11-13-2014

Task-based Robotic Grasp Planning

Yun Lin
University of South Florida, gtlinyun@gmail.com

Task-based Robotic Grasp Planning

by

Yun Lin

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Computer Science and Engineering
College of Engineering
University of South Florida

Major Professor: Yu Sun, Ph.D.
Lawrence O. Hall, Ph.D.
Luther Palmer III, Ph.D.
Rajiv V. Dubey, Ph.D.
Gangaram S. Ladde, Ph.D.

Date of Approval:
November 13, 2014

Keywords: Task Modeling, Grasp Optimization, Grasp Quality Measure,
Disturbance Wrench, Manipulative Motion

Acknowledgments

Foremost, I would like to gratefully and sincerely thank my advisor Prof. Yu Sun for his guidance, motivation and patience during my Ph.D study and research. His support helped me throughout my entire research and dissertation process. In his lab, I was given the opportunity to be involved in the intriguing world of robotics. His experience in the field gave me the chance to face interdisciplinary and demanding projects that helped me develop my background. His guidance and advice, both in a technical and in a personal level were crucial for my success. His inspiring and motivational personality as well as his trust in me strengthened my decision to follow my dream of pursuing a PhD and a career in this field.

Besides my advisor, I would like to extend my sincere gratitude to the remaining of my dissertation committee: Dr. Lawrence O. Hall, Dr. Luther Palmer, Dr. Rajiv V. Dubey, Dr. Gangaram S. Ladde for their careful reading of my dissertation. Their valuable discussions, insightful comments and hard questions helped improve the writing of my dissertation.

I thank my fellow labmates in RPAL Lab: Bingxiong Lin, James Robe, Yongqiang Huang, Adrian Johnson, Wei Dai for all the good times we had in my PhD study, and for their inspirational advice.

Finally, and most importantly, I would like to thank my parents, and my husband Zhou Ma for their priceless moral, emotional and economical support and understanding during the whole period of my studies. I consider myself as really lucky to be a member of such a peaceful and cultivated family.

Table of Contents

List of Tables

List of Figures

Abstract

Grasp should be selected intelligently to fulfill different stability properties and manipulative requirements. Currently, most grasping approaches consider only pick-and-place tasks without any physical interaction with other objects or the environment, which are common in an industry setting with limited uncertainty. When robots move to our daily-living environment and perform a broad range of tasks in an unstructured environment, all sorts of physical interactions will occur, which will result in random physical interactive wrenches: forces and torques on the tool.

In addition, for a tool to perform a required task, certain motions need to occur. We call it "functional tool motion," which represents the innate function of the tool and the nature of the task. Grasping with a robotic hand gives flexibility in "mounting" the tool onto the robotic arm – a different grasp will connect the tool to the robotic arm with a different hand posture, then the inverse kinematics approach will result in a different joint motion of the arm in order to achieve the same functional tool motion. Thus, the grasp and the functional tool motion decide the manipulator's motion, as well as the effort to achieve the motion.

Therefore, we propose to establish two objectives to serve the purpose of a grasp: the grasp should maintain a firm grip and withstand interactive wrenches on the tool during the task; and the grasp should enable the manipulator to carry out the task most efficiently with little motion effort, and then search for a grasp to optimize both objectives. For this purpose, two grasp criteria are presented to evaluate the grasp: the task wrench coverage criterion and the task motion effort criterion. The two grasp criteria are used as objective functions to search for the optimal grasp for grasp planning.

To reduce the computational complexity of the search in high-dimensional robotic hand configuration space, we propose a novel grasp synthesis approach that integrates two human grasp

strategies – grasp type, and thumb placement (position and direction) – into grasp planning. The grasping strategies abstracted from humans should meet two important criteria: they should reflect the demonstrator's intention, and they should be general enough to be used by various robotic hand models. Different abstractions of human grasp constrain the grasp synthesis and narrow down the solutions of grasp generation to different levels. If a strict constraint is imposed, such as defining all contact points of the fingers on the object, the strategy loses flexibility and becomes rarely achievable for a robotic hand with a different kinematic model. Thus, the choice of grasp strategies should balance the learned constraints and required flexibility to accommodate the difference between a human hand and a robotic hand. The human strategies of grasp type and thumb placement have such a balance while conveying important human intents to the robotic grasping.

The proposed approach has been thoroughly evaluated both in simulation and on a real robotic system for multiple objects that would be encountered in daily living.

Chapter 1

Introduction

As the role of robots in assisting humans in their everyday life continues to develop, one of the primary goals of a robot is to assist humans to manipulate objects and tools. One important aspect that affects object manipulation is the grasp of a target object. Before manipulating an object, the robot needs to determine how the object is gripped in the hand, so the object can be stably held by the hand throughout the manipulation procedure. Currently, most grasping approaches compute grasps only for simple pick-and-place tasks. However, with the rising need to design general purpose robotic manipulators capable of assisting people in daily life to perform a large variety of manipulative tasks, such as writing with a pen, pouring water from a cup, using a screwdriver, and relocating an object, etc., advanced approaches should be developed to find the appropriate grasp depending on the task to be executed.

Motivated by the above purpose, in this dissertation, we address the grasp synthesis problem: given an object and a task, choose an appropriate grasp to achieve the task requirements. The grasp synthesis problem leads to some core questions tackled in this dissertation:

1. How to model a manipulation task? A manipulation task is to be mathematically modeled, so candidate grasps can be computed given a task model.

2. How to evaluate a grasp? An appropriate grasp should be stable and reachable by the arm, as well as facilitate the desired manipulation task. To evaluate if a grasp meets the above requirements, suitable criteria must be constructed.

3. What algorithm is to be developed to efficiently search for suitable grasps? A grasp is described as the finger placements on the object and the configurations of the manipulator.

Searching for candidate grasps is done in a high dimensional space of the manipulator configurations. Therefore, efficient approaches need to be developed to establish a quality grasp.

1.1. Human Grasp

The human hand is the most dexterous hand for tool manipulations, when compared with other animals such as apes. It seems to be an easy task for human beings to manipulate tools or objects and intelligently choose the optimal grasp according to the manipulation. Thus, studying the human hand and how humans grasp objects for manipulation can help researchers to design the robotic manipulator and develop a grasping algorithm.

It is not surprising that a large number of robot hands are designed to be anthropomorphic, in hopes that the robot can perform the manipulation as intelligently as humans. Hence, we begin by studying human choices of grasps, in terms of the object to be manipulated and the manipulation to be performed.

Human grasping has been studied a lot in anthropology to facilitate research in various areas, such as prosthetics and robotic grasping. Humans adopt different hand shapes according to the size and shape of objects - they pinch a key, wrap a book, hook a bucket handle, spherically wrap a ball, etc. The grasp is decided even more by force and torque conditions as well as the dexterity requirements of the task. In [1], human grasp is characterized by two basic types for all prehensile activities: the precision grip, in which objects are gripped by only fingertips, and the power grip, in which objects are wrapped around by the thumb and some fingers. The precision grasp is usually applied in tasks that require little power, such as picking up a small object with the fingers. In contrast, the power grasp is applied with a large contact area between the hand and the object in tasks that require a large amount of power, such as striking with a hammer and picking up a heavy object. Derived from these two basic grips, Cutkosky and Wright [2] developed a hierarchical tree of grasp types to facilitate the design and control of robotic hands for various manipulation purposes. Other alternative grasp taxonomies were also suggested (e.g. [3]).

Grasp types are determined by hand configurations, and hand configurations should be associated with arm position and orientation relative to the object to assure contact of the hand with the object without slipping, with respect to external disturbance. Besides the stability requirement to prevent slipping, people usually choose grasps that facilitate subsequent manipulations, so the manipulation can be performed in a comfortable way, in terms of joint ability and power consumption. They tend to minimize muscle tension change and power consumption [4, 5, 6, 7]. People also manipulate the object with less movement-related effort, preferring to move their wrist and fingers rather than their arms for dexterous manipulation. The exception would be those tasks that require large force exerted from the arm, but in general, making object motion by moving the entire arm is often more difficult and time consuming. Taking rotational manipulation for example, such as turning a handle and using a screwdriver, the normal direction of the palm is typically aligned with the rotational axis of the object, so the turning motion involves little arm motion.

In robotics, this idea was introduced into the study of motion planning, where motion effort was used as a criterion to compute the optimal trajectory for a manipulator, given a start point and an end point of the end-effector. See examples of related research in [8, 9].

1.2. Our Approach

Similar to the way humans grasp objects, a grasp should be selected intelligently by the robot to fulfill different stability properties and manipulation requirements. Currently, most grasping approaches consider only pick-and-place tasks without any physical interaction with other objects or the environment. This is common in an industry setting where there is a low probability of unexpected events. For this general purpose, one typical approach is grasp planning. A typical grasp planning approach requires a geometrical model of the object and grasp model. Then it uses mathematical optimization to search for the optimal contact placement on an object, which gives rise to difficulty in choosing a quality criterion for the optimization procedure.

There are two main challenges in the grasp planning approach. Such an approach requires good knowledge of the system, which is difficult to obtain. In addition to knowledge of the object model, it requires the task to be executed, the kinematics of the manipulator, and the constraints from the outside environment that must be satisfied during the task execution. Moreover, the computational effort is also not trivial in searching for feasible grasps that satisfy all the physical conditions and facilitate successful task execution. Such an approach, however, is closer to the way humans make decisions based on knowledge of the outside world. Also, with the rapid development in other fields such as object recognition and pose estimation, 3D reconstruction, and 3D virtual reality and physical simulation, grasp planning has been a feasible approach to compute a successful grasp in real time.

In the context of this dissertation, we adopt the grasp planning approach to address the grasp synthesis problem. The key problem is to construct task-based grasp quality criteria for the grasp planning procedure, so the optimal grasp selected by the grasp planning algorithm should be subject to task requirements. Associated with the problem are task modeling to mathematically describe the task requirements, and the optimization algorithm to search for the optimal grasp efficiently.

1.2.1. Task Modeling

To construct a task-dependent grasp metric, one primary problem is to model the task. A manipulation task refers to "the process of moving or rearranging objects in the environment [10]". The task requirements can be divided into two parts: force requirement and motion requirement [11].

The force requirement refers to the interactive force that occurs during the manipulation. When robots work in our daily-living environment and perform a broad range of tasks in an unstructured environment, all sorts of physical interactions will occur, which will result in random physical interactive wrenches (in other words, force and torque) on the object. For a certain manipulation, the interactive wrench is not necessarily evenly distributed along every direction of the wrench space, but has different distributions. For example, for a task of lifting a plate to a shelf, the external wrench is distributed in a narrow cone along the gravity direction considering uncertainties. A grasp, then, does not need to be

force-closure (resisting force in all directions), but merely able to resist gravity with uncertainties. As a result, a quality grasp should be efficient in withstanding the main interactive wrench that occurred during the manipulation.

In addition, for a tool to perform a required task, certain motions need to occur; we call it "functional tool motion," which represents the innate function of the tool and the nature of the task. A different grasp will connect the tool to the robotic arm with a different pose, and then the inverse kinematics approach will result in a different joint motion to achieve the same functional tool motion. Thus, the grasp and the functional tool motion decide the manipulator's motion, as well as the effort to achieve the motion. At the same time, a different manipulator motion will have a different efficiency in transferring the motion from joints to the tool [12], resulting in different motion efforts to achieve the functional tool motion. As a result, the efficiency of the manipulation motion should be used to evaluate the grasp as well. Similar to humans, who tend to manipulate the tool with less movement-related effort, the robot's motion effort should be used to evaluate the grasp as well.

Due to the difficulty of modeling a task, few have considered task information in grasp planning [11, 13, 14]. The related work that considered force requirements in grasp planning includes [11, 14, 15, 16, 17]. Even less work has taken into account the motion to be executed. Instead of assuming the task to be known or empirically approximating the task, here, we model both force and motion requirements from human demonstration data. The task was demonstrated by users in a virtual reality environment. The users can interact with the environment via a haptic device, the Phantom OMNI, and have a haptic feel when they control the virtual object. The properties of the task, as well as external disturbance, were captured during a task demonstration, unlike in other work in which the experience is only estimated or approximated.

1.2.2. Grasp Quality Metrics

Given the task requirements of both force and motion, we propose to establish two objectives to serve the purpose of a grasp: the grasp should maintain a firm grip and withstand interactive wrench

applied to the tool during the task execution; and the grasp should enable the manipulator to carry out the task with little motion effort. Then, we search for a grasp to optimize both objectives. For this purpose, two grasp criteria are used to plan the grasp: the task wrench coverage criterion and the motion effort criterion.

The task wrench coverage criterion measures the ability of a grasp to resist task disturbance. Since the force requirement is modeled as task wrench distribution other than assuming an evenly-distributed task wrench space (TWS), it is possible that some wrenches occur more frequently in some areas than others, even if the wrenches that occur frequently have a smaller magnitude than the wrenches that rarely occur. The importance of the wrenches should be weighted by the distribution of the wrenches which have occurred, and more weight should be put on wrenches with a high distribution than wrenches that occur less frequently. As a result, different distributions would result in different grasps.

The task motion effort criterion is also introduced to measure the effort made by the manipulator to perform the manipulation motion. A grasp establishes a kinematical connection between the object and the manipulator. Then, given a functional tool motion sequence, the manipulator configurations can be computed via an inverse kinematic solver. If there is no solution for the inverse kinematic solver, then the grasp is not reachable throughout the whole manipulation motion. The joint torques can be computed via the system dynamics. The motion effort depends on the joint torque and distance covered by each joint motion, and joint torques are dependent on joint trajectory, velocity, and acceleration. Hence, low motion effort implies small joint torques, small arm motion, and small velocity changes (in other words a smooth motion).

1.2.3. Incorporation of Physical Constraints into Grasp Planning

Since a number of anthropomorphic hands have a high number of degrees of freedom (DOF) to be as powerful as the human hand, it introduces complexity to the optimization search. There is work which has focused on providing constraints in the search space to reduce the computational complexity in a high dimensional robotic hand configuration space. Besides interactive wrench and manipulation

motion, there are also additional tasks and grasp cues to constrain the grasp. For example, for the tools which have handles, including a knife, hammer, mug, etc., the grasp is preferred to be constrained to the handle. In addition, the surface of the tool in order to be in contact with the environment should not be blocked by the grasp. Research in [18-22] imposed appropriate contact points on the object by human teachers. The constraint on contact points, however, is assumed to be independent of the physical constraints of a given hand, which is not often true. It raises the problem of solving the inverse kinematics to satisfy the rigid constraints imposed by contact points [23]. Work presented in [24-27] abstracted an approach vector together with grasp type to map the location and pose of the robot hand from human demonstration. The approach vector can only be obtained by observing a human demonstration. It provides a good initial pose to start with, but it does not indicate which surface or part of the object is to be gripped.

Therefore, to avoid the problem that arises from the constraints of contact points and approach vector and instead combine the strength of them, we propose a novel grasp synthesis approach that integrates human grasp strategies – grasp type, and thumb placement (position and direction) – into grasp planning. We learn grasp types as the first grasp cue from human input, because grasp types provide important task-oriented experience – the manner in which a human grips an object according to specific manipulation purposes. An object can be grasped in different ways to conform to task requirements. One example is the procedure of screwing on a jar lid, which starts with a precision grasp using only the fingertips, because the jar lid is loose at the beginning and can be rotated efficiently and flexibly without much power. As the jar lid gets tighter, one may switch to a power grasp to apply a larger force on the jar lid.

Thumb placement is the second grasp cue from humans, for two main reasons. Firstly, almost all robotic hands have a long and strong opposable thumb, although robotic hands are usually simplified to different extents compared to the human hand. Secondly, the thumb plays a key role in gripping an object efficiently, according to anthropology research. Thumb placement indicates which surface or part of the

object to grip. Because of this, thumb placement is directly related to the task and confine the grasp, which constrains the workspace of the wrist and other fingers while preserving necessary flexibility to meet kinematic constraints.

The candidate grasp was computed from a set of given thumb positions on the object surface, as well as the directions in which the thumb should point. Thumb positions offer a general reference to be gripped on the object body; thumb directions provide constraints on wrist positions and orientations. The constraint of thumb placement can be labeled manually on the object or generated automatically from examples. Grasp type can be input by the human or recognized from a human demonstration. A novel recognition approach using trajectory data at the joint level will also be presented in this dissertation. We believe that grasp type and the thumb placement are simple, effective and informative representations of grasp features.

1.2.4. Grasp Optimization Algorithm

The general idea of the optimization scheme that we propose is summarized in Figure 1. A grasp can be described by the finger posture and wrist position. Grasp planning finds the best finger posture and wrist position to optimize the task-dependent grasp criteria.

The algorithm takes as input the task requirements (task wrench cluster and task motion sequence), system information (geometry and pose of the object, kinematic model of the manipulator, and the environment information), and additional grasp and task cues extracted from human experience (thumb placement and grasp types). They provide a set of feasible grasps. The feasible grasps are evaluated by the task wrench coverage measure and the task motion effort measure. The resulting grasp is intended to perform the task with ease and be able to resist task interactive wrenches.

The main contributions of this dissertation are summarized as follows:

- The manipulation motion to be performed with the grasp was introduced as a task requirement to the grasp planning algorithm.

- The task properties, including both force and motion requirements, were measured from human demonstration data, other than assumed to be known or estimated or approximated by experience as in previous related work.
- The force requirement is modeled on the distribution of the interactive wrenches, instead of assuming an evenly-distributed TWS or a task wrench ellipsoid.
- Novel task-oriented grasp quality metrics were constructed that take into consideration the interactive wrenches and the effort to perform the manipulation motion.
- Grasp types, as well as thumb placement (position and direction), were introduced as additional grasp and task cues and as physical cues for the grasp planning algorithm. The proposed constraints are independent of the physical hand models, which avoids the correspondence problem.
- A novel grasp type recognition algorithm was proposed based on a human-demonstrated grasping motion sequence rather than a static final grasping pose.

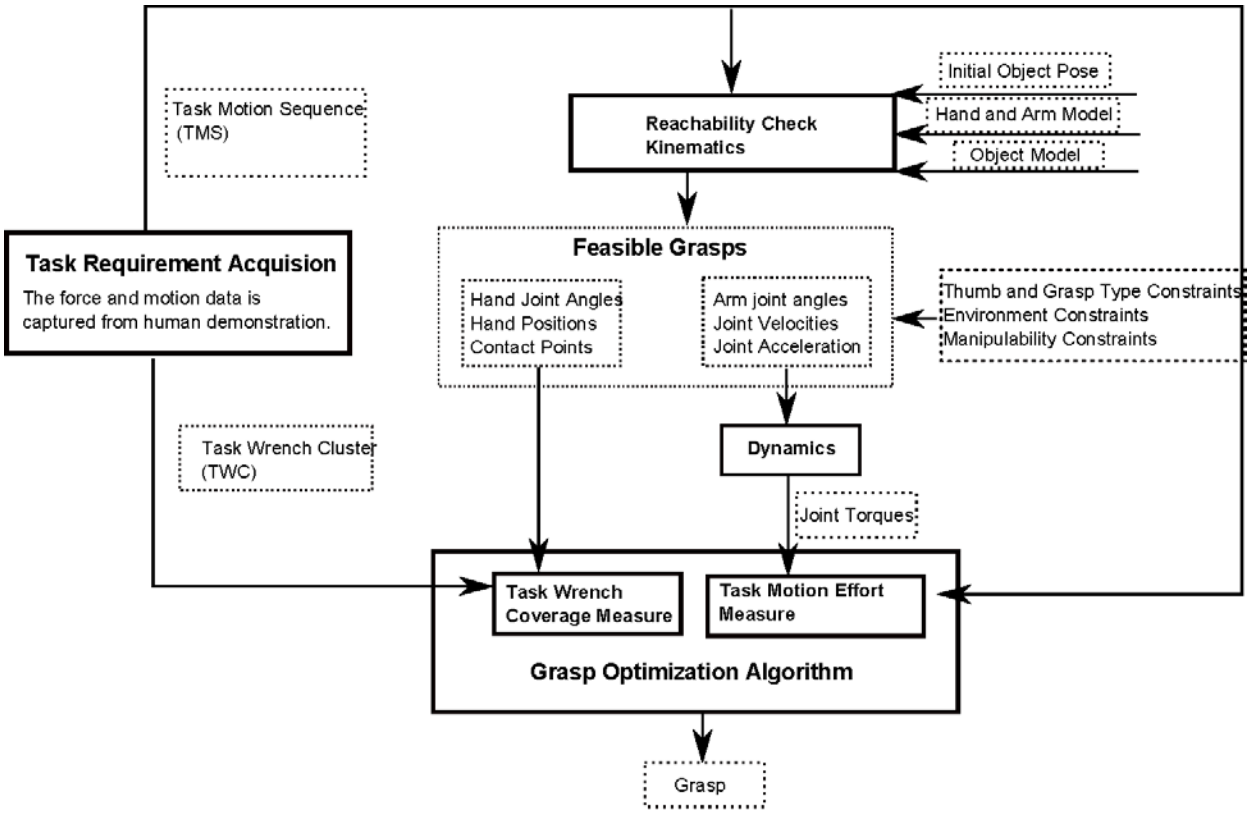


Figure 1. Diagram of the proposed grasp planning approach.

Chapter 2

Background and Related Work

The primary goals of robotic hands are manipulation and grasp. In robotics, manipulation and grasp have been active research areas studied over several decades. The manipulation goal is to relocate and reorient an object by a manipulator while the object interacts with the outside environment. Grasp by a mechanical hand is to immobilize the object in the hand and maintain contact under external disturbances during the manipulation process. The grasp problem has two aspects: grasp synthesis and grasp control. Grasp synthesis is the choice of grasps given the manipulation task. Following grasp synthesis is the grasp control problem – the control of the force and position of the manipulator to maintain the grasp while performing the manipulation in the presence of uncertainties. Each aspect is an active research area in its own right. In this dissertation, our work is focused on the former problem, grasp synthesis; we do not address the control problem that follows the grasp synthesis problem.

A large variety of robotic hands have been developed for various purposes during the last two decades. The robotic hands differ in levels of anthromorphism and dexterity, as discussed by Martell et al. [28], who reviewed the design of multifingered robotic hands. They can either be as complex as the anthropomorphic robotic hands for dexterous manipulation purposes, see for instance [29], or as simple as a 2-DOF gripper for manufacturing purposes. Grasp synthesis algorithms have been developed that are applicable for different robotic hands and dexterous levels of manipulations. Some earlier work has presented detailed surveys on grasp studies. Shimoga [30] reviewed grasp synthesis algorithms, including quality measures associated with hand configuration. Bicchi and Kumar [31] described the state of the art in robotic grasping, including a discussion about the relevance of grasp quality measures. In [13, 32, 33],

the authors gave detailed reviews of data-driven grasp synthesis approaches. In this chapter, a brief review is given to discuss the related approaches to addressing the grasp synthesis problem.

2.1. Background

The manipulation task is to be performed by a manipulator. The manipulator is referred to as a robotic arm equipped with a robotic hand at the end-effector of the arm. Therefore, a grasp G can be specified by the wrist position and orientation relative to the object, as well as the hand posture: $G(w,p)$ where w is a 6-D vector representing the 3-D position and 3-D orientation of the wrist, and hand posture $p(\theta_1,\theta_{2,\dots}\theta_D) \in \Re^{\mathrm{D}}$ is a vector defined by a set of D variables (such as D number of joint angles) that represent the hand configurations.

A grasp comprises multiple contact points between the hand and the grasped object. The object is grasped and manipulated by contacts with fingers. The contact locations are dependent on the hand posture and the wrist position with respect to the object, so an alternative description of a grasp is the contact. The contact model implies two transitions of motion and force associated with a grasp [34]. The first transition is the Jacobian of the hand, which relates joint velocity $\dot{\theta}$ to the velocities v at the contact points, and the transpose of the Jacobian relates the fingertip force f to the joint torque T:

$$v = J\dot{\theta} \tag{2.1}$$

$$T = J^T f \tag{2.2}$$

The second transition is the grasp matrix. If we group force and torque together in a wrench vector, the grasp matrix transmits the net wrench of the object from the wrench f produced on the fingertips through the contact points, and the transpose of the grasp matrix relates the object velocity $\dot{x}$ to the velocities produced by the contacts v:

$$W = Gf \tag{2.3}$$

$$v = G^T \dot{x} \tag{2.4}$$

The contact force can act only against the object (push but not pull). It can be decomposed as the normal force component, and the components of the tangential friction and torsional friction about the

normal. The contact model has three different types: point contact without friction, point contact with friction or rigid contact (PCwF), and soft contact. The second contact model, PCwF, is the most commonly used model. It assumes the contact can apply normal force and tangential friction. According to the Coulomb friction model, the tangential force is restricted in relation to the normal force of the friction coefficient μ to prevent slippage. All feasible contact forces are constrained to the friction cone aligned at the contact normal.

