/SICE International Symposium on System Integration (SII)*, pages 187–192. IEEE, 2015.

M. Orsag, C. Korpela, S. Bogdan, and P. Oh. Valve turning using a dual-arm aerial manipulator. In *2014 international conference on unmanned aircraft systems (ICUAS)*, pages 836–841. IEEE, 2014.

R. Patel and N. Correll. Integrated force and distance sensing using elastomer-embedded commodity proximity sensors. In *Robotics: Science and systems*, 2016.

R. Patel, R. E. Cox, and N. Correll. Integrated force and distance sensing using elastomer-embedded commodity proximity sensors. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2017.

R. Patel, R. Cox, and N. Correll. Integrated proximity, contact and force sensing using elastomer-embedded commodity proximity sensors. *Autonomous Robots*, pages 1–16, 2018.

R. Pelrine, R. Kornbluh, Q. Pei, and J. Joseph. High-speed electrically actuated elastomers with strain greater than 100%. *Science*, 287(5454):836–839, 2000.

J. M. Phillips, R. Liu, and C. Tomasi. Outlier robust icp for minimizing fractional rmsd. In *3-D Digital Imaging and Modeling, 2007. 3DIM'07. Sixth International Conference on*, pages 427–434. IEEE, 2007.

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

R. Platt, F. Permenter, and J. Pfeiffer. Using bayesian filtering to localize flexible materials during manipulation. *IEEE Transactions on Robotics*, 27(3):586–598, 2011.

M. Plooij, G. Mathijssen, P. Cherelle, D. Lefeber, and B. Vanderborght. Lock your robot: A review of locking devices in robotics. *IEEE Robotics & Automation Magazine*, 22(1): 106–117, 2015.

P. Polygerinos, N. Correll, S. A. Morin, B. Mosadegh, C. D. Onal, K. Petersen, M. Cianchetti, M. T. Tolley, and R. F. Shepherd. Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction. *Advanced Engineering Materials*, 19(12):1700016, 2017.

V. Pradeep, K. Konolige, and E. Berger. Calibrating a multi-arm multi-sensor robot: A bundle adjustment approach. In *Experimental robotics*, pages 211–225. Springer, 2014.

H. Prahlad, R. Pelrine, S. Stanford, J. Marlow, and R. Kornbluh. Electroadhesive robots—wall climbing robots enabled by a novel, robust, and electrically controllable adhesion technology. In *2008 IEEE international conference on robotics and automation*, pages 3028–3033. IEEE, 2008.

J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1316–1322. IEEE, 2015.

S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

L. Riano. pr2 rubiks solver, 2011. URL https://github.com/uu-isrc-robotics/pr2_rubiks_solver.

T. Rokicki, H. Kociemba, M. Davidson, and J. Dethridge. The diameter of the rubik's cube group is twenty. *SIAM Review*, 56(4):645–670, 2014.

J.-P. Rospars and N. Meyer-Vernet. Force per cross-sectional area from molecules to muscles: a general property of biological motors. *Royal Society open science*, 3(7):160313, 2016.

R. B. Rusu and S. Cousins. 3d is here: Point cloud library (pcl). In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1–4. IEEE, 2011.

K. Sasaki, K. Koyama, A. Ming, M. Shimojo, R. Plateaux, and J.-Y. Choley. Robotic grasping using proximity sensors for detecting both target object and support surface. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2925–2932. IEEE, 2018.

L. Savioli, G. Sguotti, A. Francesconi, F. Branz, J. Krahn, and C. Menon. Morphing electroadhesive interface to manipulate uncooperative objects. In *Sensors and Smart Struc-*

tures Technologies for Civil, Mechanical, and Aerospace Systems 2014, volume 9061, page 906129. International Society for Optics and Photonics, 2014.

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

S. Schulz. First experiences with the vincent hand. In *Myoelectric Symposium*, 2011.

J. Shintake, S. Rosset, B. Schubert, D. Floreano, and H. Shea. Versatile soft grippers with intrinsic electroadhesion based on multifunctional polymer actuators. *Advanced materials*, 28(2):231–238, 2016.

Proximity and ambient light sensing (ALS) module. ST Microelectronics, 8 2014. Rev. 6.

B. Steder, R. B. Rusu, K. Konolige, and W. Burgard. Point feature extraction on 3d range scans taking into account object boundaries. In *Robotics and automation (icra), 2011 ieee international conference on*, pages 2601–2608. IEEE, 2011.

M. Tavakoli and A. T. de Almeida. Adaptive under-actuated anthropomorphic hand: Isr-softhand. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1629–1634. IEEE, 2014.

S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial intelligence*, 128(1-2):99–141, 2001.

A. A. Transeth, K. Y. Pettersen, and P. Liljebäck. A survey on snake robot modeling and locomotion. *Robotica*, 27(7):999–1015, 2009.

L.-W. Tsai. Design of tendon-driven manipulators. Technical report, 1995.

Y.-T. Tu. Ray optics simulation. https://ricktu288.github.io/ray-optics/, 2018.

P. Tuffield and H. Elias. The shadow robot mimics human actions. *Industrial Robot: An International Journal*, 2003.

H. Van Hoof, T. Hermans, G. Neumann, and J. Peters. Learning robot in-hand manipulation with tactile features. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 121–127. IEEE, 2015.

G. Williams, N. Wagener, B. Goldfain, P. Drews, J. M. Rehg, B. Boots, and E. A. Theodorou. Information theoretic mpc for model-based reinforcement learning. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1714–1721. IEEE, 2017.

R. Wistort and J. R. Smith. Electric field servoing for robotic manipulation. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 494–499. IEEE, 2008.

C. Wright, A. Buchan, B. Brown, J. Geist, M. Schwerin, D. Rollinson, M. Tesch, and H. Choset. Design and architecture of the unified modular snake robot. In *2012 IEEE International Conference on Robotics and Automation*, pages 4347–4354. IEEE, 2012.

Z. Xu. The functional capacity of the humanlike robotic hands. In *Hand Function*, pages 291–312. Springer, 2019.

Z. Xu and E. Todorov. Design of a highly biomimetic anthropomorphic robotic hand towards artificial limb regeneration. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3485–3492. IEEE, 2016.

Z. Xu, V. Kumar, Y. Matsuoka, and E. Todorov. Design of an anthropomorphic robotic finger system with biomimetic artificial joints. In *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 568–574. IEEE, 2012.

B. Yang, P. Lancaster, and J. R. Smith. Pre-touch sensing for sequential manipulation. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 5088–5095. IEEE, 2017.

W. Yuan, S. Dong, and E. H. Adelson. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762, 2017.

K. Zhang and S. Follmer. Electrostatic adhesive brakes for high spatial resolution refreshable 2.5 d tactile shape displays. In *2018 IEEE Haptics Symposium (HAPTICS)*, pages 319–326. IEEE, 2018.

J. Zhong, M. Oller, T. Power, and K. Taekyung. PyTorch MPPI, 12 2019. URL https://github.com/UM-ARM-Lab/pytorch_mppi.

H. Zhu, A. Gupta, V. Kumar, A. Rajeswaran, and S. Levine. Dexterous manipulation with reinforcement learning: Efficient, general, and low-cost, Aug 2018. URL https://bair.berkeley.edu/blog/2018/08/31/dexterous-manip/.

H. Zhu, A. Gupta, A. Rajeswaran, S. Levine, and V. Kumar. Dexterous manipulation with deep reinforcement learning: Efficient, general, and low-cost. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3651–3657. IEEE, 2019.