2.2. Related Work

Having introduced the grasp as described by two aspects – wrist position and hand posture as well as the contact model, the grasp synthesis approach aims to find an appropriate wrist position and hand posture associated with the contact model.

The approaches to grasp synthesis and the study of human grasp have been divided into two areas by Cutkosky et al. [32]: the analytical approach, considering physical and mechanical properties involved in grasping; and the empirical approach, or data-driven approach, trying to mimic the behavior of the human hand. The approaches in both categories were reviewed in detail by Sahbani et al. [13], who used the same classification of approaches.

2.2.1. Analytical Approach

The analytical approach or model-based approach models the grasp and task by the laws of physics, kinematics and dynamics. It intrinsically computes the grasp that satisfies the manipulation requirements, by considering physical and mechanical properties of the robot and the object. The approach relies on good quantitative knowledge from the grasp, including the system parameters of the robot, object geometry, the task property and the surrounding environment. As a result, it requires complicated computations to do the analysis but it is closer to the human approach of making decisions based on knowledge of both the internal and external world.

The analytic approach typically uses optimization theory to search for the optimal hand configurations or contact placements on an object, which gives rise to the problem of choosing a goal

function to be optimized; in other words, the quality criterion to evaluate the grasp. In the literature, the existing grasp quality measures can be associated with either hand configurations or contact locations.

The hand configuration refers to how the robotic hand should be configured for creating a specific posture, which includes position and orientation for the palm, spreading angle and opening/closing control for all fingers. The quality measures associated with hand configurations require the hand Jacobian J to quantify the quality. In order to keep the hand configuration out of singular configurations, it is desirable to maximize the smallest singular value of the hand Jacobian [30, 35].

Instead of considering only one singular value of J, in order to consider all the singular values of J, the volume of the ellipsoid through which the hand is manipulated is proposed as a quality measure. This grasp quality measure is referred to as a grasp manipulability, i.e. the ability of the manipulator to impart arbitrary motions at the end-effector [18]. The measure of manipulability can help avoid the singularity situation where there is a fairly low transition rate from joint velocity to the end-effectors velocity, but it does not consider the requirement of the task in the measure where only a few directions of motion are favored. The problem was addressed by Chiu [36], who defined a task compatibility index to measure the transformation ratio along the desired directions, instead of all directions.

When the grasp is modeled as contact positions, one considers the force being applied by fingers at the contact. The form-closure and force-closure are two of the most basic properties that a grasp should satisfy for achieving equilibrium. For form-closure grasps, the maintenance of the object's equilibrium is guaranteed by the geometric constraints from fingers, despite the externally applied wrench. Namely, the finger contacts prevent all motions of the grasped object; while for the force-closure grasps, the maintenance of the object's equilibrium requires the application of an externally applied wrench. Namely, the finger contacts do not prevent all infinitesimal motions of the grasped object. In order to resist any wrench during grasp tasks, most researchers defined the metrics for force-closure grasps in the formats of the force that needs to be applied in worst cases scenarios. The value of the metric is defined as the radius of the largest wrench space ball that fits within the unit grasp wrench

space [37]. Related research on computing the force-closure property was developed in [15, 38, 39, 40]. Other quality measures regarding the force requirement include internal forces and stability. Detailed reviews of grasp measures can be found in [31, 41, 42].

These quality measures are all task independent, where an equal force requirement in every direction is assumed. When executing a specific task, however, force requirement is not equal in every direction, but emphasized in some directions. Thus, it makes more sense to measure how efficiently a grasp can balance a special set of disturbances related to a desired task. Hence, alternatively, the force-closure property can be modified as task-dependent metrics, where the shape of the task wrench space is adaptable to a task instead of a uniformly distributed ball. Li and Sastry [11] used an ellipsoid in wrench space to better approximate a task to facilitate the computation of the quality measure. The grasp quality metric then measures the largest force in order to resist any disturbance in the task. In the description of quality metric, the task ellipsoid is assumed to be known already, but the problem of modeling a given task by an ellipsoid was quite complicated, as stated by the authors.

van der Stappen et al. [44, 45, 46] developed algorithms to compute two or three feasible contact locations capable of resisting a single given wrench and wrenches in its neighborhood. However, their work was not generalized to most of the manipulation tasks where multiple interactive wrenches may occur.

The authors in [17] developed an algorithm to compute the task-oriented quality measure. The algorithm specifies a primary direction in which the largest wrench is along.

Instead of evaluating a grasp by the contacts, the authors in [43] proposed an approach that predicts the grasp quality using hand-preshapes without knowing the contacts. Specifically, the robot aligned the task frame of the hand with a selected target frame to accomplish the task. A simplified model of the object (a hierarchy of boxes) and the manipulation direction were adapted to select the hand reshape and its corresponding target frame.

Pollard [15] proposed the object wrench space (OWS), which takes complete object geometry into consideration. The OWS integrates all disturbance wrenches that can be exerted anywhere on the object. Borst et al. [14] combined the idea of the task ellipsoid with the concept of the OWS. In order to reduce the computational complexity, the authors presented an algorithm to approximate the OWS by an ellipsoid and to measure how well the OWS ellipsoid can be fit into a GWS with a linear scale. The idea of the OWS is to take all possible disturbances into account, which is good for unknown tasks but is not task-specific; for a specific task, a grasp does not need to operate in the whole OWS but to operate within the required TWS, which should be a subset of the whole OWS, so the grasp can be optimally chosen for the specific task.

The task wrench space is known to be difficult to model. To obtain the TWS in reality, necessary sensors are required to measure the contact regions and contact normals in human demonstration, which remains a challenge. This is the main reason why most work empirically approximates the TWS rather than actually measure it.

Aleotti and Caselli [54] proposed a method to obtain the TWS by human demonstration. The user operated a virtual robotic hand to grasp the object in virtual reality. They used data gloves to map human hand to robotic hand workspace and captured TWS in the virtual environment. They also considered a database of candidate grasps, and grasps were evaluated by a task-related quality measure. However, the correspondence problem has been a crucial issue to map between two different configuration spaces of the human hand and the robotic hand.

Another difficulty of task-oriented grasp planning is the computational complexity of searching in a high-dimensional hand configuration space. It is, therefore, natural to introduce human experience to constrain the search space [47-49].

2.2.2. Data-driven Approach

The data-driven, or empirical approach, in contrast, tries to imitate grasping done by humans. Humans are capable of grasping and manipulation, and various robotic hands are designed to be

anthropomorphic to grasp and manipulate like humans. Thus, a large amount of research has emerged that generates grasps according to the data obtained from existing human grasp examples, which avoids the complicated computation of the mathematical models of the world.

Numerous work has been done to abstract various useful grasp features from human grasp. Grasp features can be learned by observing human demonstration. Perhaps the most straightforward way of learning from human demonstration is direct mapping, which can be further divided into finger joint space [50] or fingertip workspace [51]. Mapping in joint space finds the correspondence of joint angles in a robot's configuration space, which usually creates a hand pose for the robotic hand similar to the human hand pose. This is suitable for power grasps but not for precision grasps, because it may lead to low accuracy in fingertip positions. Mapping in fingertip space, on the other hand, is more suitable for precision grasps, because it computes the correspondence of fingertip positions in workspace. In [50], neural network techniques were used to calibrate the relationship between the Cartesian position of the fingertips and data glove readings. Mapping between a human hand and a robot hand is determined by a linear translation between the two workspaces. In [51], a point-to-point algorithm was used to map a human hand to a three-finger gripper in a teleoperation. However, in general, since most robotic hand designs simplify the mechanical structures to make control easier, some fingertip positions demonstrated by humans are not reachable by all robotic hands. Direct mapping imposes a strict constraint on the robot grasp, leading to a lack of flexibility that is required to compensate the kinematics difference between the robotic and human hands. The key points of the proposed techniques are sensors and signal processing. Data gloves are used by some researchers to map the human hand to the robotic hand workspace. Recently, estimating the demonstrator's hand posture from vision [49] became possible and accessible, and led to promising work that transferred the hand posture to robotic hands with dissimilar kinematics [52].

Instead of direct mapping, classification approaches have been developed to avoid the correspondence problem. Human grasps are classified into different grasp types, and human

demonstration is recognized as one of the grasp types [53-55]. Many of the grasp classifications are based on Cutkosky's grasp taxonomy [56]. Cutkosky's taxonomy classifies common user-performed grasps into 16 classes based on task requirements and dexterities. To recognize the demonstrated grasp as one type in Cutkosky's taxonomy, pattern recognition techniques can be applied. In [53], Hidden Markov models were used to recognize grasp type from the taxonomy based on an entire grasp sequence. A neural network was used to map in a configuration space. The recognition rate was 97% for a single user, which exists in both the training and test dataset when there are 10 grasp types. The recognition rate dropped to 65% for unseen users that were not in the training dataset. There is no information on performance if unseen objects were tested. Aleotti and Caselli [54] performed grasp recognition using static grasp poses in virtual reality for six grasp types, with a recognition rate of 94% for two expert users without using real objects. In [55], different classification methods for recognizing six grasp types were compared. The best recognition rate of all the methods was 83.82% for seen users and 71.67% for both unseen users and objects.

However, the grasp type alone cannot form good grasps. Therefore, some researchers introduced additional information to grasp type. Work represented in [24-27] abstracted approach vector together with grasp type to map the location and pose of the robot hand from human demonstration. The approach vector can only be obtained by observing human demonstration. It provides a good initial pose to start with, but it does not indicate which surface or part of the object to grip.

Alternatively, grasp features can be abstracted based on the observation of the object [13]. One approach is to learn contact points on an object surface ([18-22, 57]), which has the problem of solving the inverse kinematics that satisfies the constraints imposed by contact points [23]. Pollard [58] presented an efficient data-driven algorithm for grasp synthesis by providing successful grasp examples. Contact regions are projected to the object surface by preserving similar quality measures from a demonstrated grasp. Saxena et al. [59] collected a database by labeling grasp points on 2-D images on the object. The grasp points are predicted based on examples using logistic regression method.

Li and Pollard [60] searched for candidate grasps by a shape-matching algorithm and evaluated the grasps by a task-oriented criterion. The algorithm compares the shape features between the given object model and that of the objects in the database to find the candidate grasp of relative contact position, and evaluates the grasp by form or force closure property.

Research in [61] adapted the contact points to similar objects by finding point-to-point correspondence between two meshes. The approach proposed in [62] learned grasp types from demonstration and generalized contact points by the contact warping algorithm, but since contact points were learned to generate the grasp, no power grasps were being tested on the robot because learning contact points is more suitable for precision grasp. Similar work was presented in [63], who combined hand shape with pregrasp contact locations together as human grasp features.

A functional affordance learning approach was developed in [64], in which the object is decomposed into graspable parts. The interaction between the user and the object is observed to determine the affordance part by the interaction region.

Transferring grasp features to the robot may generate unstable grasp candidates. Thus, work such as [52, 60, 61, 65], combined grasp quality analysis with human grasp strategies to find force-closure grasps close to human demonstration.

While some related research has shown success in generating successful grasps from examples, other researchers argued that the data-driven approach assumes the working conditions, such as the robotic hand and the object to be grasped, have to be similar to the human demonstration example. Transferring grasps to a distinct working condition from the example may lead to unexpected grasps. Without knowing the physical attributes of the world, the new grasp cannot guarantee a successful grasp.

Chapter 3

Task Modeling[1]

Object manipulation is central to many of the activities in daily life. To perform a task, the object to be manipulated would interact with the environment and suffer from outside disturbances along some directions. Often, during tool usage and other common examples of object manipulation, an appropriate grasp needs to be chosen to firmly hold the object with ease without being broken by the interactive disturbance, so that the object moves with the manipulator. Nevertheless, few take the task into account, as a consequence of the difficulties of task modeling and the computational effort of task-oriented grasp evaluation and grasp optimization.

Although tasks can be semantically described, such as pouring water using a bottle, pick up the bottle and open the bottle cap, etc., they are known to be difficult to be mathematically modeled. While most of the researchers placed their emphasis on grasp metric definition and computation, they assumed the task being known or approximated by experience. As introduced in Chapter 1, the task requirements are separated into two parts by Li and Sastry [11]: force requirement and motion requirement, both of which are used in our task modeling. A task can be obtained by methods categorized in two techniques: observation of a human who grasp the object and observation of the object that is grasped by a human. Both require necessary sensors to capture the force and motion data.

In this chapter, we propose two solutions in both categories in the following two sections. The first solution is to observe a human operator performing demonstrations in reality using a vision system. The second solution is that the demonstrator operates the object in a virtual reality environment, and then

[1] This chapter was published in IEEE Intl. Conference on Robotics and Automation, 2012, Lin, Y., Ren, S., Clevenger, M., and Sun, Y., "Learning Grasping Force from Demonstration"; and IEEE Intl. Conference on Robotics and Automation, 2011, Lin, Y., Sun, Y., " 5-D force control system for fingernail imaging calibration". Permission is included in Appendix A.

the forces applied to the object by the environment and the object motions are captured during the demonstration.

3.1. Task Modeling Based on Human Observation

In this section, we propose a preliminary work to learn object motion and interactive force from human demonstration using a vision system. In the literature, computer vision techniques have been introduced to track object motions [66]. The object motion and interactive force is achieved by applying force at the contact points. Since applying force is an important aspect of the interaction with the environment, and essential for many tasks, it is not surprising that humans exhibit complicated force strategies for different tasks. Measuring the grasping force along with the hand motion and understanding human force strategies are important for learning grasping and manipulation. However, unlike motion tracking systems that are usually not intrusive, most grasping force measurement approaches require force sensors being placed on the grasping points, either on the object or on the demonstrator's fingertips. In many neuroscience studies, specially designed objects have to be fabricated to incorporate force sensors and the points of grasping have to be defined beforehand, which pose big limitations for grasping demonstration.

We utilized the previous work, a force imaging approach [67], to estimate fingertip force from the images of the back of the fingertip. This approach does not encumber a subject and there is no need for sensor fabrication or embedded sensors on the objects so that everyday objects can be used for studies and demonstration. The existence of low-cost cameras and image processing methods readily performed on PCs makes the instrumentation of such an approach a relatively inexpensive means to be able to capture both motion and force data.

3.1.1. Fingertip Force Estimation System

The fingertip force is measured remotely by the color of the corresponding fingernail and its surrounding skin. The relation between the color and the force can be modeled with a generalized least-squares model [67]:

$$\hat{f}=(B^T\Sigma^{-1}B)^{-1}B^T\Sigma^{-1}(h-a) \quad (3.1)$$

where h is the vector with the color reading of all the pixels in the fingernail and surrounding skin. Parameter vectors a and B are learned linear parameters relating the color response to the force. The covariance matrix Σ is estimated from the data which represent the weights of all pixels contributing to the estimation of force [67].

3.1.2. Training of the Fingertip Imaging System

The least-squares model needs to be trained individually for every finger that the system wants to measure force on. An inexpensive automated calibration system is designed to apply calibration force trajectory on a human fingertip with high precision and take calibration images and force data simultaneously (Figure 2A). The calibration system is composed of a 5-DOF actuation device, a 6-axis force sensor, a video camera, a hand-rest stage, and a finger restraint.

The actuation device is an integration of two Novint Falcon devices linked by two universal joints and a rigid bar to provide 5-DOF motion and force, with feedback from an ATI Nano 17 force/torque sensor. The kinematic model of the actuation device is shown in Figure 2C. A force controller is designed with an inner position control to meet the calibration goal and requirement. The system was capable of controlling force with a settling time of less than 0.25 seconds, and tracking force trajectories with an interval of 0.3 seconds and step sizes of 0.1 N and 1 $N \cdot mm$. Root-mean-squared errors are 0.02-0.04 N for forces and 0.39 $N \cdot mm$ for torques. The design and implementation of the actuation device was described in details in [68].

A Point Grey Flea video camera is used to take training images with the force reading from the force sensor. The force sensor reading is sampled at 1 kHz and the video camera works at 30 frames-per-second (fps). One original image for training is shown in Figure 3A. Its fingertip image after segmentation and normalization is shown in Figure 3B.

A large wooden hand rest is used to support hand weight during calibration (Figure 2B). The wide base of the hand rest provides stability to the structure and allows for the addition of the unbalanced

weight of the aluminum insert and finger restraint. The finger restraint has an L-shaped base that conforms to the tree shape of the wooden hand rest, capable of restraining a finger on either hand. The constraint prevents unwanted finger movement that may cause disturbances and noise in the collected data.

To verify the calibration result, we used two sets of training data to train the GLS model in Section 3.1.1, and then verify it with the third data set. Figure 4 shows the verification result of the calibration. The root-mean-square (RMS) error is 0.3012 *N*; that is consistent with our previous finding [67].

To compare the result of the force measurement from our imaging approach to the embedded sensor approach, we placed two FlexiForce force sensors on one object for a pick-and-place task. FlexiForce is a thin film force sensor which measures one dimensional pressure. The other types of force sensors are able to measure higher dimensional and more accurate forces, but are much larger in size. Applying the trained GLS model, the force during a pick-and-place task is estimated and displayed in Figure 5. For comparison, the force measured with the FlexiForce thin film force sensor is displayed in Figure 6, showing a large amount of noise in the force measurement by FlexiForce sensors. According to Park [69], the repeatability of the FlexiForce senor ranges from 75% to 91% and it is worst for lowest forces.

Compared to regular force sensors, the imaging approach to measuring force is not intrusive, though accurate enough for the applications of force measurement and grasp studies.

3.1.3. Motion and Force LfD Station

Both position and force are measured with two Point Grey Flea video cameras with 16 mm lenses, attached on two parallel half circle rails as shown in Figure 7A. The number of the cameras and the selection of the lenses on those cameras are decided based on the tasks. For a pick-and-place task using two fingers in a confined space, two cameras with relatively narrow lenses are sufficient. More cameras would certainly provide better coverage for larger motions.

The two cameras are configured to capture the images of the fingernail and surrounding skin areas of the thumb and index finger along with the fingertip motion during a grasping-placing process. Software was developed to capture the fingertips of both index finger and thumb from both cameras simultaneously at 30 frames-per-second (fps). The fingertips are tracked and segmented from the background and then the calibrated force estimation model is used to estimate the force of both fingers during the tasks. The estimated force is then combined with the fingertip motion to provide the training data.

The recorded demonstrated motions and forces are then modeled with a Gaussian Mixture Model (GMM) that summarizes a probabilistic representation of multiple demonstrations [70]. Different from previous approaches, the proposed motion and force LfD system has extra three dimensions that represent a 3-dimension force vector. At any time point, a 6-dimension motion vector and a 3-dimension force vector are combined to represent an action state of a task.

The demonstrated motion and force can be encoded together by GMM. Given a set of data points of dimensionality D, one dimension is the time step, while the other dimensions are motion and force trajectories.

The dataset is defined by $\xi_j = \{\xi_{t,j}, \xi_{m,j}, \xi_{f,j}\}$, where $j = 1, 2, ...N$ is the number of trials, $\xi_{t,j}$ represents the time step, $\xi_{m,j} \in \Re^6$ is the position and orientation vector, $\xi_{f,j} \in \Re^6$ is the force vector. The dataset is modeled by a mixture of Gaussian distributions of K components. K is defined as 4 in the pick-and-place task, because we segmented the whole grasping process into 4 stages – grasp the object, lift the object, place the object back, and release the object. The GMM model is defined by the function:

$$p(\xi_j) = \sum_{k=1}^{K} p(k)\, p(\xi_j \mid k) \tag{3.2}$$

where $p(k)$ is the prior, and $p(\xi_j \mid k)$ is the conditional probability of the *j*th data given the *k*th Gaussian distribution. They are further defined as:

$$p(k) = \pi_k \tag{3.3}$$

$$p(\xi_j \mid k) = N(\xi_j; \mu_k, \Sigma_k)$$

$$= \frac{1}{\sqrt{(2\pi)^D \mid \Sigma_k \mid}} e^{-\frac{1}{2}((\xi_j - \mu_k)^T \Sigma_k^{-1} (\xi_j - \mu_k))} \tag{3.4}$$

for $k = 1$ to K. The parameters of prior π_k, mean μ_k, covariance matrix Σ_k of the Gaussian Mixture Model are estimated by an expectation-maximization (EM) algorithm [71], which maximize the likelihood of $p(\xi \mid \pi)$.

Several objects are used to verify the grasping force learning from demonstration framework. For example, Figure 8 shows the demonstration grasping force reading from the index finger of picking and placing a red pepper and its position along the vertical direction of the end-effector. Most attention is paid to the normal force and height because they are highly related to each other in the pick-and-place task. The dataset of force and motion during 3 trials of pick-and-place demonstration is modeled as in Section 3.1.1. The GMM model is shown in Figure 9. The number of components is selected to be 4 so that the components naturally represent the mental intentions of the human user - grasp, lift, place and release.

This method allows the fingers to directly contact the environment without obstructing the human natural haptic senses, but the disadvantage of this method is obvious – the result varies a lot from person to person. Besides, the calibration procedure is difficult. In our current setting, the motion range is also limited by the vision system. Therefore, we propose in the next section another task modeling approach based on observation of an object being manipulated in a virtual reality environment.

3.2. Task Modeling Based on Object Observation

Due to the limitations of modeling task information in reality, we provide another solution to model the task by observing the object being manipulated using a virtual reality system. In this category, the interactive wrenches between the object and the environment are captured, instead of capturing the grasping force. The space of all interactive wrenches given a task is called task wrench space (TWS) in

the literature. All interactive wrenches included in the TWS are required to be resisted in the task. Without knowing the specific task requirement, one can assume the TWS to be a ball, where wrenches to be resisted are of the same magnitude along all directions. However, the task ball does not help find the optimal grasp that is the best fit for a specific task. Instead of a wrench space ball used in a force-closure quality measure, Li and Sastry [11] used a six-dimensional wrench space ellipsoid to better approximate a task to facilitate the computation of the quality measure. The research in [17] predicted the contact disturbance and approximated the TWS as a task polytope. For example, to manipulate such tools as a pen, screwdriver, scoop, fork, toothbrush, etc., the contact disturbance is expected to be applied on the tip of those tools. Then, the empirical task-oriented disturbance wrench space is a friction cone applied to the tip.

Moreover, the task is considered only by the shape of the TWS without taking the wrench distribution into account, where the wrenches are assumed to be uniformly distributed in the space. However, this is rarely the case. Compare a writing task and the manipulation of a screwdriver, for example. Although both require the grasp to resist the disturbance force applied to the tip, they have different disturbance distributions, as illustrated in Figure 10. For the writing task, the main disturbance wrench is the force pointed to the upper-left direction and the torque generated along with the force. Hence, the GWS should be able to apply the opposite force to resist the disturbance, which is distributed primarily in the right area of the friction cone shown in the figure. The main disturbance wrench of the screwdriver is the normal force to the surface and the rotational torque around the principal axis of the screwdriver. In Figure 10, we only show the force subspace for comparison purpose. Also, the expected disturbance force of the screwdriver is larger than that of the pen. As a result, different distributions of wrenches in a TWS would prefer different grasps.

Therefore, we propose to characterize the TWS and represent the task wrench distribution in it with task wrench cluster (TWC). It is not trivial in capturing the task wrenches applied on physical objects, and collect enough wrenches to form a cluster that could represent the task wrench distribution

well. Thus, we implemented a virtual reality environment to provide a graphical user interface to the human demonstrator. A 6-D haptic device, Phantom Omni, allows users to control a virtual object in terms of 3-D position and 3-D orientation, and provides them with 3-D haptic feedback of the interaction force between the haptic interaction point (HIP) and the virtual environment. Although a single device cannot control a multi-link body or soft objects, the HIP can be a representation of a rigid body composing the majority of daily tools.

The virtual reality environment was developed based on Chai3D [72], an open source C++ library for computing haptic feedback, visualization, and interactive real-time simulation. It integrates Open Dynamic Engine (ODE) for collision detection and dynamics simulation and OpenGL library for graphical visualization. For each task, a user was asked to manipulate a virtual tool using the haptic device (see Figure 11 for example). The 3-D collision forces of the tool with the environment were captured in every iteration, with a sampling rate of 100 Hz, and 3-D torques were computed from the collision forces. All wrenches (forces and torques) collected during the task form a TWC for that particular task:

$$TWC = \{w(i) \in \Re^6 \mid w(i) = w_c(i) + w_n(i), i = 0,1,...,t\} \tag{3.5}$$

where $w(i)$ is a wrench at the ith iteration, $w_c(i)$ is the wrench generated by the contacts between the tool and the environment, and $w_n(i)$ is a non-contact wrench.

The non-contact wrench $w_n(i)$ is an offset wrench that includes forces not acting on the surface of the object, such as gravity and the inertial force. Gravity was considered as the force acting on the center of mass of the object. If the center of mass is set as the torque origin, the wrench compensated by the gravity is a wrench with zero torque. If no contacts occur during the manipulation, only non-contact wrenches are required to be balanced by the grasp, e.g., when lifting a book on an open palm, where the task wrench stabilizes the effect of gravity along a single direction. Note that the direction of the gravity disturbance relative to the object coordinate frame changes with the motion of the object, e.g.,

when rotating a book by hand, where the task wrench stabilizes the effect of gravity along multiple directions.

Similarly, the task motion sequence (TMS) of the object is described by a sequence of vectors $\{u, \dot{u}, \ddot{u}\} \in \Re^m$ representing the position and orientation, velocity and acceleration of the object in task coordinate.

$$TMS = \{u(i), \dot{u}(i), \ddot{u}(i) \mid i = 0,1,...,t\} \tag{3.6}$$

3.3. Study on Disturbance Distribution

In this section, the simulated disturbance distribution is verified and the distribution model is studied.

3.3.1. Verification of Simulated Disturbance Data

Since the proposed grasp planning relies heavily on the disturbance measure, it is necessary to validate how reliable and realistic the disturbance distribution in our simulation is. Thus, we evaluated the simulated data by comparing it with real measurements on a physical tool. To measure the real data, we designed a physical tool, as shown in Figure 12, which is incorporated with a 6-axis Nano17 force sensor connecting a handle and a stick. The object wrenches can be measured by the force sensor. This tool can mimic a long-shaped tool that has a handle, such as a screwdriver, a knife, a fork, a scoop and a marker, etc. The same manipulation tasks can be demonstrated both in simulation and using the physical tool. The captured disturbance data was compared in both environments.

We used this tool to execute some interactive tasks with a plane. The actions being tested included sliding and rotational motions, which produce supportive force, frictional force, and rotational torques on the object. The same model of the tool was used in simulation. The distributions from the measured wrenches by the sensor were compared with the wrench distributions obtained in simulation. Each task presented similar distributions. Figures 13 and 14 report disturbance force measured from two example manipulation tasks: a screwdriver manipulation and a cutting task using a knife. Figures (a-b) of both figures show the disturbance forces in simulation and on the physical tool. To compare the

distributions, quantile-quantile (Q-Q) plots were used to plot the relationship between the two sets of data: simulation and reality, as given in Figures 13(c)-(e) and 14(c)-(e) for the two example tasks.

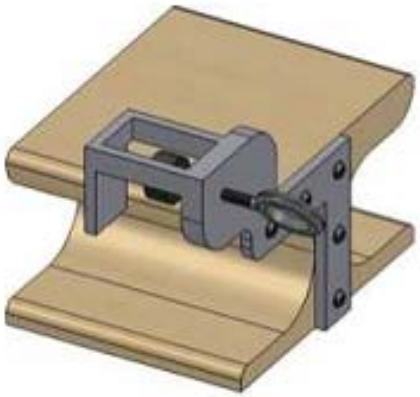
A Q-Q plot is an excellent graphical method to compare the shapes of distributions and provide a graphical view of how their distribution properties such as location, scale, and skewness are similar or different. In such a plot, points are formed from the quantiles of the data. If the resulting points roughly follow a straight line with a positive slope, the two distributions being compared are similar or linearly related. If the line is diagonal, the two distributions are the same. It can be observed that the points in this study lie roughly on a straight line except for a few outliers, indicating that the distributions of simulation are close to the real measurement, but with a longer tail (e.g., Figure 14(d)) or a shorter tail (e.g., Figure 13(e)). The position and slope of the line indicate the relative shift in location and scale of the samples, which is reasonable, because the samples are measured in different coordinate systems, and the sample scale in simulation can be adjusted by the parameters of the haptic generation model. Therefore, it was verified that the simulation could be used to characterize the wrench distribution of those tasks involving rotational and sliding interactions, though we were unable to verify the complete performance of the simulator for every possible interaction.

3.3.2. Study of the Distribution Model

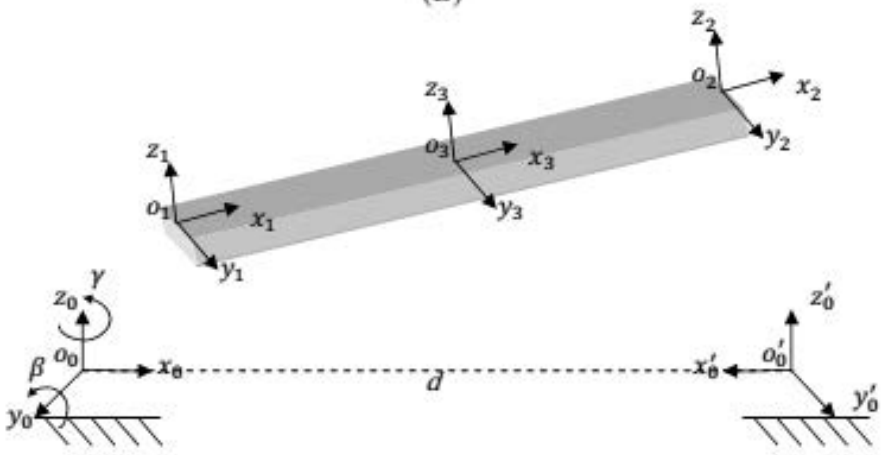
To study the distribution of the disturbance, we compared the distribution data with a standard uniform distribution and a normal distribution by a Q-Q plot. Figure 15 shows Q-Q plots of sample data against a standard uniform distribution and a normal distribution for the two aforementioned example tasks. According to the Q-Q plots versus a uniform distribution shown in the left column of the figure, both task disturbance distributions are distinct from a uniform distribution, so the task disturbance is not evenly distributed. The distributions of task disturbance are also not close to a normal distribution (Figure 15(b) and (d)), since none of them lie roughly on a line. Therefore, the distribution model of a task disturbance cannot be characterized as a uniform distribution or a normal distribution.

Because the probability distribution model of disturbance is unknown and the shapes of the Q-Q plot change with each task, we built a non-parametric statistical distribution of the disturbance from the TWC measured by demonstration for each task. Then, to reduce the computational complexity, a smaller set of data points could be randomly down-sampled based on the non-parametric statistical distribution.

(A)

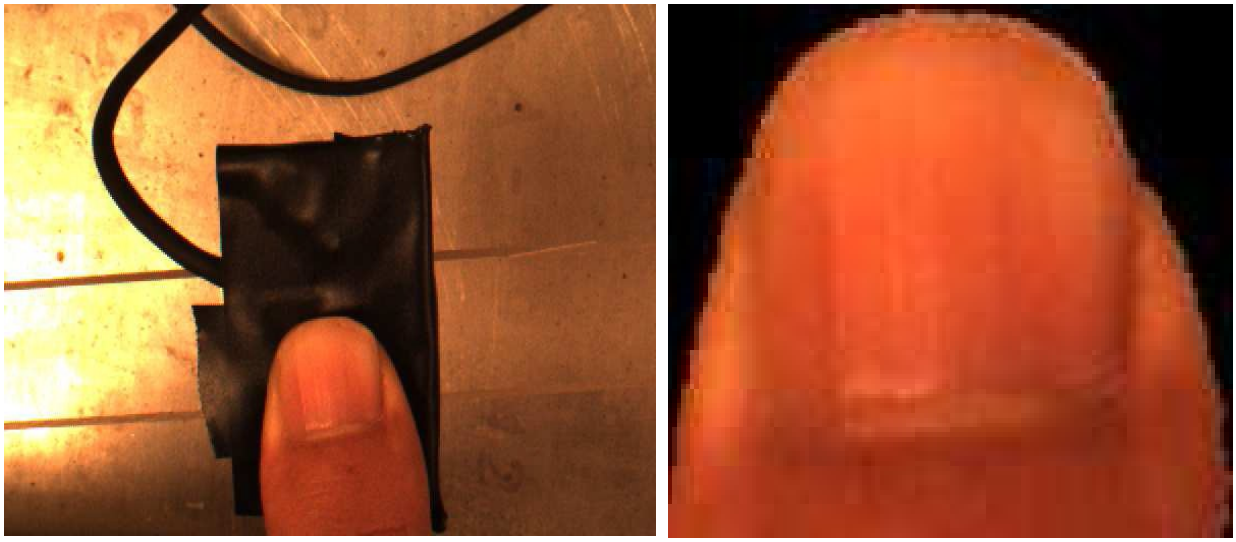


(B)



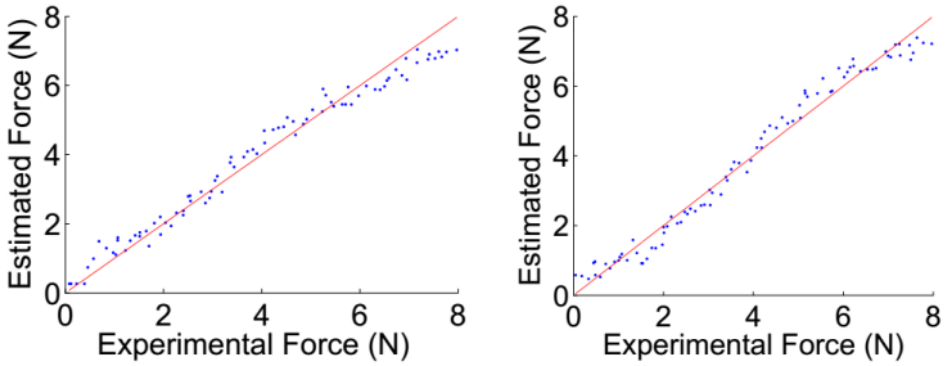
(C)

Figure 2. The Fingertip imaging system. (A) The 5-DOF calibration stage based on two Novint Falcon Haptics Controllers; (B) the hand rest with finger restraint; (C) the kinematic model of the calibration station.



(A) (B)

Figure 3. The fingertip images taken before and after segmentation and normalization. (A) Original image taken with the camera at the calibration station; (B) The fingertip image after segmentation and normalization.



(A) (B)

Figure 4. The estimation result of the GLS model on a new verification data set. (A) The verification result for an index finger; (B) the verification result for a thumb.

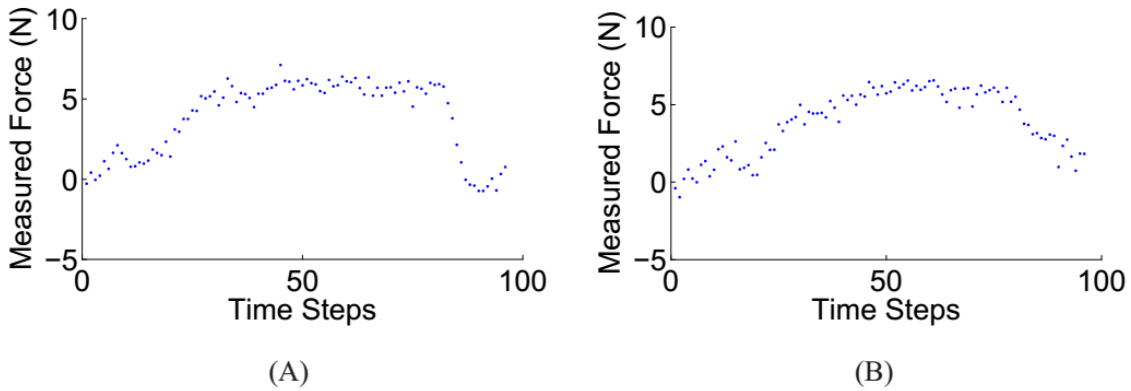


Figure 5. Estimated force during a pick-and-place process from the images of the fingernail and surrounding skin. (A) The estimated force on the index finger; (B) The estimated force on the thumb.

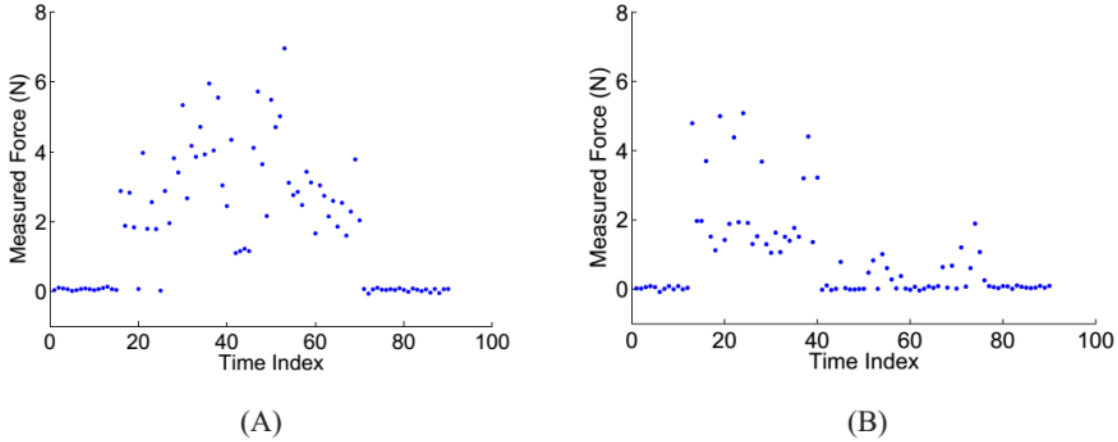
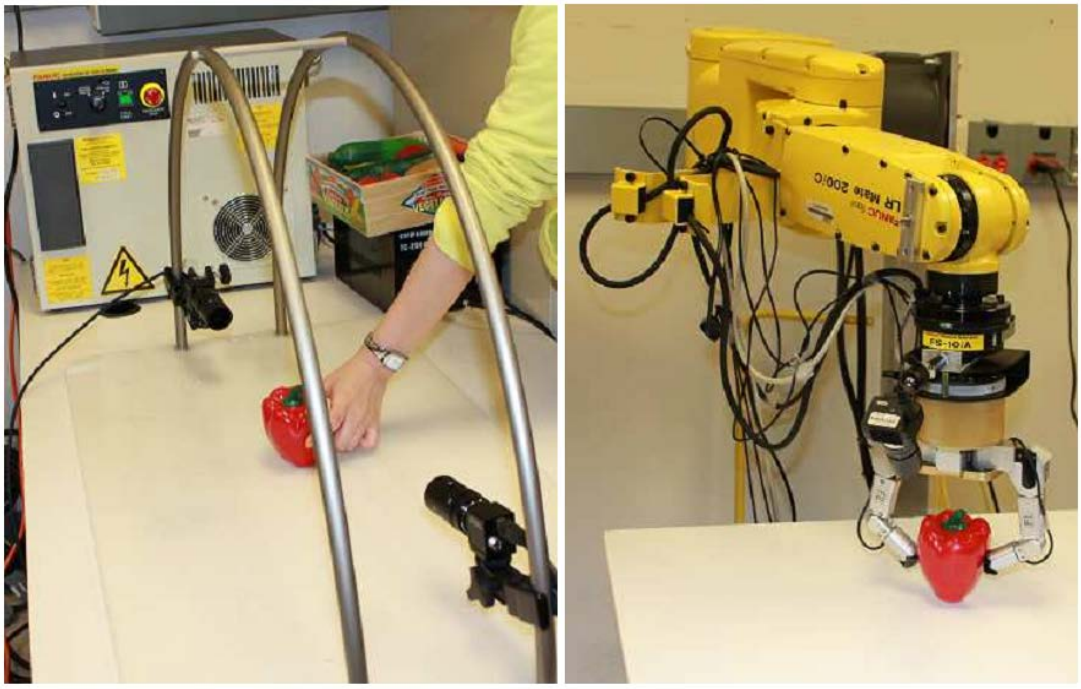


Figure 6. The measured force by the FlexiForce sensor. (A) The measured force on the index finger; (B) The measured force on the thumb.



(A) (B)

Figure 7. The experimental platform.(A) Grasping force demonstration platform; (B) The robot grasping and manipulation station.

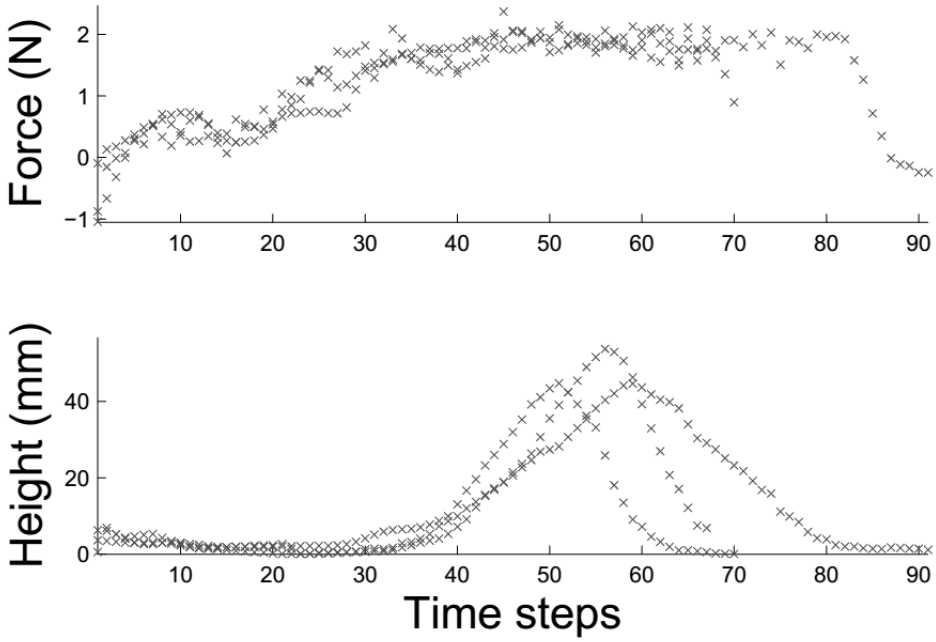


Figure 8. The demonstrated dataset.

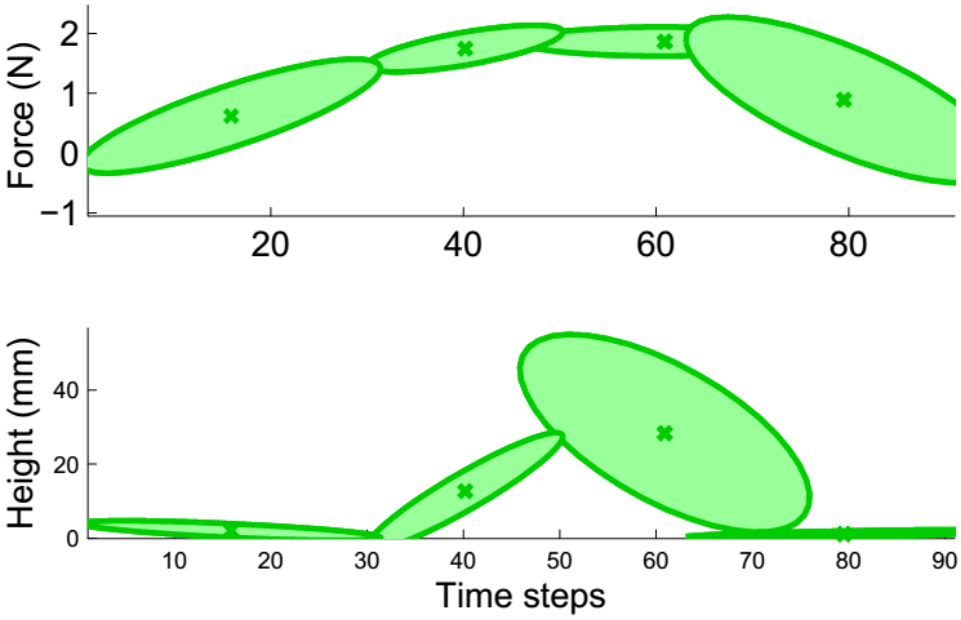


Figure 9. The GMM model results.

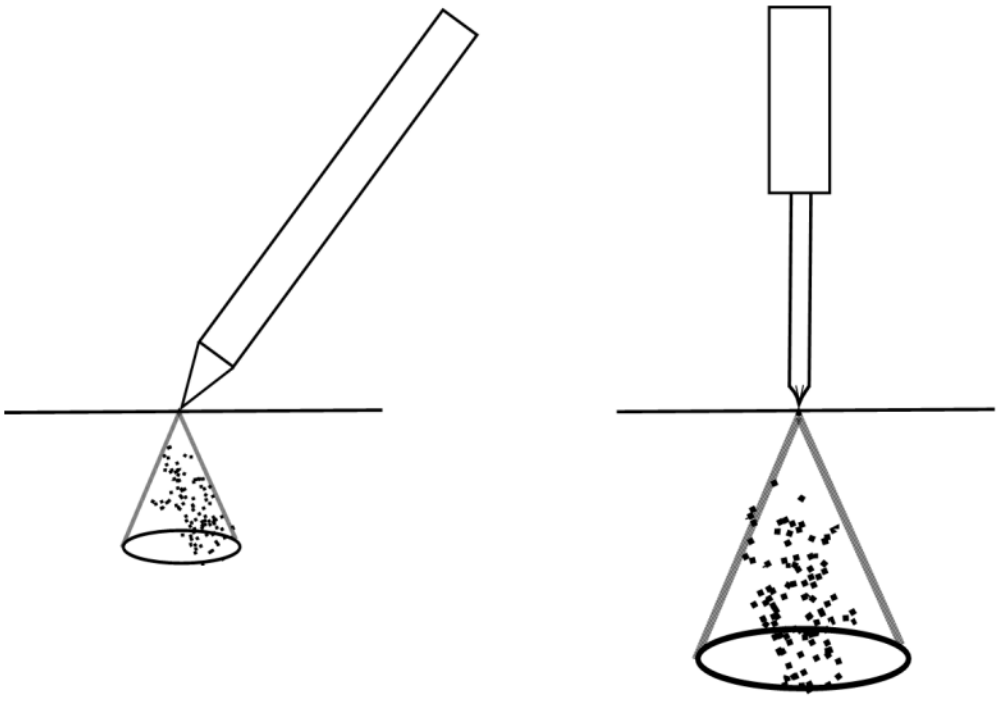


Figure 10. Disturbance distribution of two tasks. Left: a writing task with a pen; right: a screwing task with a screwdriver.

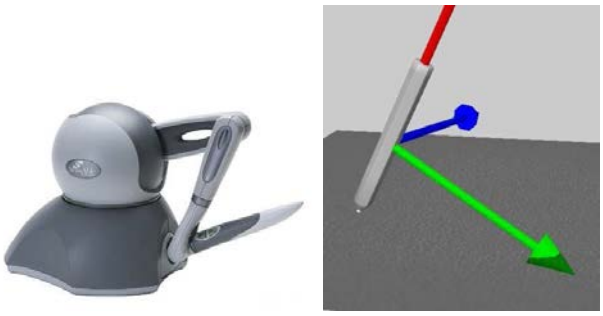


Figure 11. A user interface for demonstration. Left: the haptic device, Phantom Omni; right: virtual environment.

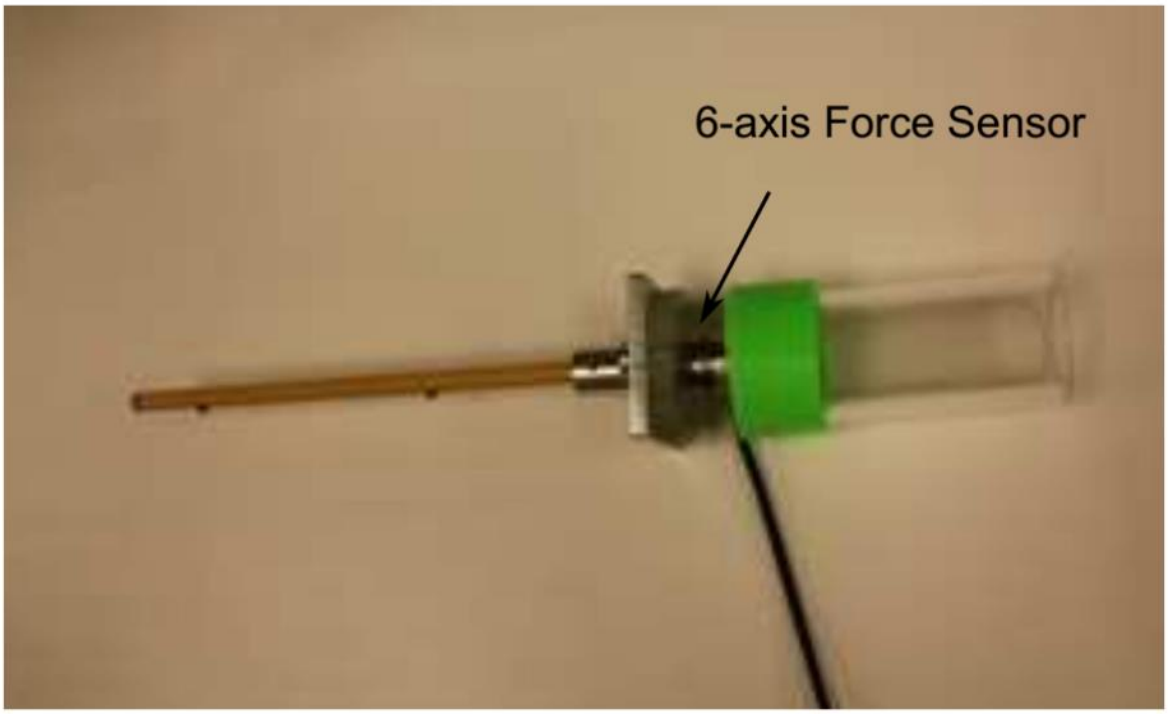


Figure 12. A tool designed to verify the disturbance obtained in simulation. This tool has a force sensor, a handle, and a stick.

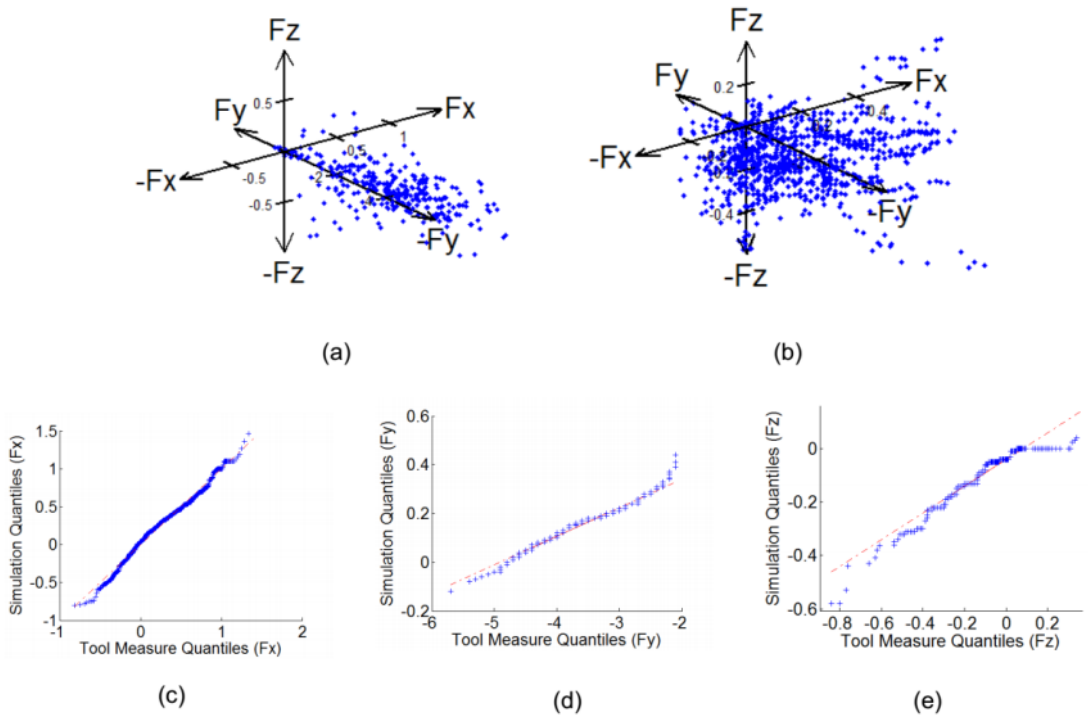


Figure 13. Example measurement of disturbance force when manipulating a screwdriver. Figure (a): disturbance force measured in simulation; (b): disturbance force measured from the tool sensor; (c)-(e): Q-Q plot of distribution of Fx, Fy and Fz in simulation against the real tool measure.

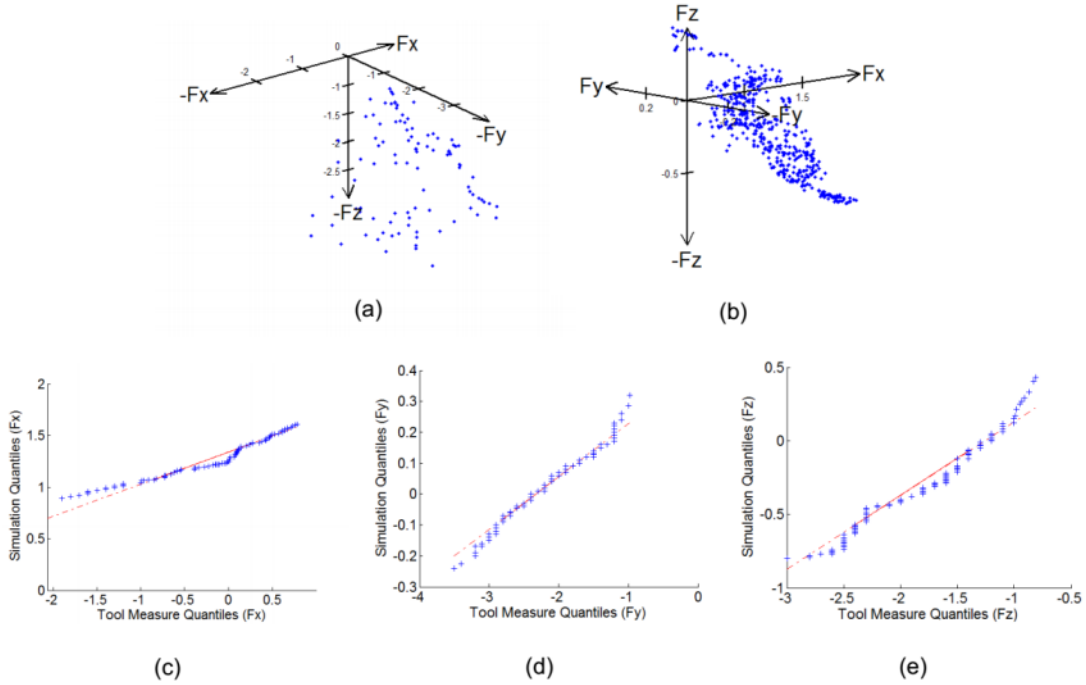


Figure 14. Example measurement of disturbance force when manipulating a cutting task using a knife. Figure (a): disturbance force measured in simulation; (b): disturbance force measured from the tool sensor; (c)-(e): Q-Q-plot of distribution of Fx, Fy and Fz in simulation against the tool measure.

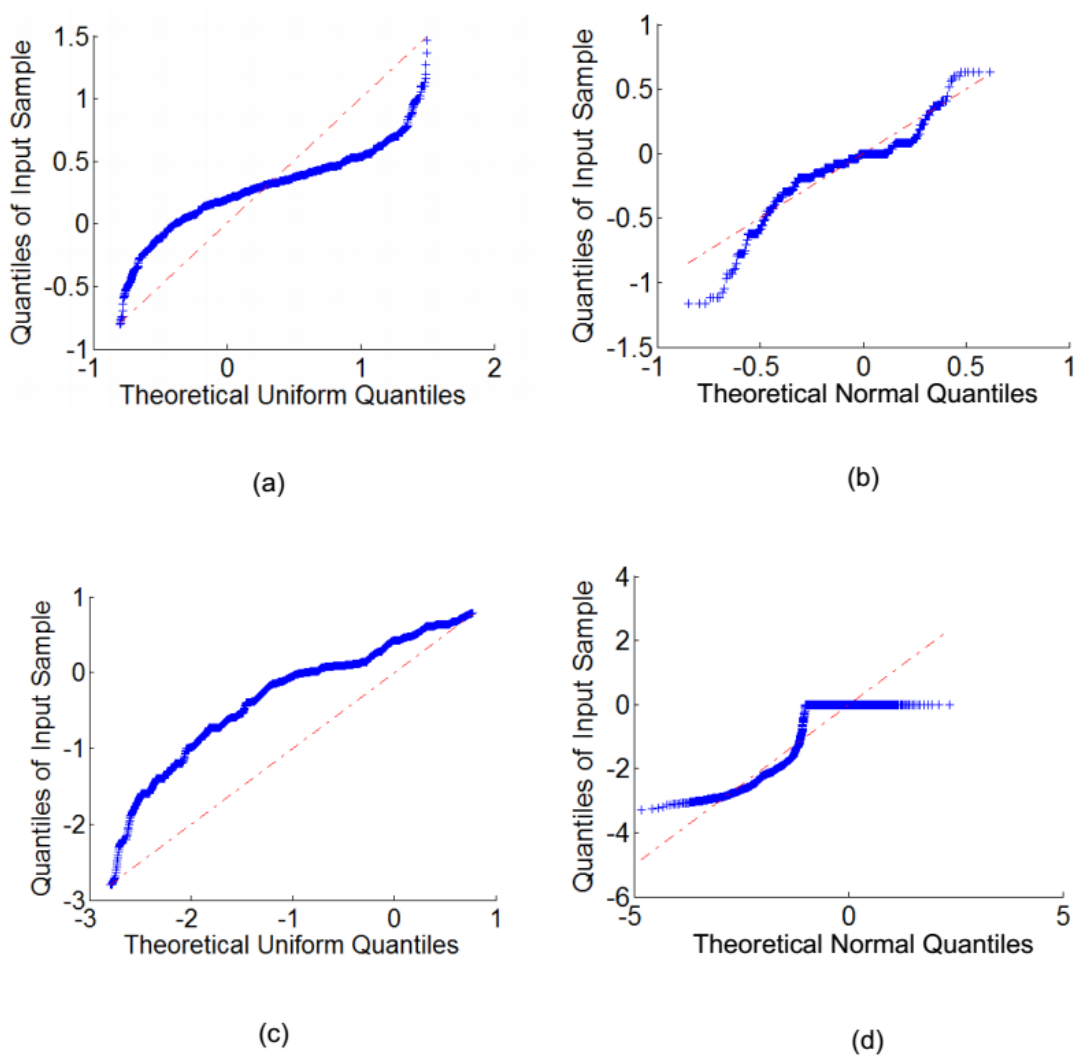


Figure 15. Q-Q plots to compare sample disturbance data with two standard distribution models. Top row: Q-Q plots for a screwdriver task; bottom row: Q-Q plots for a knife cutting task; left column: Q-Q plots of sample data versus uniform distribution; right column: Q-Q plots of sample data versus normal distribution.

Chapter 4

Grasp Quality Measures to Fulfill Force Requirement[2]

Grasp planning can be treated as an optimization problem, which searches the maximum value of the high-dimensional quality function Q (e.g. force-closure property). The quality measure is determined by contact points of the hand on the object, and contact points are further determined by the hand posture as well as the relative wrist positions and orientations. Therefore, Q is a function of hand posture and position:

$$Q = f(p, w) \tag{4.1}$$

where $p \in \Re^D$ is the hand posture and $p \in R^6$ is the position and orientation vector of the wrist. The dimensionality D depends on the degrees of freedom of the robotic hand. p is the vector that contains the robotic hand's joint angles.

4.1. Grasp Analysis

Considering a multi-fingered robotic hand grasping an object, a grasp comprises n contact points. The contact can be modeled by three types: point contact without friction, hard finger, and soft finger [38]. Here, we consider only the hard finger model, i.e., point contact with friction (PCWF), which is widely used in grasp analysis because it has no difficulty in generalizing the model. Using Coulomb's friction model, the i^{th} contact force $f_i \in \Re 3$ has the following constraint:

$$F_i = \{ f_i \mid \frac{1}{\mu_i^2}(f_{it_1}^2 + f_{it_2}^2) < f_{i\perp}^2 \} \tag{4.2}$$

[2] This chapter was published in Proc. IEEE Intl. Conf. Robotics and Automation, pp. 1374-1379, 2011. Y. Lin and Y. Sun, "5-D Force Control System for Fingernail Imaging Calibration," Permission is included in Appendix A.

where f_{it_1} and f_{it_2} are two perpendicular tangential force components, $f_{i\perp}$ is the normal force component, and μ_i is the coefficient of friction. With a given friction coefficient, the maximum tangential force is determined by the normal force. Thus, each contact force f_i is constrained to a friction cone. The friction has a vertex at the contact point, and the axis is along the contact normal, with an opening angle of $2\tan^{-1}\mu$.

For the convenience of computation, the circular friction cone is usually linearized and approximated with an m-sided pyramid. Then, any contact force $\mathbf{f_i}$ that is within the constraint of friction cone can be represented as a convex combination of the m force vectors on the boundary of the cone.

$$f_i \approx \sum_{j=1}^{m} \alpha_j f_{ij} \tag{4.3}$$

where coefficient $\alpha_j \geq 0$, and $\Sigma_{j=1}^{m} \alpha_j \leq 1$ If the boundary force vector f_{ij} is assumed to be of a unit magnitude along the normal component, then the normal magnitude of the contact force f_i is bounded to be 1, i.e., $\| f_{i\perp} \| \leq 1$. Each local contact force will generate a force and torque in the object coordinate, grouped as an object wrench $w_i \in \Re 6$. Then, the net resultant wrench can be transformed from the local contact forces through the grasp matrix:

$$w = Gf \tag{4.4}$$

where $f = [f_1^T, ..., f_n^T]^T$ is a vector concatenating all contact forces together, and $G = [G_1, ..., G_n] \in \Re^{6\times 3n}$ is the grasp matrix. Grasp wrench space (GWS) is defined as the set of all possible resultant wrenches w that can be generated by all feasible contact forces lying within the friction cone. Figure 16 shows an example of the friction cone at each contact generated by a grasp in the left figure, and the set of object net forces visualized in the right figure. GWS is the convex hull of all the object wrenches. The magnitude of object wrenches scales linearly with the contact normal forces that are limited by the joint

capabilities. For comparison, the upper bound is typically set to be 1. Ferrari and Canny [38] defined the unit GWS by bounding unitary L_1 or L_∞ norm of the normal contact force. Since the contact force is under the frictional constraint, constraint on the normal component also imposes an upper-bound on the contact force. Here, by imposing a limit on the L1 norm (the maximum normal contact force) to be 1, the set of all possible resultant object wrenches produced by the grasp is defined as unit GWS (UGWS):

$$UGWS = \{Gf \mid f_i \in F_i, \sum_{i=1}^{n} \| f_{i\perp} \| \leq 1\} \tag{4.5}$$

An important grasp property is force-closure. A grasp is force-closure if for any wrench w, there exists a solution $f \in F$. In other words, a force-closure grasp is able to equilibrate external force and torque in any direction without violating the friction constraints. It is equivalent to the condition that there exist strict internal forces f_{in}, such that $Gf_{in} = 0$. Thus, if the origin of the wrench space is included in the interior of GWS, then the grasp is force-closure. Similar to the GWS, a task can also be described as the space of disturbance wrenches from the environment that the object must resist. Ferrari and Canny [38] quantified the force-closure property by the magnitude of disturbance wrench that can be compensated by the grasp in the worst case. If no task-oriented information is provided to form a subset of the whole space of wrenches, a typical TWS is a 6D ball T_{ball} centered at the wrench space origin, where external disturbance is uniformly weighted (left in Figure 17). The grasp quality is the reciprocal of the scale to enlarge the UGWS so that it contains the whole TWS:

$$Q_m(G) = \frac{1}{k_m} \tag{4.6}$$

$$k_m(G) = min(k) \mid T_{ball} \subset k \cdot UGWS \tag{4.7}$$

where $k_m(G)$ is the minimum sum magnitude of normal contact force in order to resist all task wrenches. The larger k_m is, the greater is the effort needed for a grasp to balance the task wrench along the weakest direction. Grasp planning is to find the maximum $Q(G)$, the reciprocal of $k_m(G)$.

Instead of using a uniform ball, the quality measure in Eq.4.6 can also be used for different task requirements, such as a TWS approximated by a task ellipsoid or a polytope.

4.2. Task Wrench Coverage Measure

The quality measure k_m in Eq. 4.6 measures the minimum scale of UGWS in order to enclose the entire task wrench space, which quantifies an absolute constraint in the worst case where the robot should not drop the object. This grasp metric is reasonable when considering only the geometry of the task wrench space, but it does not take into account the distribution of wrench samples in the space. Consider the scenario of two different GWS for the same TWC in a TWS shown in Figure 18(a) and 18(b), respectively. The TWS have wrench samples distributed mainly in the left area and some scattered in the right area. The two UGWS have the same shape but differ in their locations. Grasp 1 has a higher ability to apply wrenches in the left area, while grasp 2 has a higher ability to apply wrenches in the right area. If measured by k_m, it makes no difference between the two grasps considering only the shape of TWS, as they require the same minimum effort in order to balance all task wrenches. However, with a relatively small value of scale $k < k_m$, grasp 1 captures the majority of wrench samples in TWC. It implies that, compared to grasp 2, grasp 1 requires less effort to resist the frequently-occurring wrenches, thereby resulting in a higher efficiency of power consumption.

In cases where a robotic hand is not capable of covering all the TWC without exceeding some force limit, it makes more sense to capture as much of the TWC as possible. Since the task wrench distribution is modeled from the measurement and the tasks are performed in an environment full of uncertainties, there are few large task wrench spikes other than the commonly-occurring wrenches, which are caused by rare events or sensor noise. It is unrealistic and unnecessary to compute a grasp to cover all the worst cases, since it is unlikely that the sparse large task wrench spikes will replicate in the same task.

4.2.1. The Proposed Grasp Quality Measure

Intuitively, the grasp quality can be defined as the ratio of TWC that can be covered by the GWS that is linearly scaling with UGWS by a factor of k (the sum of contact normal force, which is an indicator of the robotic fingers capability). We define $W(G)$ as the subset of all wrenches in TWC, which are within the scaled GWS for a given k :

$$W(G) = \{\forall w(i) \in TWC \mid w(i) \in GWS\} \tag{4.8}$$

The grasp quality can be represented as:

$$Q(G) = \frac{|W(G)|}{|TWC|} \tag{4.9}$$

where $|W(G)|$ is the cardinality of the set $W(G)$, i.e., the number of the task wrench samples covered by the GWS, and $|TWC|$ is the cardinality of the set TWC, i.e., the number of all wrenches in the TWC. Obviously, $0 \le Q(G) \le 1$ .The physical meaning of quality measure $Q(G)$ is, given a capability constraint k on the normal contact forces, the percentage of task wrenches that can be resisted by the grasp. With this new grasp quality measure, we will be able to find an optimal grasp G that maximizes $Q(G)$, so it is able to apply the required task wrenches as much as possible, without exceeding a certain amount of contact forces.

4.2.2. Selecting *k*

In some scenarios, the selection of k , the scale factor to determine the size of GWS from UGWS, will affect the selection of grasp, since as k increases, Q increases nonlinearly. Here, we illustrate the influence of k with two grasps in three different scenarios. Since the two grasps may have distinct UGWSs, the two UGWSs would require different scale factors, k_{m1} and k_{m2}, to cover the entire TWC. In scenario 1, shown in Figure 18(c), where $k_{m1} = k_{m2}$ and for all $k < k_{m1}, k_{m2}$, $Q(G1) > Q(G2)$, selecting grasp 1 is always better than or equal to selecting grasp 2. In scenario 2, shown in Figure 18(d), where $k_{m1} > k_{m2}$, and for all $k < k_{m2}$, $Q(G1) > Q(G2)$, selecting grasp 1 is always better than or equal to

selecting grasp 2. However, in scenario 3, shown in Figure 18(e), the two grasp quality plots intersect at some random $k_c < k_{m1}, k_{m2}$, so $Q(G1) > Q(G2)$ when $k < k_c$, and it is opposite when $k > k_c$. Then, selecting grasp 1 or grasp 2 depends on the selection of k. Therefore, it is important to choose a meaningful and reasonable k.

Since scale k stands for the amount of normal contact force the robotic hand is expected to apply, we suggested a scale k_0 by considering both the capability of the robotic hand and the task requirement. We provide an example of k_0 selection that considers both the robotic hand capability and the features of the TWC.

In our validation experiments, we also used this k_0 selection scheme as a consistent rule for comparisons. Let $a(i) = \| f(i) \|$, which is the magnitude of the force component $f(i)$ in a given TWC; then, $max(a(i))$ represents the maximum magnitude of all contact forces in TWC. k_0 is determined by the smaller value between $max(a(i))$ and the maximum force $f_{\max}$ that can be applied by the robotic hand, considering the capability of robot actuators, written as:

$$k_0 = min(max(a(i)), f_{max}) \tag{4.10}$$

for all $i = 1,...,T$, where T is the number of wrench samples in TWC. Because the shape of the GWS does not usually fit with the shape of TWC well, when k_0 is selected by $max(a(i))$, GWS does not always cover all the wrenches in TWC.

In this dissertation, we used a Barrett hand for the experiment in a real environment. The maximal finger force of the Barrett hand is $20N$, so we set $f_{max} = 20$ in order to bound k_0. k_0 can also be set to other empirical values, e.g., the amount of force that humans usually apply in a manipulation. k_0 can also simply be the hand capability constraint on the contact force, so a grasp will be generated to best allocate the capability of the robotic hand to resist most disturbances in the task. For fragile objects, k_0

can be a smaller value to impose a strict limit on normal contact force, so that the resulting grasp will not break the object but can still hold the object with small contact forces.

4.2.3. Computational Efficiency of Grasp Quality

To compute the task wrench coverage of a grasp, the convex hull needs to be computed from the contact points, and then every sample of the TWC must be checked to determine if they are inside the scaled GWS. The convex hull is computed by a quick hull algorithm using Qhull C++ library, where the average case complexity is considered to be $O(mnlog(mn))$ [74], where n is the number of contact points, and m is the number of vectors used to approximate the friction cone.

To check if a point is inside the scaled GWS, one can test if the query point lies in the inward area of each facet of the convex hull. Comparing the point with one facet of the convex hull takes constant time. Thus, comparing a point with all facets of the convex hull is the worst case, taking $O(K)$ time, where K is the number of facets of the convex hull. To check if all samples are inside the convex hull takes $O(KL)$, where L is the number of task sampling points from the distribution of the disturbance.

4.3. Experiment

4.3.1. Simulation Results

In simulation, we tested our approach for several tasks with different objects. The data collection of TWC and grasp planning were programmed with Chai3D and ODE. Non-expert subjects were asked to manipulate an object in the user interface via Phantom OMNI. The interactive wrenches between the object and the environment were captured during the demonstration with a sample rate of 100 Hz. The data set of the disturbance, compensated by object gravity, was recorded. Then, from the data set, a non-parametric statistical distribution of the disturbance was built. To reduce the computational complexity, a smaller set of data points was randomly sampled based on the non-parametric statistical distribution.

A Barrett hand model and a Shadow hand model were tested during the simulation for task-oriented grasp planning. Here, we set the friction coefficient μ to be 1. The friction cone is approximated by an eight-sided pyramid. For each hand configuration, contact points and contact normals can be obtained by ODE, and then GWS can be computed. Grasp quality $Q(G)$ was calculated based on the GWS and the distribution of disturbance. The grasp planning searches for the best grasp configuration that maximizes $Q(G)$.

Figure 19 to Figure 21 show three examples of object manipulation. In the first example, the user was asked to perform a writing motion with a pencil, where the pencil interacts with the environment at the tip. The chosen grasp should be excellent for balancing the pressure and friction on the tip. As shown in Figure 19(a)-(c), task wrenches are biased to the positive directions of Fy and Fz, other than the full space of the friction cone. The resulting grasp is, therefore, close to the tip. For the hand configuration shown in Figure 19(d), $Q(G1) = 0.1969$ at $k = 2.6$, meaning it covers 19.69% of task wrenches, which is much lower than that of Figure 19(e), where $Q(G2) = 0.8459$ at the same k, because G2 is better to apply force along the +Fy and +Fz directions than that in G1. The quality measures $Q(G1)$ and $Q(G2)$ changing with scale k for the two grasps are compared in Figure 19(f). The resulting grasp looks different from a human grasp for a writing task, since the proposed grasp quality measure considers only from the perspective of the force requirement, and the hand kinematics is also different from the hand of humans.

In the second experiment, grasps were compared for two tasks using a knife. The user was asked to perform a cutting motion along one direction (+x marked by red in Figure 20 and a butter-spreading motion using both sides of the blade. The disturbance distributions for the two tasks are reported in Figure 20(a)-(d). As shown for the cutting task in Figure 20(a), a grasp should be able to generate pressure along the -z direction and friction mainly along the +x direction to the blade. Torque generated along with the force is shown in Figure 20(b). For the butter-spreading task shown in Figure 20(c) and

(d), the task wrenches cover a partial area of two opposite friction cones, i.e., the grasp should be able to apply pressure along both +y and -y, and friction along +z. Figure 20(e)-(g) contains evaluations of three grasps for the two tasks ($Q1$ for cutting task and $Q2$ butter spreading task). For the cutting task, the scale k was set to be 8.04 and larger than $k = 3.25$ for the butter-spreading task. It can be seen that for the cutting task, the hand configuration in Figure 20(e) is better to apply force in -Fz, along with -Ty. The hand configuration in Figure 20(g) has the worst quality measure of the three due to its deficient ability to apply force along z directions, whereas for the butter-spreading task, the hand configuration in Figure 20(g) is the best and in Figure 20(e) is the worst.

In the third task, the user was asked to strike a plane with a hammer, and the grasp planning was performed to compare results of the Barrett hand model and the Shadow hand model. It can be imagined that the chosen grasp should be excellent for balancing large pressure on the head of the hammer. As shown in Figure 21(a)-(b), the distribution covers almost the whole space of the friction cone, whose axis is along +z direction, and the pressure between the hammer and the environment along +z direction is as large as $20N$. The designated grasp type during grasp planning is a power grasp in order to perform powerful manipulation; the scale k of GWS is set to be 20 for the computation of quality measure. Figure 21 shows the searching results for the Barrett hand model (Figure 21(c)-(g)) and for the Shadow hand model (Figure 21(h)-(k)). It can be seen that the grasp that is closer to the head is better for counterbalancing the forces that occur at the head. Note that the result of a hammer grasp is different from the intuitive grasping style of humans, who prefer to hold the handle on the other side away from the head, because humans desire to reach a large swing motion with a relatively small arm motion, as well as to generate a large impact force. The grasp optimization considers only the ability to apply force by the fingers other than the arm and wrist motions. It can be observed from the figure that similar results were obtained for the two hand models, because task distributions are independent of hand mechanical structures.

As concluded from the experiments, the resulting grasp with a higher grasp quality criterion tends to be more efficient to apply frequently-occurring force, using the same magnitude of resultant force as the low-quality grasp, thus improving the efficiency of power consumption.

4.3.2. Experimental Results on a Real Platform

Experiments were also performed with a real robot system composed of a Barrett robotic hand and a 6-DOF FANUC LR Mate 200iC robotic arm. The Barrett hand has a 4-DOF with a three-fingered programmable grasper. The system is shown in Figure 22.

In the experiment, objects and manipulation tasks were carefully selected to evaluate the proposed approach. First, because we focused on the success rate of the resulting grasps to execute manipulation tasks rather than simple pick-up tasks, we did not consider tiny objects, such as a pen and a scoop, due to the physical limitation of the Barrett hand to grip tiny objects precisely. Also, only representative tasks were considered to avoid repetitive tasks. As a result, three representative manipulation tasks were selected to evaluate the proposed approach, including:

- Task 1: move a computer mouse on a table.
- Task 2: plug a power adapter into a power strip.
- Task 3: screw a light bulb into a socket.

Task 1 represents a sliding interaction with the environment. Similar tasks include using a dry-erase eraser, moving a broom or a vacuum cleaner on the floor, painting, etc. Task 2 represents a peg-in-hole motion. Similar tasks include inserting a key or a flash drive, etc. Task 3 represents a screwing motion. Similar tasks include screwing a screwdriver, jar lid, knob, key, switch, etc.

Each task was executed in 10 trials. The target object was placed at the same known location and orientation in the robot's workspace for each trial. Before each execution, the robot arm was reset to an initial position, and the robotic hand was kept open. The execution procedure was divided into four steps, as illustrated in Figure 22. The first step was to approach the target object and reach the wrist position and orientation relative to the object, which resulted from the algorithm. To avoid potential collision, the

robot first moved to 200 mm higher than the target pose, and then went straight down until reaching the target pose. Then, the robotic hand was commanded to close its fingers on the object and lift the object up from the table. These first two steps were performed autonomously. The next manipulation step was executed either autonomously or guided by humans, depending on the complexity of the manipulation. The first task, i.e., moving a mouse around on a table, was relatively simple, so it was executed in a predefined routine, in which the mouse was moved along a squared path on the table. The other two tasks, however, required a complicated sensing and manipulating technique to accomplish the task, which is beyond the focus of this dissertation. Therefore, we introduced human participation in completing the remaining task by teleoperating the robot using a haptic device, Phantom Omni. After the manipulation step was accomplished, the robot hand was then commanded to open the fingers and move up away from the object.

The Omni device was chosen due to its compact design and the intuitive positional abilities that we felt would be a good choice for the teleoperation of the robotic arm. The FANUC arm and Barrett hand were connected to the same PC with the Phantom Omni. The manipulator was teleoperated at a position-based and bilateral mode, for which force feedback was provided to the user. The positions and gimbal angles of the OMNI stylus were continuously transmitted to the PC server in real time. The position and orientation of the OMNI stylus were transformed to the corresponding position and orientation of the robot end-effector in its feasible workspace. The robot arm and hand incorporate their own motion controllers. The position commands were streamed from the PC server to the robot controller, so the manipulator was able to follow the OMNI motion in real time. For safety, the speed was constrained up to a feed rate of 30% of the maximum speed. The force sensed by the force sensor embedded in the robot wrist was feedback to the OMNI, so the user was able to feel the contact force when the manipulator was in contact with the environment.

To evaluate the proposed algorithm on a physical robotic platform, we compared the success rate with that of the widely used non-task-oriented planning method that optimizes the epsilon quality ε [11].

The epsilon quality ε measures the radius of the largest six-dimensional ball centered at the origin and enclosed with the convex hull of the UGWS. The epsilon quality refers to the magnitude of the disturbance wrenches that can be compensated by the grasp in the worst case. The larger the epsilon quality is, the more stable the grasp is in terms of resisting the worst-case disturbance. We did not compare it to the quality measure km discussed above, because it is not trivial to find such an absolute scale of km to fit the entire TWS into GWS without simplifying the geometry of TWS. We did not compare the results with other task-oriented methods either, because to the best of our knowledge, none of the related research on task-oriented grasp planning has reported any success rate on the execution of manipulation tasks in a physical system. Most work on real robotic platforms only tested pick-up tasks.

The grasps were being searched and evaluated by the epsilon quality measure and our proposed quality measure. The resulting hand configurations from both the proposed grasp quality measure and the non-task-oriented grasp quality measure are shown in Figure 23. The first two columns show the optimal grasps resulting from our approach and the epsilon quality measure, with both of the two quality metrics Q and ε marked in each corresponding figure. Column 3 and Column 4 compare the coverage of the TWC by the two grasps shown in the first two columns. In the figures, the wrench samples in the GWS are marked by blue points, while the wrench samples outside the GWS are marked by red circles. Again, the scale k of the marked quality metric Q was chosen by Eq. 4.10. The fifth column shows the proposed task-disturbance-based grasp quality Q as a function of scale k for the two grasps. Since grasps in column 1 are those that maximize our proposed Q and grasps in column 2 are those that maximize ε, $Q(G1)$ in column 1 is greater than $Q(G2)$ in column 2, while $\varepsilon(G1)$ in column 1 is less than column 2.

The execution on a real robotic platform of both the proposed algorithm and the non-task-oriented planning method for all of the three tasks were compared in Figure 24, and the success rate of the resulting grasp from both algorithms was compared in Table 5. If the object was sliding out of the robotic hand during the task execution because of outside disturbance from collision, it was counted as a failure.

Otherwise, if the robot successfully completed the task without dropping the object, it was counted as a success. It can be observed that, overall, hand configurations resulting from the proposed algorithm have a higher success rate (average of 70%) compared to that of the non-task-oriented planning algorithm (average of 43.3%) in executing the manipulation tasks.

The results imply that the proposed quality metric is more consistent with the success rate of the manipulation tasks than the non-task-oriented algorithm.

Task 1 required the robot to slide the mouse on a plane while keeping in touch with the plane. The disturbance was distributed mainly on the boundary of the friction cone (Row 1, Column 3 in Figure 23), so the grasp was to counteract the friction sliding on the plane and the support force from the plane. Both resulting hand configurations grasped the mouse on the side face (Row 1, Column 1 and 2 in Figure 23), but in the latter hand configuration, the fingers were closer to the bottom edge. It was observed in the experiment that in the latter hand configuration, the mouse was easier to slide up from the fingers during the execution, because the side faces were inclined inward. In Task 2, both the success rate and the quality Q as a function of scale k are close to each other, although they gripped the object on different faces. The success rate of both approaches appeared to be similar as well. In task 3, in the latter hand configuration resulting from the non-task-oriented approach, the other two fingers were closer to the base than the thumb. When the robot was trying to screw the bulb into a socket, it was fairly easy for the bulb to be dropped by the robot. In task 3, the success rate of our approach was much higher than the non-task-oriented approach, demonstrating its higher capability to resist the task disturbance.

Table 1 Comparison of success rate between proposed approaches using task disturbance with non-task-oriented approach.

Task	Success Rate of Task Disturbance Based Grasp Planning	Success Rate of non-task oriented Grasp Planning
Task 1	60%	40%
Task 2	80%	70%
Task 3	70%	20%
Overall	70%	43.3%

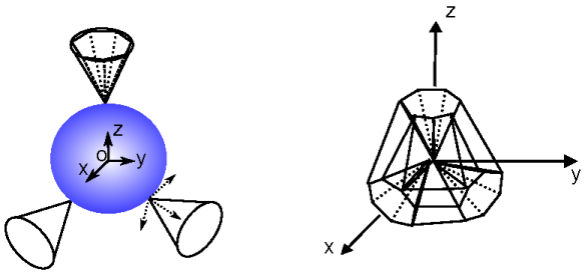


Figure 16. An example of a grasp. Left: grasp generates three contacts, and contact force is constrained in a friction cone. Right: set of object net force associating with the grasp.

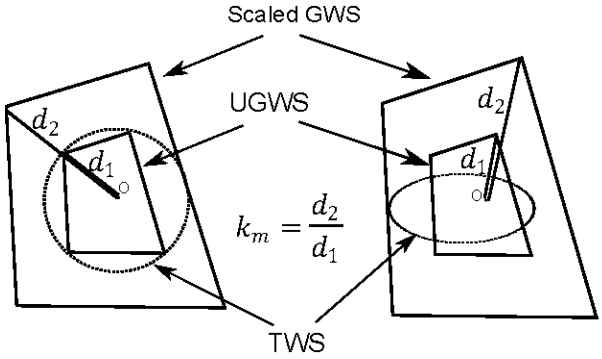


Figure 17. Grasp quality measures for (left) task ball represented by the dashed circle, and (right) task ellipsoid represented by the dashed ellipse. km is the linear scale factor to enlarge UGWS so that it encloses the entire TWS.

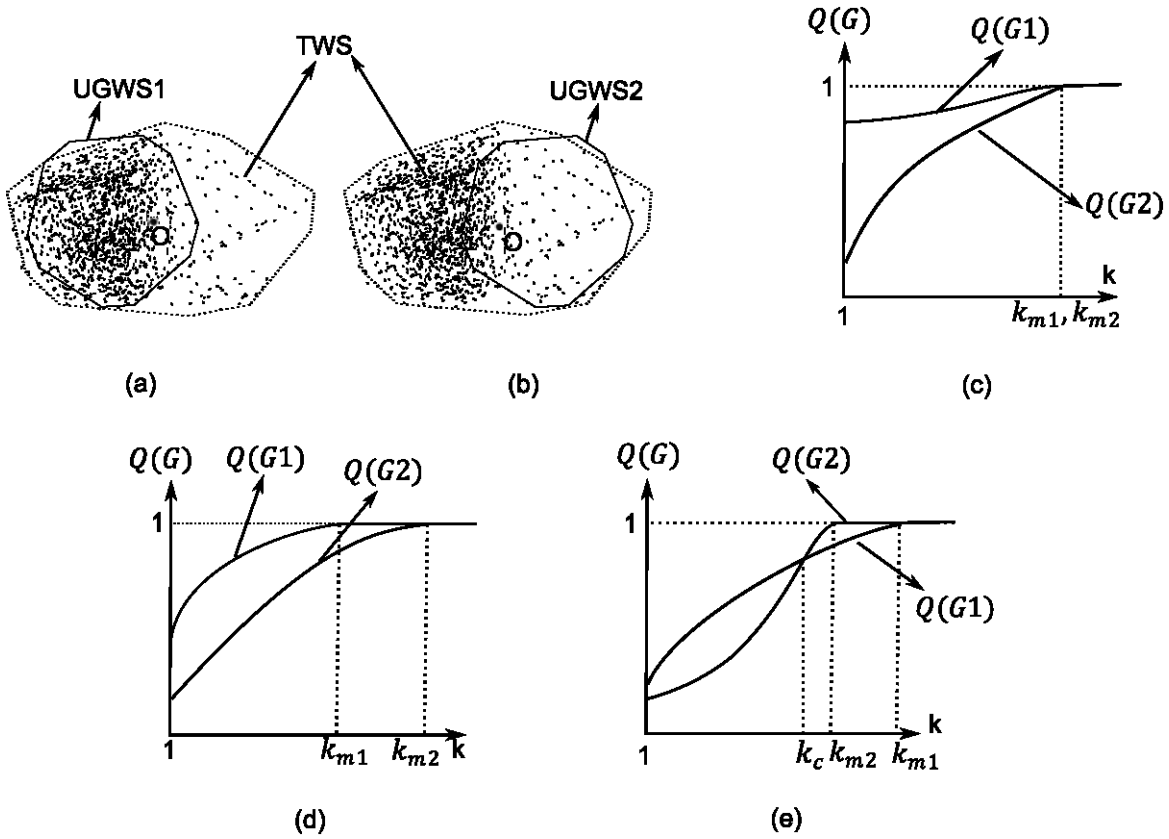


Figure 18. Comparison of quality measure Q in different scenarios. (a), (b): two GWS, given the same TWS; (c-e): compare the plot of Q(G1) and Q(G2) as a function of scale k in three cases. (c) Case 1: the quality plots of the two grasp intersect at km1 = km2, so for all k < km, Q(G1) > Q(G2). Thus, if measured by Q(G), grasp 1 is better than grasp 2, but they have the same quality if measured by the existing quality metric km. (d) Case 2: the two grasp plots intersect at km2, and km2 > km1. For all k < km2, Q(G1) > Q(G2). Thus, the result of our proposed quality measure agrees with the existing quality measure km. (e) Case 3: the two grasp plots intersect at some k < km, then the comparison of Q(G) between the two grasps differs upon the choice of k.

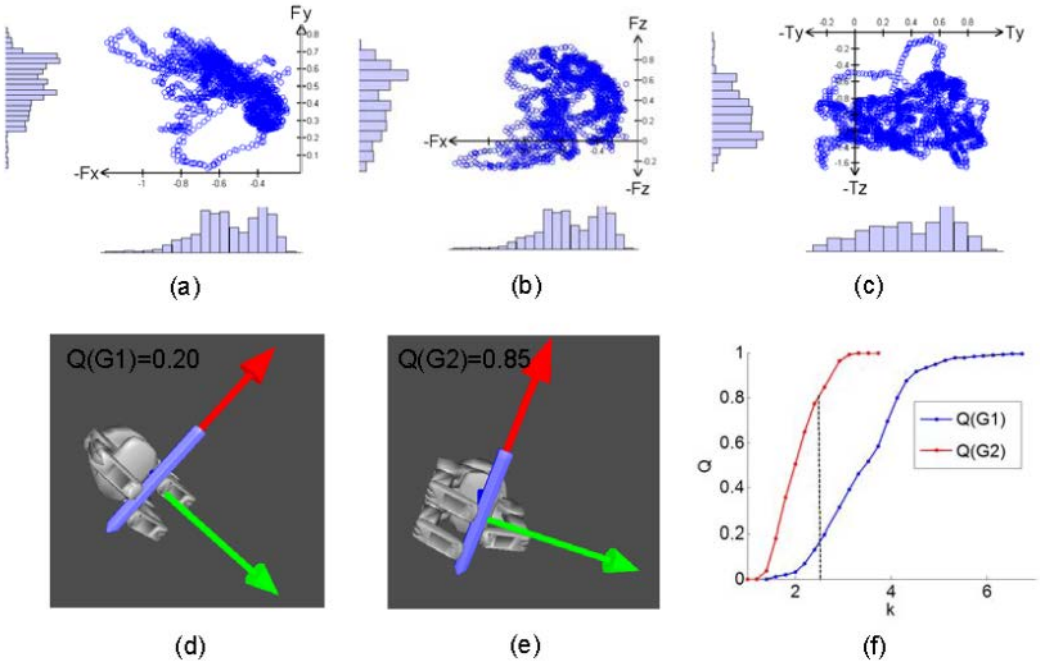


Figure 19. Planning results for a writing task with a pencil. The center of mass is set to be the origin of the coordinate frame, where axes x, y, and z are indicated by red, green and blue (shown in Figure (d) and (e)). (a)-(c): distribution of task wrench projected to Fx-Fy, Fx-Fz, Ty-Tz subspace, respectively, where the task wrench is distributed mainly along the Fx, Fy and Fz directions; torque Tz is small so it is not reported here. (d)-(e): two different hand configurations; (f): grasp quality Q versus scale k for the two hand configurations shown in (d) and (e).

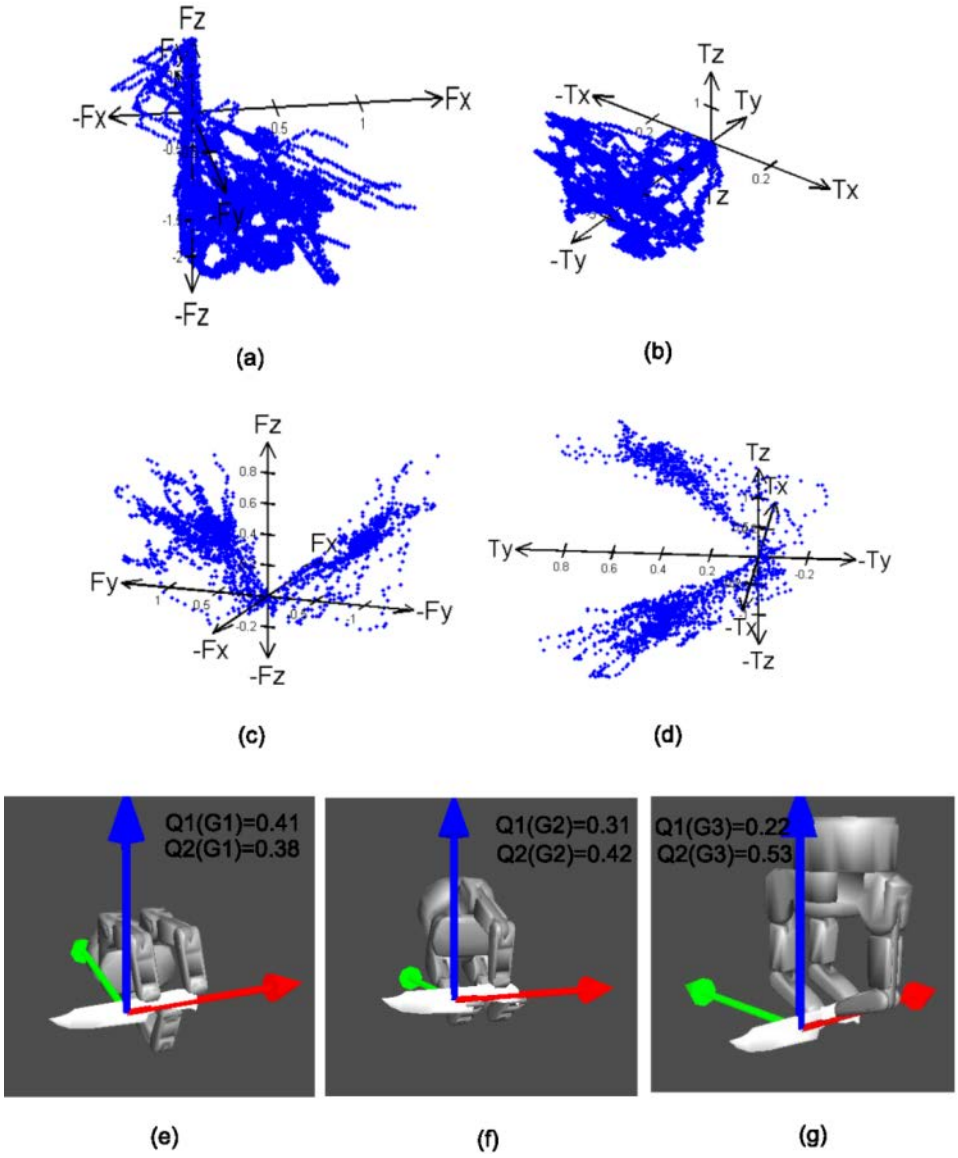


Figure 20. Planning results for a cutting task and a butter spreading task with a knife. (a)-(b): cutting task distribution of task wrenches projected to Fx-Fy-Fz and Tx-Ty-Tz subspaces, respectively, where the task wrenches are distributed mainly in -Fz and Fx; (c)-(d): corresponding task wrench distribution for butter-spreading task, where the task wrenches are distributed primarily in +Fy, -Fy, +Fz, +Tz, -Tz; (e)-(g): three different hand configurations; Q1 is the quality measure for the first task, and Q2 is the quality measure for the second task. Scale k is set to be 8.04 and 3.25 of the two tasks for a precision grasp planning.

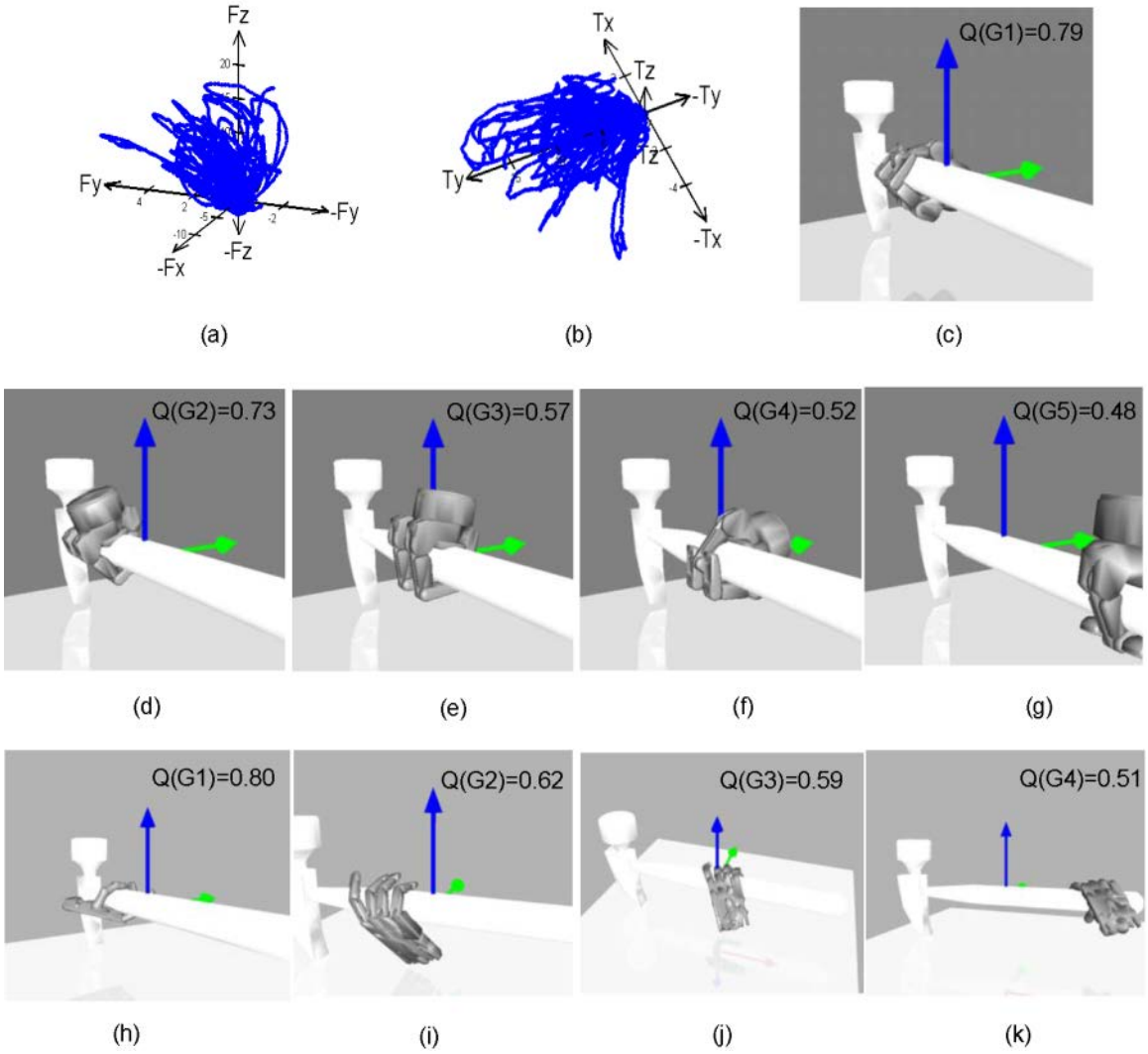


Figure 21. Planning results for a hammer. A power grasp is searched because a large power is needed. (a)-(b): distribution of task wrenches projected to Fx-Fy-Fz and Tx-Ty-Tz subspace, respectively, where the task wrenches are distributed mainly in Fz and Ty; (c)-(g): five different hand configurations of the Barrett hand model; (h)-(k): four different hand configurations of the Shadow hand model. Scale k is set to be 20.

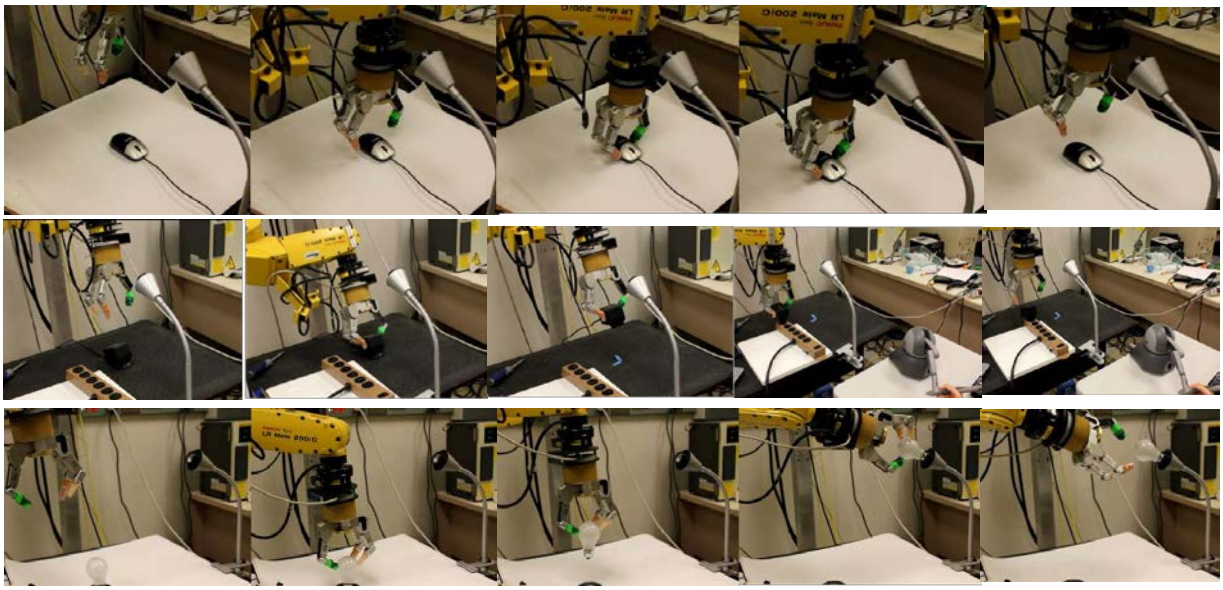


Figure 22. Snapshots of the robot execution. Top row: the robot approached the mouse, grasped it, and moved it on the table. The whole procedure was done autonomously. Middle row: the robot approached the power adaptor, grasped it, and lifted it up autonomously. Then, teleoperation by Phantom OMNI was enabled, and the manipulation task was performed by the user. Bottom row: the bulb was screwed into a socket. The whole procedure setting was similar to the plug manipulation.

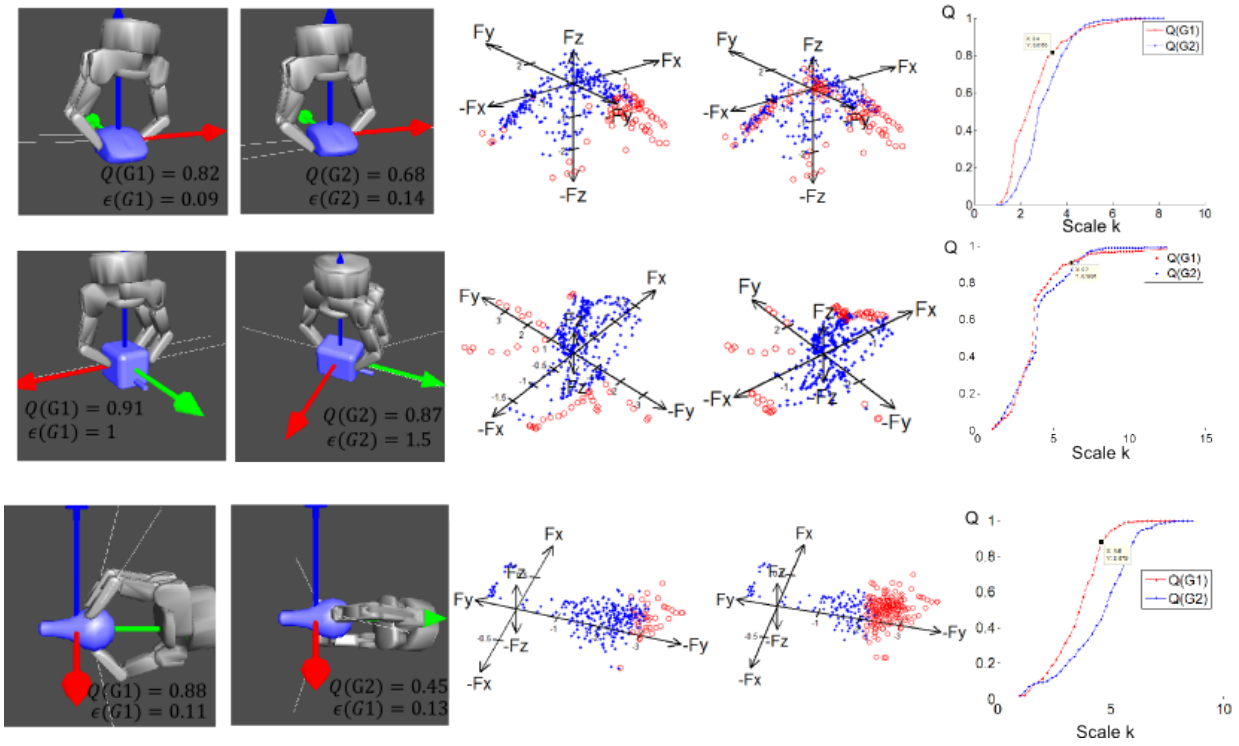


Figure 23. Simulation results for the selected tasks. The first two columns show the optimal grasps measured by our proposed quality measure Q and by the epsilon quality, respectively. The corresponding quality Q and quality ε are marked in each figure. In task 1, k = 3.4; in task 2, k = 6.2; and in task 3, k = 4.6. Columns 3 and 4 compare the coverage of task wrenches by the two grasps shown in the first two columns. Here we only visualized the wrenches in the 3-D force subspace. The wrench samples in the GWS are labeled by blue points, while the wrench samples outside the GWS are labeled by red circles. The fifth column shows the proposed quality Q as a function of scale k for the two grasps.

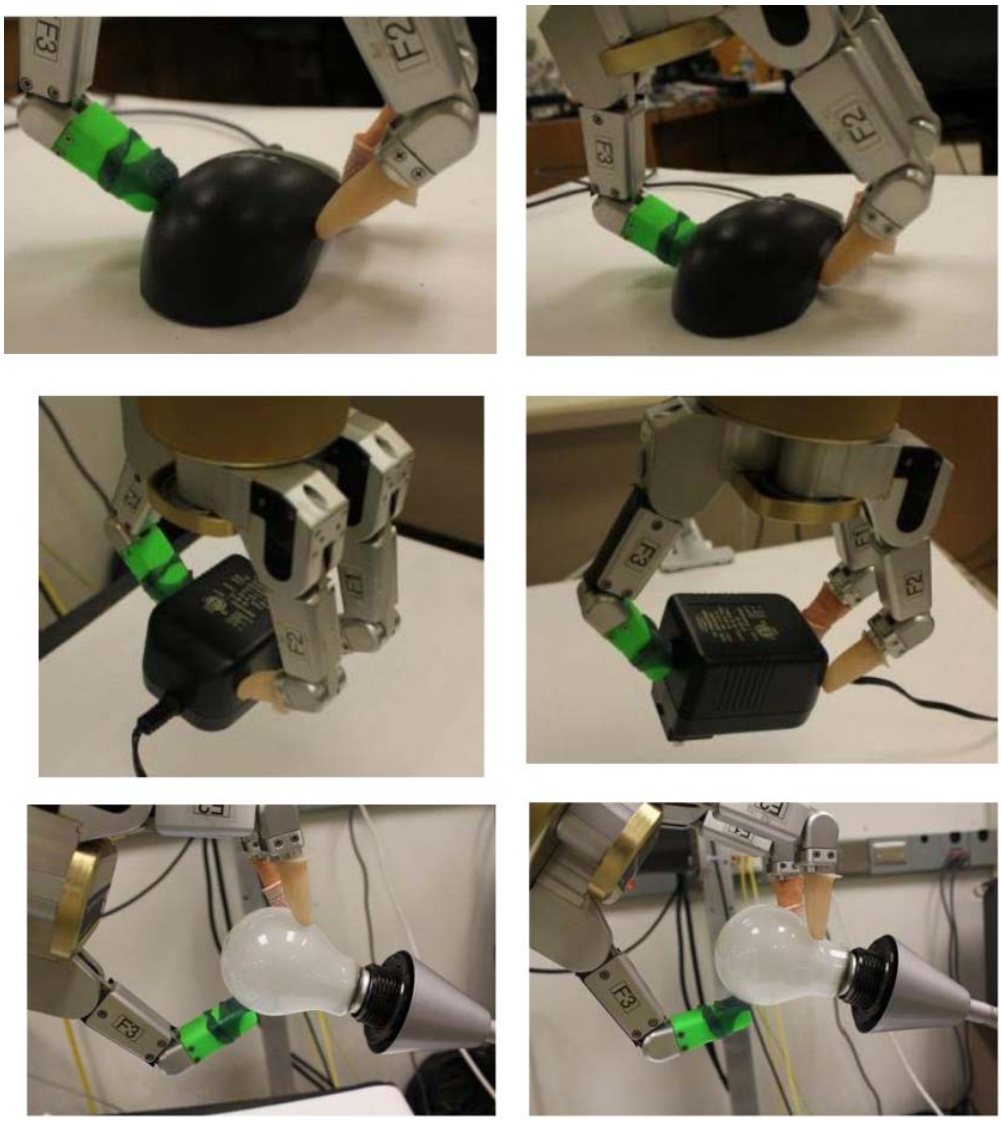


Figure 24. Evaluation of simulation on the real robotic platform. Left column: execution result from our approach. Right column: execution result from force-closure based approach.

Chapter 5

Quality Measure to Fulfill Motion Requirement

In addition to the force criterion in grasp planning, motion achieved by the grasp can also be considered as another performance criterion, similar to humans who choose grasps considering their motions.

Assume there is no dexterous in-hand manipulation in the task, such as rolling or slipping motions made by the fingers. In other words, only one rigid grasp is used throughout the whole manipulation procedure, so the relative position of the wrist frame with respect to the object frame is fixed. Therefore, the manipulation motion is achieved by moving the robotic arm. For more complicated manipulations which require a transition between different grasps in order to reach the target manipulation motion, they can be divided into sub-tasks to plan an optimal grasp for each sub-task.

5.1. Task Motion Effort Measure

Given an object and a manipulation task, there are multiple solutions to the grasps. Work in the literature considers only the force requirement to compute feasible contact points on the object, and manipulability to compute the appropriate configuration space. The optimal grasp is then chosen from all reachable grasps by force or manipulability criteria. None of the grasp planning approaches takes into account the arm motion which is affected by the wrist position. Given the wrist position included in a grasp and the target object motion, the manipulator's motion can be decided via inverse kinematics from the end-effector motion. Moreover, different manipulator motions will have different efficiency in transferring the motion from joints to the tool, resulting in different motion efforts to achieve the functional tool motion. As a result, the efficiency of the manipulation motion should be used to evaluate the grasp as well. Similar to humans who tend to manipulate the tool with less movement-

related effort, robots' motion effort should be used to evaluate the grasp as well. For example, to execute a task such as turning a tool like a screwdriver, light bulb or jar lid, one can grasp the object in one's hand in multiple ways without dropping it, but a grasp aligned with the rotational axis of the tool is favored by humans, because the motion can be accomplished by rotating only the wrist.

The idea has been adopted in motion planning algorithms. A manipulation task is usually specified in Cartesian task coordinates. e.g. end-effector coordinates, while a manipulator is controlled by its actuators, described in configuration coordinates, e.g. joint coordinates. When a manipulator is commanded to execute a motion, the command in the task coordinates must be transformed to the configuration space to control the actuators. Redundant manipulators have more degrees of freedom than that are required for a task. Theoretically in a 3-D workspace, the end-effector of a manipulator can reach an arbitrary pose with 6 degrees of freedom. Then, a manipulator with more degrees of freedom than 6 is a redundant manipulator. For a redundant manipulator, in general, there are infinite possible postures to achieve a given end-effector motion. In the literature, most techniques on the trajectory planning problem use optimization algorithm to find the optimal joint configurations. Work on control of redundant manipulators has included the use of these extra degrees of freedom to minimize the kinetic energy, at the same time avoiding joint limits and torque limits and improving obstacle avoidance.

For our work, the goal is to find grasps which result in small arm motion, quantified by the motion effort of the arm. Optimizing the motion effort yields smooth trajectories and avoids large motion changes, resulting in reducing the stresses to the actuators and to the manipulator structure. It also saves energy consumption of the actuators.

In Chapter 4, we describe a grasp as $g = [\mathbf{p}, \mathbf{x}]^T$, where $\mathbf{p} \in \Re^D$ is the vector of hand joints in joint coordinate, and $\mathbf{x} \in \Re^m$ is the vector of wrist position and orientation in Cartesian coordinate. The hand joints and the relative wrist position to the object are fixed given a grasp, so the wrist position and orientation $\mathbf{x}(t)$ is changing together with the object motion. Given a grasp and the target object trajectory

$u(t)$, the wrist motion $\mathbf{X} = \{\mathbf{x}(t) \mid t \in [0, t_N]\}$ can be computed via kinematic transformation in Cartesian coordinate.

The Cartesian trajectory of the end-effector $x(t)$ is followed by providing with sequential joint motions $q(t)$ of the arm. There is no guarantee that every end-effector pose is reachable by the arm, and then the corresponding grasp is not a feasible grasp. On the other hand, there may also be a number of solutions to the arm configurations given an end-effector pose. It involves the path tracking problem of computing a feasible joint path sequence for a manipulator, given a specified Cartesian path of the end-effector, while meeting certain constraints such as kinematic feasibility, joint velocity and joint accelerations, as well as manipulability constraints, to keep the arm out of unreasonable singular configurations. Therefore, we formulate the path tracking problem to be solving the kinematics problem subject to the constraints in the free configuration space of the physical model, and the torque/acceleration limits in the joints:

$$\begin{aligned} &\mathbf{x}(t) = f\left(\mathbf{q}(t)\right) \\ &\forall t \in [0, t_N], \mathbf{q}(t) \in C_{free}, \ddot{\mathbf{q}}(t) \in \ddot{q}_{lim} \end{aligned} \tag{5.1}$$

where f is the forward kinematic function which transforms the arm joint vector in the configuration space to the end-effector Cartesian space; the vectors of arm configurations, joint accelerations are presented as $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \Re^n$, and n is the number of DOF of the arm. C_{free} is the constraint of the configuration space in which the robot is free of collision. The solutions to the equation find joint angles of the arm given an end-effector Cartesian trajectory. Combined with the grasp, the arm would manipulate the object with the specified trajectory $\mathrm{x}(t)$.

Various methods have been developed for the robot path tracking problem, such as the analytic and numeric approaches. In this dissertation, we utilized the analytic approach implemented in the Matlab robotic toolbox [75] to compute the solutions for two robotic arms tested in our experiments: the PUMA560 robotic arm and the FANUC LR Mate 200iC robotic arm, both of which are a 6-axis robot arm

with a spherical wrist. In general there are eight sets of joint coordinates for 6-axis robotic arms that give the same end-effector pose. The solutions found by the toolbox include the left- and right-handed kinematic configurations, elbow up and down solutions, and wrist flipped and not flipped solutions. The solutions being utilized need to be consistent, so the joint motion should be continuous, avoiding sudden changes in the joint space which result in high joint acceleration.

Due to mechanical limits on joint angles and possible collisions between links, not all of the eight solutions are physically achievable. For each end-effector trajectory, we pick the solution that achieves the smallest motion effort. The arm motion is controlled by outputting joint torque from joint actuators, written as the dynamic equation:

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + J(q^T)f \tag{5.2}$$

where τ is the m-joint vector of torques from the actuator, $q, \dot{q}, \ddot{q}$ are respectively the vector of the joint, velocities and accelerations of the arm. The first term is the inertial forces due to acceleration of the joints, and M is the symmetric positive definite mass-inertia matrix. $C(q,\dot{q})$ is the Coriolis and centripetal coupling matrix, $F(\dot{q})$ is the friction force, $G(q)$ is the gravity term, $J(q^T)$ is the Jacobian matrix, and f is the external load applied to the end-effector. The last term $J(q^T)f$ is the joint torque transmitted from the end-effector.

There are various performance measures to the manipulative motion used as the cost function in the literature, such as energy consumption, motion jerks, effort, or their combinations [8]. Various performance measures can be used as the cost function depending on the given objectives, such as motion effort, energy consumption, time duration, stability, jerk, or any of their combinations. Motion effort measures the effort made by actuators. Torques are applied to each joint by the actuators, so the measure of the effort is defined as the integration of squares of all joint torques over time during the motion:

$$Q_e = \int_{t_0}^{t_N} \tau(t)^T \tau(t) dt \tag{5.3}$$

where $\tau = [\tau_1, \ldots \tau_k]^T$ is the vector of joint torques, k is the number of arm joints. Note that different weights can be put to emphasize some of the joints such as the joint on the shoulder or robot base.

By comparison, one can quantify the motion performance by the square of the mechanical energy:

$$Q_{se} = \int_{t_0}^{t_N} (\tau(t)\dot{q}(t))^T (\tau(t)\dot{q}(t))dt \tag{5.4}$$

According to Equ. 5.3, the amount of motion effort depends on joint torques, so a small measure of motion effort implies an ease of the actuators. The motion effort relates closely to the energy of the system. According to the dynamic equation in Equ. 5.2, joint torques depend on the joint motion (joint displacement, velocity and acceleration) and the external load applied to the end-effector. Therefore, a small motion effort also implies a smooth joint motion. Equ. 5.4 measures the mechanical energy consumed during an arm movement. If there is a large joint torque but a small motion, the equation may still get a small result. In our experiment, we used the motion effort to evaluate the grasps. The force and motion criteria evaluate the grasp from two distinct aspects. Grasps can be evaluated by a combination of the two criteria. The grasp can be selected by the two quality measures in several ways. For example, the grasp can be selected by either sequential evaluations under these two criteria, or a global criterion combining them together. The way of selecting the optimal grasp should take into concerns both applications and the hand capability.

5.1.1. Computational Efficiency of Task Motion Effort

To compute the task motion effort of a grasp, the path tracking problem has to be solved. The efficiency to solve the path tracking problem depends on the algorithm. The numeric approach is usually used to solve path tracking for complicated redundant manipulators, by iteratively converging into goal positions. Here we use the analytic approach to solve the problem for the 6-axis FANUC i200C and PUMA 560 arm. Computing joint angles from each way point on the end-effector trajectory in Cartesian space takes $O(k)$ time, where k is the number of joint variables. Similarly, solving the inverse dynamics problem using the recursive Newton-Euler algorithm takes $O(k)$ time, the same as the motion

effort computation. Therefore, the computational efficiency for the whole trajectory takes $O(kN)$ time, where N is the number of way points on the trajectory.

5.2. Experiment on Grasp Quality Measures

We describe the grasp planning problem by finding the optimal grasp in terms of both force and motion requirements, measured by the wrench coverage Q_w (described in Chapter 4) and the arm motion effort Q_e (described in Chapter 5), subject to physical constraints such as velocity and accelerations. We compared the two grasp measures Q_w and Q_e for different grasps and objects, as summarized in Figure 26. The comparison was tested with the Barrett hand and the Fanuc arm. Three grasps were selected for each object, sorted by the wrench coverage measure in a descending order, as shown in the left of Figure 26. It can be observed in the right figure that the order of motion effort with the corresponding grasps differs from object to object, and there is no correlation between the two grasp measures. For the light bulb task, grasp 1 is the best of the three if measured by the task wrench coverage, but if measured by motion effort, grasp 3 is the best of the three if measured by the task motion effort; for the hammer task, grasp 1 is the best evaluated by both measures; and for the cup task, grasp 2 is the best if measured by the task motion effort. It is also in accordance with the physical world that there is no correlation between the two grasp measures. Therefore, these two factors should be considered comprehensively for specific robots and tasks.

Two selected grasps of each object are visualized in Figure 27 as detailed illustrations. Each grasp is marked by three grasp measures: force-closure property ε [38] and the two proposed quality measures. In the left column, the "screw in a light bulb" task, the task quality measure Q_w is 0.92 for the grasp shown in the upper figure, meaning it has larger task coverage than that in the lower figure. The task independent force-closure property ε is not coincident with the proposed task wrench coverage measure. Although the ε value of the lower grasp is higher than the higher grasp, it has a lower ability to resist disturbance that occurs in the specific lightbulb task. The motion effort $Q_e = 1.54$ of the upper

grasp is also much less than the lower grasp, because it requires only the wrist motion to screw the light bulb. Therefore, the upper grasp is “better” than the lower grasp, in terms of both task wrench coverage and motion effort.

In the middle column, the upper grasp has a higher ability than the lower to resist the disturbance that occurs on the head of the mallet (we have discussed in Chapter 4 this contradicts the human common sense of grasping a hammer, due to the limitation of the grasp model, which only considers the wrench can only be achieved by fingers via contacts, but does not consider that the striking force can be achieved by arm motion), but it requires larger motion effort to achieve the striking motion. Different grasps can be selected in different situations. For example, the lower grasp may be chosen for a weak hand to better resist both heavy gravity as well as striking force. However, for the lower grasp, a larger torque is applied to the hand from the head of the mallet, but the manipulation can be achieved with less motion effort.

In the right column, all three measures of the grasp in the upper figure are “better” than the lower grasp. Figure 31 shows the manipulation process of a cup as an example to compare the two different grasps.

According to the results, there is no correlation between the two grasp measures. The two quality measures can be combined in different ways according to the applications and task requirements.

The two criteria can be combined as one global measure using a weighted sum, but the performance depends on different weights. If the execution efficiency is of more concern, then a larger weight can be put to the motion measure. Also, the computation of the motion effort is more expensive, for it involves the iterative computation of inverse kinematics at every object pose.

Another solution is to combine the two measures in a serial way. For example, by setting a threshold for the task wrench coverage criterion, the set of preliminary candidate grasps can be computed, without computing arm configurations. The candidate grasp set is, therefore, independent of the arm kinematic model, as well as the position and orientation of the target object relative to the

manipulator, so it can be generated off-line. Then, each grasp in the candidate grasp set is measured by the motion effort criterion. The grasps which cannot meet the reachability and acceleration constraint are rejected. The optimal grasp is then selected under the measure of the motion effort. This computation of a grasp is in coincidence with the intuitive consideration in the physical world, because the grasp should be stable, in the first place, to ensure the object is not to be dropped under the outside disturbance; then the motion effort is considered, in the second place, to yield a small and smooth arm motion. For the hammer example shown in the top column in Figure 28, if the threshold is set to be a value higher than 0.63, G(1) is selected out of the four grasps because only G(1) meets the force requirement. G(2) will not be selected, since it is unreachable by the arm. If the threshold is set to be a value lower than 0.63, then G(3) is selected because of its low motion effort to execute the manipulation. In this case, the threshold task wrench coverage criterion decides the choice of grasps. In contrast, for the bulb example shown in the bottom column, in terms of both quality measures, G(1) is the best out of the four listed grasps. The optimal grasp is not affected by the choice of threshold. The simulated execution for two grasps of the hammer and bulb are presented in Figure 29 and 30 respectively. And the execution result of the cup on a real robotic platform is shown in Figure 31.

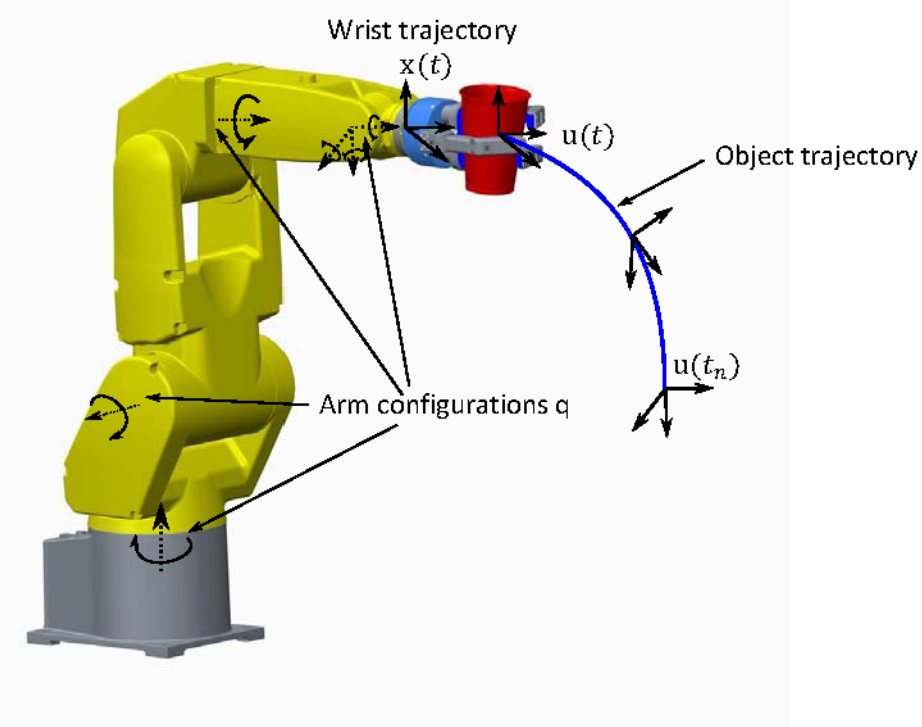


Figure 25. An example of grasp and manipulation.

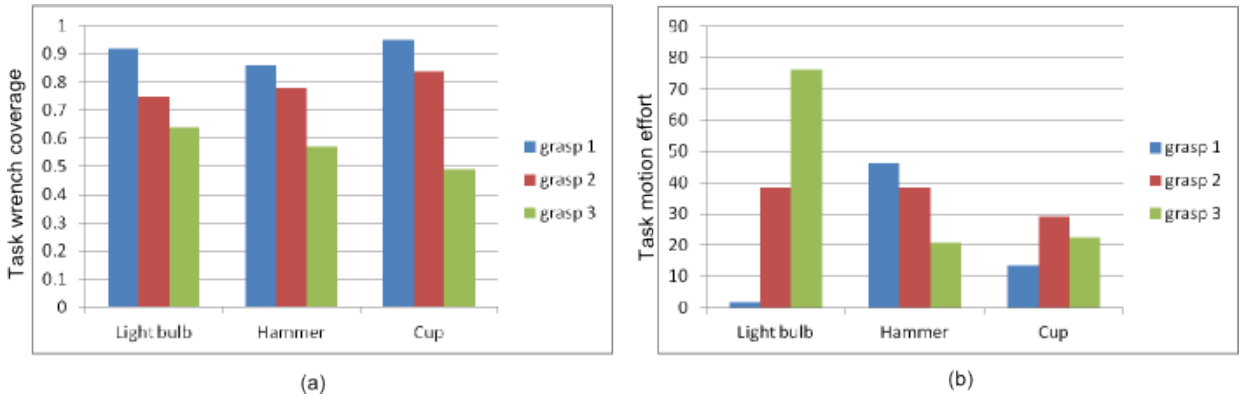


Figure 26. Measures of wrench coverage and motion effort of different grasps. Left figure: wrench coverage measure. Right figure: motion effort measure.

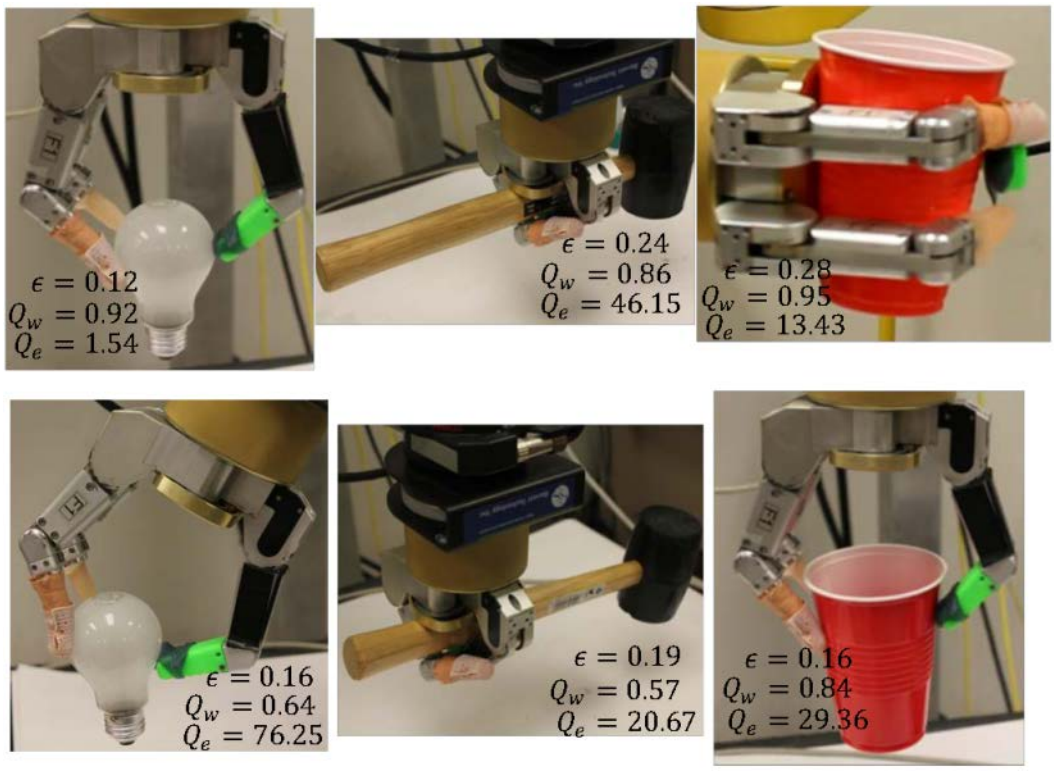


Figure 27. The grasps tested in the experiment.

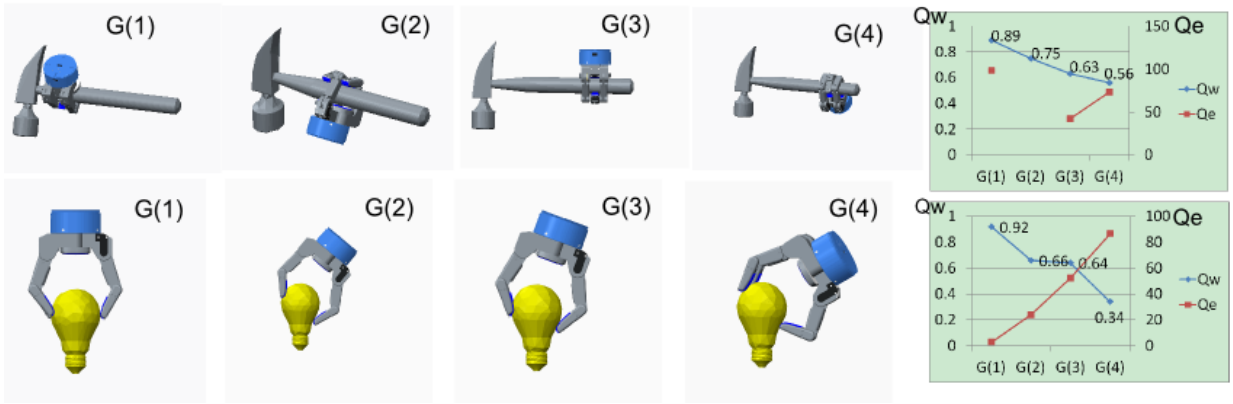


Figure 28. Example of grasps for the hammer and the light bulb. The right column shows the two quality measures for the grasps shown in the left four columns. G(2) of the hammer is not reachable by the robotic arm, so there is no measure of the task motion effort $\mathbf{Q_e}$.

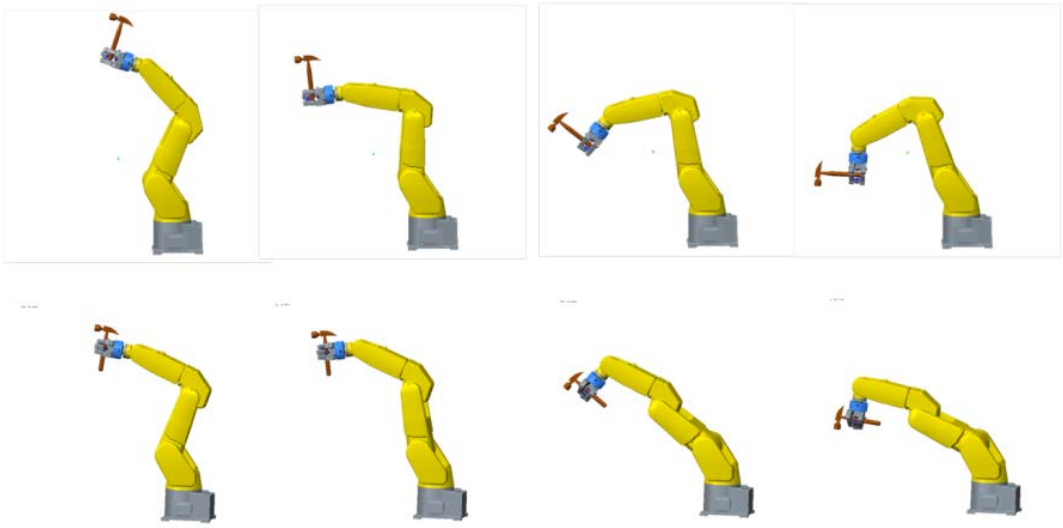


Figure 29. Comparison of two grasps to execute the hammer task.

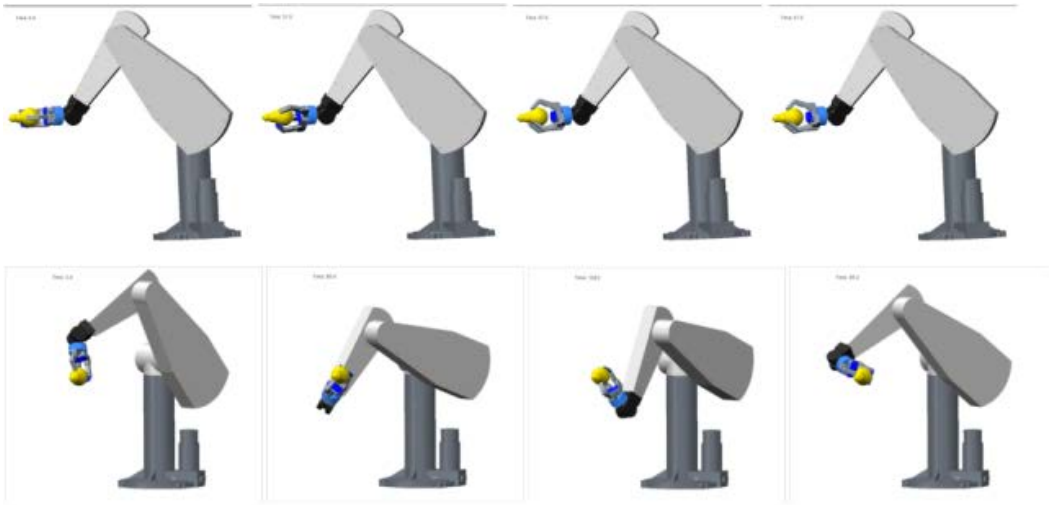


Figure 30. Comparison of two grasps to execute the light bulb task.

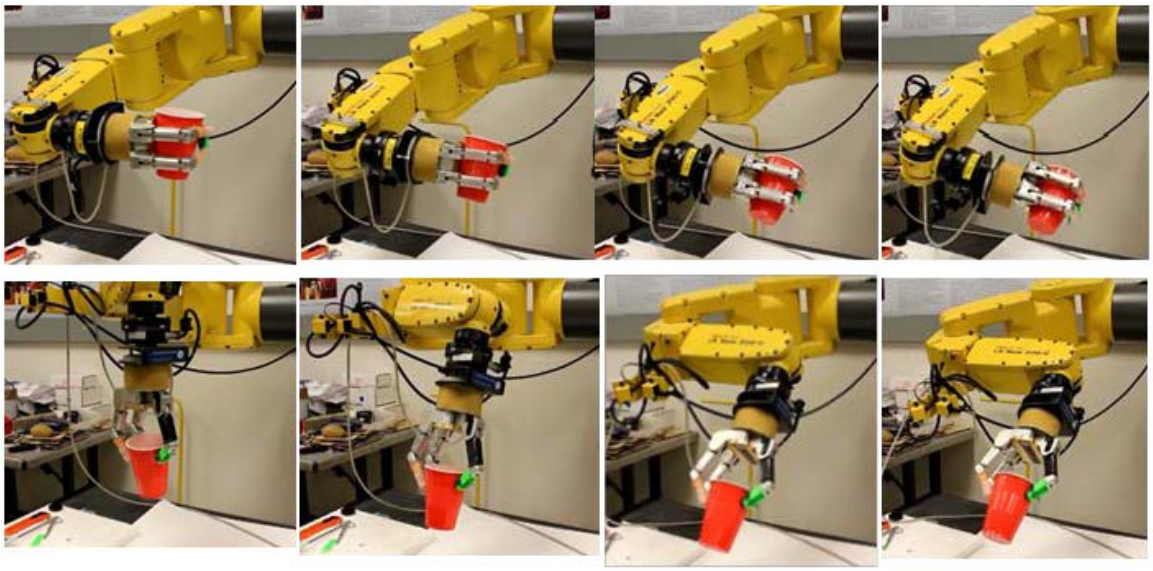


Figure 31. Compare two different grasps to execute the same manipulation motion of a cup.

Chapter 6

Incorporation of Human Grasp Strategies to Constrain Grasp Planning[3]

There is general agreement that the crucial feature distinguishing the human hand from other apes is the opposable thumb [1, 76-79]. The long, strong and mobile opposable thumb, combined with the other shortened fingers, make possible sophisticated manipulative skills and tool using, whereas the hands of apes, who are considered to be closest to humans, are evolved mainly for climbing but not for manual dexterity. Without an opposable thumb, humans would not have been able to develop the efficient grasping and dexterous manipulation skills.

In [1], the opposable thumb characterizes humans by two basic grasp types for all prehensile activities: the precision grip, in which objects are gripped by only the fingertip of the thumb and some fingers, and the power grip, in which objects are wrapped around by the thumb and some fingers. Derived from these two basic grips, Cutkosky and Wright [2] developed a hierarchical tree of grasp types to facilitate the design and control of robotic hands for various manipulation purposes. Almost all grasp types in the taxonomy, even as simple as a two-finger precision grip, are formed by the thumb and some other fingers. The grasp types defined in other alternative grasp taxonomies (e.g. [3]) also have similar features. The only exception is the non-prehensile grasp which does not require the thumb at all. Similarly in robotics, Arbib et al. [80] simplified the robotic models to a thumb and virtual finger (defined by a set of fingers opposing a thumb). Such robotic hand models include all the popular hand models such as the NASA Robonaut hand [81], the Shadow hand [82], Utah/MIT hand [83] and the DLR Hand II [84], PR2 gripper, etc. To meet a specific task requirement and object shape, the robotic hand

[3] This chapter was published in Proc. IEEE Intl. Conf. Robotics and Automation, pp. 1068-1073, 2013. Y. Lin and Y. Sun, " Grasp Mapping Using Locality Preserving Projections and KNN Regression," Permission is included in Appendix A.

can choose one particular grasp type from the taxonomy. Once the thumb is placed on the target object, the other fingers can be placed naturally against the thumb to form that particular grasp. Clearly, the thumb and grasp types are closely related and they are two critical features that characterize a human grasp.

In this chapter, we propose a novel grasp synthesis approach that integrates two human grasp strategies – grasp type, and thumb placement (position and direction) – into grasp planning. The proposed technique fits in the learning from demonstration (LfD) framework that allows robots to learn skills from humans. One application of LfD is to abstract human grasping strategies from demonstration such that the skill abstraction can meet two important criteria: it should reflect the demonstrator's intention, and it should be general enough to be used by a robotic hand. Different abstractions of human grasp constrain the grasp synthesis and narrow down the solutions of grasp generation to different levels. If imposing a strict constraint, such as defining all contact points of the fingers on the object, it loses flexibility and becomes rarely achievable for a robotic hand with a different kinematic model. Thus, the choice of grasp strategies should balance the learned constraints and required flexibility to accommodate the difference between the human hand and robotic hands. The human strategies of grasp type and thumb have such a balance while conveying important human intents to the robotic grasping.

Grasp types abstract the manner in which a human grips an object for a manipulation. One example is the procedure of screwing on a jar lid, which starts with a precision grasp using only the fingertips, because the jar lid is loose at the beginning and can be screwed efficiently and flexibly without much power. As the jar lid gets tighter, one may switch to a power grasp to apply larger force on the jar lid.

Nevertheless, grasp types cannot solely form appropriate contact points on the object. It has to be combined with the suitable grasp region and direction. Thus, we learn thumb placement and direction as the second grasp cue from humans, mainly for the two reasons discussed above. Firstly, almost all robotic hands have a long and strong opposable thumb, although robotic hands are usually simplified to different extents from the human hand, so thumb placement and direction is also independent

of robot kinematics. Secondly, the thumb plays a key role in gripping an object efficiently, according to the anthropology research. Thumb placement indicates which surface or part of the object is to be gripped. It also constrains the workspace of the wrist and other fingers while preserving necessary flexibilities to meet kinematic constraints.

The grasp type and thumb placement establish a grasp strategy that confines the hand configuration space, but they cannot fully determine a grasp or guarantee that a grasp is stable. The confined space leaves enough room for grasp planning to find the optimal stable grasp which is adapted to different robotic hand models.

In this chapter, we provide a detailed discussion on how the proposed grasp strategies can be obtained, and how they are beneficial for establishing a robotic grasp. Grasp type can be input by the human or recognized from human demonstration. A novel recognition approach using trajectory data at the joint level is presented in Section 6.1. Thumb information can also be observed from human demonstration, or labeled directly on the object surface, as discussed in Section 6.3. We believe that grasp type and the thumb are simple, effective and informative representations of the grasp features.

6.1. Task Taxonomy and Classification

The grasp type can either be input by the human teacher, or recognized from the human demonstration. In this section, a novel grasp recognition approach using grasp motion trajectories at joint level is presented.

Many grasp classifications are defined based on Cutkosky's grasp taxonomy [2], which classifies user performed grasps into 16 classes that vary by task requirement and dexterities. To recognize the demonstrated grasp as a type from the Cutkosky's taxonomy, pattern classification techniques can be applied.

Ekvall and Kragic [53] used Hidden Markov models to recognize grasp type from the taxonomy based on an entire grasp sequence. The recognition rate was 97% for a single user and known objects existing in both the training and test dataset when there are 10 grasp types. The recognition rate dropped

to 65% for known objects, and unknown users that were not in the training dataset. There is no information on performance if unknown objects were tested. Aleotti and Caselli [54] performed grasp recognition using static grasp poses in virtual reality for six grasp types, with a recognition rate of 94% for two expert users without using real objects. Heumer et al. [55] compared the different classification methods for recognizing six grasp types. The best recognition rate of all the methods was 83.82% for known users and 71.67% for both unknown users and objects.

Our previous work [85] has shown that trajectories of the hand joints provide more information than static poses. It is necessary to disambiguate between different grasp types that share similar static poses but differ in grasp trajectories, because some similar poses belonging to different classes in the human hand configuration space may be far apart from each other in the robotic hand configuration space. For example, the lateral pinch and small wrap have similar measures on joint angles using a data glove, whereas, due to much less dexterity in some robotic hands, e.g., a Barrett hand, the lateral pinch has to be performed in a way that is distinguishable from the small wrap (Figure 32).

Human hand joint motions can be treated as a high dimensional time series. Given a training dataset of grasp motion sequences, the high-dimensional hand motions in the dataset usually have undesirable properties that bias the learning results. Dimensionality reduction is a typical approach for finding a lower intrinsic dimensionality of data while removing the undesirable noise and variance and leaving a minimum number of needed properties of the data. Typical methods for dimensionality reduction include linear methods, such as principal component analysis (PCA) [86] and linear discriminant analysis (LDA) [87], both of which find a linear transformation of high-dimensional data to their low-dimensional counterpart. Nonlinear methods, such as isometric feature mapping (Isomap) [88], local linear embedding (LLE) [89], and Laplacian Eigen map (LE) [90], can model the manifold of the data in the high-dimensional space. These nonlinear methods do not provide a nonlinear transformation that project new data points to the latent low-dimensional space.

For dimensionality reduction, we use locality preserving projections (LPP) [91] to find the low dimensional manifold of the training motion data. LPP is a linear technique that combines the benefits of both linear and nonlinear methods [91]. It finds a linear mapping function that minimizes the cost function of LEs; thus, new demonstrated data can be easily projected to the low-dimensional space by a linear transformation computed by LPP. In the lower dimensional space, the k-nearest neighbored trajectories to the demonstrated motion can be found, by computing the similarity between trajectories. One approach to measure the similarity between trajectories is Hausdorff Distance. Then, the grasp is recognized by its nearest neighboring trajectories in low-dimensional subspaces.

6.1.1. Dimensionality Reduction Using LPP

The problem of linear dimensionality reduction is briefly described as follows based on [91]. Given a set of data points $x_1, x_2, ..., x_n$ in high dimensional space $\Re^D$, find a transformation matrix A that maps these n points to a set of points $y_1, y_2, ..., y_n$ in low dimensional space $R_d (d << D),$ such that $y_i = A^T x_i$, where i is among 1 to n.

The first step is to the adjacency graph. A weighted graph $G = (V, E)$ with n nodes can be constructed from the data set. An edge is put between the nodes i and j. If x_i and x_j are neighbors, which can be defined by either ε-neighborhoods ($\| x_i - x_j \|^2 < \varepsilon$) or k-nearest neighbors. We do not choose ε-neighborhoods, because choosing an optimal ε relies on a good understanding of the data.

The second step is to choose the weights. W is a sparse symmetric $m*n$ matrix with w_{ij} having the weight of the edge joining vertices i and j. To better separate the classes, we set $w_{ij} = 0$ if the node x_j is not in the k-nearest neighbor of the node x_i, or the nodes x_i and x_j are not in the same class; otherwise, w_{ij} is defined by heat kernel, $w_{ij} = exp(-\| x_i - x_j \|^2 / t)$, justified by Belkin and Niyogi [92]. Parameter $t \in \Re$ is the heat kernel factor.

The third step is to compute computing eigenmaps. Solve the generalized eigenvector problem:

$$XLX^T\vec{a} = \lambda XCX^T\vec{a} \tag{6.1}$$

where C is a diagonal matrix whose entries are column (or row) sums of the symmetric W, i.e. $C_{ii} = \sum w_{ji}$; $L = C - W$ is the Laplacian matrix; the i^{th} column of matrix X is the i^{th} data point x_i. The solution $\vec{a}$ is the column vector of the transformation matrix A.

In our application, LPP is performed on the motion sequence of finger joints, which ignores the time component, so the result is invariant to time and speed. Figure 33 shows the three-dimensional visualization of the 14 dimensional hand-joint trajectories captured by a dataglove for 12 different grasp types in Cutkosky's taxonomy. It demonstrates the ability of LPP to preserve the locality of the nonlinear structure. Although there is partial overlapping between two classes (such as the beginning of the motion sequence because the hand is initially open for all grasp types), there are distinguishable variances among different classes of grasp sequences but little in-class variance.

6.2. Grasp Recognition by Similarity Among Motion Trajectories

A demonstrated grasp trajectory can be recognized as a grasp type by measuring the similarity between the demonstrated grasp trajectory and the trajectory in the training dataset. The similarity is defined by Hausdorff distance, described as follows: let X and Y be two motion trajectories, and the Hausdorff distance from X to Y is represented as:

$$d_H(X,Y) = \max_{x\in X}(\min_{y\in Y}(\| x - y \|)) \tag{6.2}$$

where x and y are data points in trajectories X and Y, respectively. The distance from Y to X is represented as:

$$d_h(Y,X) = \max_{y\in Y}(\min_{x\in X}(\| x - y \|)) \tag{6.3}$$

The distance between the two trajectories X and Y is defined by:

$$D_H(X,Y) = \max(d_h(X,Y), d_h(Y,X)) \tag{6.4}$$

The Hausdorff distance handles three cases of similarity between grasp motion sequences, as illustrated in Figure 34. The two trajectories start from approximately the same position because they share the same initial pose.

Figure 34a demonstrates Case 1, where trajectory Y is roughly a part of trajectory X. This usually happens for the same grasp types but slightly different object sizes. The inter-trajectory distance, therefore, becomes the distance between the end poses of X and Y.

In Case 2 (Figure 34b), trajectory X and Y share the same start and end points but differ in intermediate paths. This usually happens when the two grasp types are different but share a similar end pose, such as a lateral pinch and a small wrap, which actually spans a larger Euclidean volume in robotic hand configuration space. In this situation, Hausdorff distance is beneficial for distinguishing between two grasp types that share ambiguous grasp poses.

Case 3 (Figure 34c) is the general case, in which trajectory X and Y differ in intermediate paths as well as end points.

Hausdorff distance can also be modified to other metrics, such as mean pairwise distance, depending on the applications.

The kNN method is used to recognize the demonstrated grasp sequence as a grasp type to which the majority of its K nearest neighbors belong.

6.3. Integration of the Extracted Strategies into Planning

In this section, a Barrett hand model is used as an example to illustrate how the search space of planning is reduced by one thumb position and constraint by grasp types.

According to Equ. (4.1), the grasp quality measure is a function of hand posture p and wrist position$w$:

$$Q = f(p, w) \tag{6.5}$$

where $p \in \Re^D$, $w \in \Re^6$. Grasp planning is to search for the optimal grasp in a $D+6$ space. D depends on the degrees of freedom of the robotic hand. For a Barrett hand, $D = 4$ and p is a four-joint angle vector:

$$p = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} \tag{6.6}$$

where θ_1 , θ_2 , and θ_3 are three flexion/extension angles of each finger and θ_4 is one abduction/adduction. Therefore, the quality function Q has 10 variables. An optimal algorithm is needed to search in the high 10-dimensional space for the maximum value of the quality function Q. Here, how the search space is reduced to 3 by teaching relative thumb positions on the object (Figure 35) is discussed as follows.

Figure 36 illustrates kinematics of a Barrett hand. Define the three fingers of a Barrett hand to be $F1, F2$ and $F3$, respectively. Let $\{N\}$ represent a world inertial frame fixed in the workspace. $\{B\}$ is fixed to the object origin. A frame $\{n_w\}$ is fixed to the wrist of the robotic hand. Frame $\{n_{ik}\}$ with axes $\{x_{ik}, y_{ik}, z_{ik}\}$ is attached to each joint of the robotic hand, where $i = 1,2,3$ is the i^{th} joint of each finger, $i = 3$ represents the end-effector and $k = 1,2,3$ is the k^{th} finger. z_{ik} is the rotational axis of joint ik ; x_{ik} is the axis along the link ik 's principal axis; y_{ik} is perpendicular to z_{ik} and x_{ik} . Let θ_{ik} denote the joint displacement rotating around rotational axis z_{ik} . Finger $F3$ is usually defined as the thumb, except for one case when lateral pinch is applied (see experiments in section 6.4).

Let $u_h \in \Re^6$ denote the vector representing the position and orientation of the center of a human thumb relative to an object coordinate (Figure 35). u_h can be mapped to u_r , the center of the thumb fingertip of the robotic hand, via linear translation. Another way of representing the position and orientation of the robot thumb fingertip with respect to $\{B\}$ is a homogeneous transformation matrix:

$$A_{33} = \begin{bmatrix} R_{33} & d_{33} \\ 0 & 1 \end{bmatrix} \tag{6.7}$$

where R_{33} is the 3×3 rotation matrix and d_{33} is the three-position vector. If the value of thumb positions and orientations are completely mapped to the robotic hand, contact points of the other fingers on the

object may vary tremendously caused by kinematic difference; then, a nonforce-closure grasp may be applied by the robot. Therefore, we exclude θ_{33}, the pitch of the thumb fingertip relative to the object coordinate, from the mapping. Hence, the matrix A_{33} is not constant, but varies as the joint displacement θ_{33} around axis Z_{33} adapted to a new value to achieve a force-closure grasp. So, A_{33} is a function of a single joint variable θ_{33}:

$$A_{33} = A_{33}(\theta_{33}) \tag{6.8}$$

Similarly, A_{i3} is the homogeneous transformation matrix of the i^{th} joint with respect to the frame $\{n_{(i+1)3}\}$. Thus, the homogeneous transformation matrix that expresses the position and orientation of the wrist with respect to the thumb fingertip $n_{(i+1)3}$ is denoted by ${}^{33}T_w$:

$${}^{33}T_w = A_{33}(\theta_{33})A_{23}(\theta_{23})A_{13}(\theta_{13})A_w \tag{6.9}$$

where joint angles θ_{13} and θ_{23} are moved jointly by one motor. θ_{13} and θ_{23} can be determined by the motor-to-joint transform from motor revolution θ_3:

$$\begin{bmatrix} \theta_{13} \\ \theta_{23} \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix} \theta_3 \tag{6.10}$$

The position and orientation of the wrist can be represented as a function of θ_{33} and θ_3 given by the transformation matrix ${}^{33}T_w(\theta_{33}, \theta_3)$. Therefore, the six position-and-orientation vector w in Equ. 6.5 can be determined as a function of θ_{33} and θ_3. Combining Equ. 6.5, 6.6, 6.9 and 6.10, Equ. 6.5 is expressed as a function of:

$$Q = f(\theta_1, \theta_2, \theta_3, \theta_4, \theta_{33}) \tag{6.11}$$

θ_3, θ_4 and θ_{33} determine the position and orientation of the wrist, thumb flexion/extension, and hand adduction/abduction. The flexion/extension of the other two fingers, θ_1 and θ_2, can be determined easily by a simple execution of a close command, so that θ_1 and θ_2 are commanded to conform to the surface of the object. Hence, Equ. 6.11 can be simplified as:

$$Q = f(\theta_3, \theta_4, \theta_{33}) \tag{6.12}$$

Mapping only thumb position from a human hand to a robot hand is simple because there is little correspondence problem and it can be easily generalized to different robotic hand models. By learning partial hand information from the demonstration, we know a reference contact position on the object, and the search space for optimization during the planning procedure is reduced to three. Figure 37 compares the 2-D workspace of the wrist with and without thumb constraint extracted from human demonstration on grasping a ball. In Figure 37(a), without any constraint, the wrist can move around the circle; while in Figure 37(b), with the thumb constraint, the workspace of the wrist is constraint to rotate around only one contact point between the thumb and the object.

The feasible configuration space is also bounded by the desired grasp type. In addition, grasp type also affects the number of variables to be optimized. The abduction/adduction angle θ4 is under planning only for a sphere-type grasp, meaning that an abduction is needed for the task. In a sphere grasp planning, further simplification can be done to select one grasp from a set of candidate grasps, by dividing the search procedure into two stages. In the first stage, a search is run on a 2-D space, θ_{33} and θ_3. The adduction/abduction angle θ_4 is searched after θ_{33} and θ_3 are established, to reduce the number of resulting combinations of adduction/abduction angles and flexion/extension angles. Therefore, the computational complexity of the optimization procedure is reduced to $O(n^2)$.

An example of the searching procedure involving a Barrett hand grasping a sphere is presented in Figure 38, which shows snapshots of the current hand posture during optimization. Postures bordered by a black box are grasps with an epsilon quality larger than 0.1. During the execution, the contact point between the thumb and the object remains the same, while the pitch of the thumb changes.

6.3.1. Generalization to Other Robotic Models

The proposed method can also be extended to other robotic models. Take the Shadow hand for instance, which is designed to closest approximate the human hand [82]. The objective function is the

same to Equ. 6.5, $Q = f(p, w)$, where $p = [\theta_1, ..., \theta_D]^T$ is a vector of finger joints, and D is the number of finger joints; $D = 22$ for a Shadow hand excluding two additional joints in wrist. Similar to the Barrett hand, once the thumb position is extracted from demonstration, the wrist position and orientation w can be determined via forward kinematics by both the relative pitch of the thumb to the object, and the thumb joint variables. The workspace of the wrist is highly constrained by the posture of thumb. Therefore, the objective function becomes:

$$Q = f(p, \gamma) \tag{6.13}$$

where γ is the thumb pitch relative to the object. The search space of the optimization depends on the DOFs of the hand joints p.

Further dimensionality reduction can be performed on hand DOFs using some existing approaches. The idea of dimensionality reduction on finger joints was proposed by Santello etc. [93], who performed the principal component analysis on the human hand motion data and revealed that the first two eigengrasps (mainly flexion/extension and adduction/abduction) capture more than 80% variance of the grasp motions, implying a substantial simplification on hand postures. The idea of eigengrasp was applied by Ciocarlie and Allen [94] in grasp planning, where the DOFs of the hand and wrist were reduced to eight. The two dominant eigengrasps are sufficient for power grasps, where less dexterity is desired. In dexterous manipulation tasks, however, a different choice of simplified DOFs is needed for precision grasps.

Hand posture p can be described as a function g of the reduced DOFs e_i, written as:

$$p = g(e_i), i = 1...d \tag{6.14}$$

where $d << D$ is the number of reduced DOFs. The function g and the value of d differ by grasp types. The wrist position can be determined by thumb posture, written as:

$$p = h(e_i, \gamma) \tag{6.15}$$

Therefore, Equ. 6.5 can be simplified and generalized as:

$$Q = f(e_i, \gamma) \tag{6.16}$$

Hence, the dimensionality of the searching space equals $d+1$. If the number of DOFs of finger joints is reduced to two, then the searching space has three dimensions. An example of power sphere grasp planning is illustrated in Figure 39, searching through the subspace of relative pitch γ.

6.3.2. Generalization on Thumb Placement

Thumb placement can be obtained by observing human demonstration. Observing human demonstration puts one thumb placement constraint on the grasp searching which assumes that the same thumb contact point is reachable by the robot. Alternatively, a set of thumb placements can be labeled directly on the object surface. Therefore, the candidate grasp can be computed from a set of given thumb placements on the object surface considering the kinematics limitations as well as the reachability properties.

The upper-left of Figure 40 shows an example of the labeled area. The participants were asked to label colors on the object to indicate the contact area and direction of the thumb. The thumb can be placed only on the colored area, with different colors specifying different thumb directions relative to the principal axis of the object. Thumb placement in the red-colored area can be pointed only to the object's axis x, while thumb placement in the green-colored area can be pointed only to the object's axis y, and so on. The following figures of Figure 40 show snapshots of a searching procedure of a power grasp throughout the constraint area of thumb placement, where a power grasp is specified.

The number of grasps to be searched depends on simulation settings, such as resolution of the object model, step size of the search space, and the physical robotic hand. Taking the tote tool in Figure 9 for example, for one single constraint of thumb placement, if the step size of searching is set to be five degrees in the configuration space, around 250 candidate grasps would be evaluated. The object model has 5666 faces in total, with 161 labeled thumb placements and directions on the handle. Thus, 40,250 candidate grasps would be evaluated in overall with the thumb labels, whereas without the constraint, the hand can be put anywhere in any direction on the object, then the number of grasps will be

much larger. Instead of an exhaustive search, of course, further improvement can be done for the optimization procedure.

6.4. Experiment

6.4.1. Evaluation on Grasp Type Recognition

We measured the sequence of hand motions for grasp type extraction using a right-handed 5DT dataglove 14 Ultra, with 14 fiber optic-based bend sensors measuring the amount of bending, shown in Figure 41. It captures proximal interphalangeal (PIP) articulations and metacarpophalangeal (MP) joints for all the five fingers, and MP joints between adjacent fingers. The flexure resolution is 12-bit for each sensor, and the minimum dynamic range is 8-bit. The sensor does not measure real joint angles. Instead, it measures the proportion of bending of the measuring joint in its full range of motion. The bending values are scaled in between 0 and 1, with 0 being fully open and 1 being fully closed. Hand motions from fully open to close were sampled at a rate of 100 Hz.

In the experiment, twelve human grasp types were defined for the human hand, as illustrated in Figure 41. Eighteen everyday objects were tested (Figure 42). Each object is associated with some predefined grasp types. For example, the screwdriver is associated with small wrap grasp and adducted thumb grasp, and the dry erase marker is associated with precision grasp and small wrap grasp. Four subjects participated in the experiment and they were asked to grasp each of the objects using the designated grasp types. They were taught every grasp type by an expert. To demonstrate a grasp, they initially opened their hands, and the object was placed in front of their palms. Then, they closed their fingers to grasp the object as instructed. Each demonstration was performed for five trials.

For kNN recognition, we chose k to be five. The dimensionality of motion data is reduced to three. To evaluate the recognition rate using grasp trajectories, we compared our results with the recognition rate in the approach using final grasp poses. The grasp poses were extracted from the grasp trajectories in the training dataset. Similar to the dimensionality reduction we applied on grasp trajectories, the dimensionality of the 14-DOF hand poses were also reduced to three using LDA. LDA

[87] is a popular linear dimensionality reduction approach commonly applied for recognition purposes that projects data points into subspaces, maximizing linear class separability and minimizing the within-class variance.

To compare the results between our approach and the LDA approach, cross-validations were performed to assess the recognition rate for both known and unknown objects (new objects). For cross-validation of known objects, each motion trajectory of all 18 objects was left out from the training data set for validation while the remaining trajectories were used for training. For cross-validation of unknown objects, all trajectories of each object were used for validation, while the trajectories of remaining objects were used for training.

The results are reported in Table 2: for known objects, our approach is slightly better than LDA approach, while for unknown objects, our approach has 11% improvement over the LDA approach in the recognition rate. Detailed comparisons of the recognition rates of all 12 grasp types for unknown objects are presented in Figure 43. It can be observed that the proposed approach has higher recognition rates than the approach using static grasping poses for all grasp types.

Table 2 Mean absolute percentage error between estimated grasp poses and the best grasp poses.

Testing Objects	Our Approach	LDA Approach
Known Objects	96%	94%
Unknown Objects	85%	74%

In addition, we performed cross-validations on unknown users (new demonstrators). In each of the four validations, each of the four subjects was selected in order for evaluation, with the other three subjects' grasp motion data used for training. Due to the limitation of the data glove, which has a large variation in measuring joint angles across users because of large geometry variances between human hands, the recognition rate drops to 61%. This is still comparable with other work [53, 54, 55], even

though our experiments were tested with more grasp types. Our study in [3] found that the differences between several grasp types are not significant based on the hand poses or motion when considering the difference between human hands.

Figure 44 illustrates three irregular grasps of a box, a pair of pliers, and a gun. The first column is the human demonstration. Columns 2 and 3 compare 3D representation of 14D demonstrated motion trajectories and poses. The trajectory marked by black stars is the testing grasp.

In Figure 44a, the participant demonstrated a large-wrap-like grasp of a box, where the little finger closed more than the other fingers. Figure 44b shows that the trajectory of the demonstrated grasp was similar to the trajectories belonging to the large wrap grasp in the training dataset. The five nearest neighbors of this demonstration were all trajectories in the large wrap grasp type. Figure 44c shows that the nearest neighbors of the demonstrated poses were the four-finger precision and tripod grasp types.

The second row illustrates a grasp of a pair of pliers. The nearest trajectories belonged to small wrap (Figure 44e), and the nearest poses belonged to small wrap and lateral pinch (Figure 44f).

The third row is the example of grasping a gun. The user employed a medium-wrap-like grasp, but the index finger was put on the trigger. The trajectory of grasping a gun was between 4-finger precision and 3-finger precision (Figure 44g), and the nearest neighbor of the grasp pose is 3-finger precision (Figure 44h).

Fewer grasp types can be defined for the robotic hand when mapping from the human, because robotic hands usually have less DOFs and then less dexterity. The fewer number of grasp types would also improve recognition. Taking the Barrett hand model for example, we defined only five grasp types, much less than the human hand: power grasp, power sphere, precision grasp, precision sphere, and lateral pinch. Some grasp types can be grouped into one. For example, four-finger-thumb precision, three-finger-thumb precision and two-finger-thumb precision grasps can be grouped together as a precision grasp for a robotic model when the two fingers opposite to the thumb are adducted together. Table 3 shows the corresponding grasp types between a human and a Barrett hand.

The five grasp types of the Barrett hand decide how the search space of the optimization procedure can be reduced: whether the grasp is a power grasp or not determines whether the object should be in contact with the palm; and whether the grasp is a sphere grasp or not determines whether the search subspace of spread angle can be rejected. If it is not a sphere grasp, the dimensionality of the search space can be reduced to two by commanding the Barrett hand to adduct completely. Determination of the sphere type is necessary because, in the experiment, we found that a sphere grasp with a larger spread angle usually has a higher quality measure than a non-sphere grasp type without a spread angle, while [95] pointed out that humans tend to use a low spread pinch, which leads to a lower force-closure quality but a higher success rate.

Table 3 The dictionary of the corresponding grasp types between the human and the robotic hand.

Human Grasp Types	Robot Grasp Types
Large Wrap Medium Wrap Small Wrap	Power
Power Sphere	Power Sphere
Precision Sphere Tripod	Precision Sphere
Four-finger-thumb Precision Three-finger-thumb Precision Two-finger-thumb	Precision
Lateral Pinch	Lateral Pinch

For other robotic hand models, of course, the grasp types can be defined dependent on the level to which the robotic hand is simplified for the human hand. For example, the same grasp types can be defined to the Shadow hand as the human hand, for it has similar dexterity to the human hand.

6.4.2. Evaluation on Resulting Simulated Grasps

In the experiment, grasp type, as well as the thumb position and orientation, were obtained from the human demonstration. The thumb position relative to the object of the demonstration was captured by an OptiTrack Mocap system, with reflecting markers on the thumb and the object. The joint angles were captured by the 5DT dataglove.

In simulation, we set the frictional coefficient μ to be 0.9. The friction cone is approximated by an eight-sided pyramid. Thus, each contact point has eight unit wrench vectors. The computation of the grasp is expensive for a convex hull, so we set the step size to be five degrees for all the angles. The grasp quality measure we used for the optimization was the commonly used epsilon quality, defined as the radius of the largest wrench ball that is enclosed by the grasp wrench space [11]. It provides a way to quantify the force-closure property, which is the necessary condition for a stable grasp. Of course, other grasp quality measures can also be used, such as the one that considers task wrench space in the measure to emphasize the ability of the grasp to resist large force disturbance in some direction [96].

To evaluate the proposed approach, simulation results of the Barrett hand model and the Shadow hand model were evaluated to compare our own simulator with the well-known simulator GraspIt!. Twenty daily objects and manipulative tools, as listed in Table 4, were tested in the experiments, which comprise basic shapes such as cuboids, spheres and cones. These basic shapes form the majority of daily objects and manipulative tools. The top ranking grasp of every object, as compared in Figure 45 for the Barrett hand and Shadow hand, were selected by our approach and Graspit!.

To quantitatively evaluate each grasp in relation to the functionality and shape of the target object, we introduced human survey to examine each grasp, as we believe that humans are experts in grasping. Therefore, we relied on human participants to use their intuition and experience in choosing a grasp suitable to an object to evaluate the simulated grasps. Thirteen people participated and were asked to choose the grasps they would like the robot to use, considering the functionality and shape of the objects.

The resulting grasps from both simulators were randomly ordered, and the color of the object models and hand models were set the same to minimize bias and eliminate any clue from the appearances of the two simulators. The participants were instructed to select all the good grasps from all the grasping results generated by both approaches.

Figures 46 and 47 report the grasp choices by thirteen human participants for the Barrett hand model and Shadow hand model respectively. On average, for the Barrett hand model, 92.31% selected the grasps resulting from our approach, compared with 5.77% of GraspIt!; while for the Shadow hand model, 85.77% selected the grasps resulting from our approach, compared with 10.38% of GraspIt!. For a majority of the grasps, more people thought the grasps resulting from our approach are the ones they would prefer the robot to use. The exceptions are the banana, light bulb and mug in the Shadow hand evaluation. One reason is the static object assumption in most of the existing simulators [97]. In the real world, the object may be moved by the fingers after contact, but this property is not implemented in simulation. The grasp planner had trouble finding a grasp for the mug like humans, who tend to wrap their fingers around the handle. Although the resulting grasp by the planner has a high force-closure property, it is not very robust in the real world.

Table 4 Twenty objects evaluated in simulation.

No.	Object Name	No.	Object Name
1	Banana	11	Dust pan
2	Beer bottle	12	Frying pan
3	Square bowl	13	Screwdriver
4	Bucket	14	Telephone handset
5	Clothes iron	15	Bucket
6	Headphones	16	Light bulb
7	Hair brush	17	Cup
8	Hammer	18	Power drill
9	Kitchen Knife	19	Onion
10	Mug	20	Spoon

Based on the observation from the simulation and the human survey, without integrating grasp type and the thumb placement into the planning, GraspIt! tends to select an encompassing grasp for each object, while with the integration of the extracted strategy from human, the resulting grasps are more close to human-style grasps.

6.4.3. Experiment on Real Robotic Platform

We also tested the proposed approach with a real Barrett hand equipped on a 6-DOF FANUC LR MATE 200iC robotic arm. Ten daily objects were evaluated and the grasp success rates were reported, shown in Figure 48. Each grasping was executed for ten trials. The object was placed at the same predefined known position and orientation in the robot workspace every time, without any obstacles between the robot and the target object. The accurate geometric model of the object was also obtained. The robot first moved to the target object and grasped the object in a resulting pose from planning, then the robot lifted the target object. The position error was around 1 cm on average. We judged a grasp to be successful if the robot could hold the object stably. A grasp is regarded as a failure if the object is dropped or slides out of the hand.

As reported in Table 5, we achieved a success rate of 79% for the ten objects. For most of the time, our approach was able to lift the object without dropping it. Overall, the power grasp achieved a higher success rate than the precision grasp (the bowl, tape and onion). The failure grasps were mostly caused by position errors. For example, the pinch grasp for the spoon has a lowest success rate of 40% due to a considerable position error for the relatively small objects.

Although our experiment included precision grasps, which are usually more sensitive to position errors, results are still comparable with the success rate of 77% for GraspIt! reported in [95], which only tested power grasps. Moreover, by incorporating human strategies of the thumb placement and grasp type, the proposed approach found more human desired grasps.

Table 5 Success rate in real experiment.

Object Name	Grasp Success Rate
Beer bottle	100%
Headphones	100%
Mug	80%
Frying pan	90%
Bowl	70%
Cup	100%
Power drill	80%
Ring shape tape	70%
Onion	60%
Spoon	40%
Overall	79%

6.4.4. Studies on the Robustness to Uncertainty

It is a common problem that uncertainty exists in the perception of human demonstrations, as well as the pose estimation of the object. Noisy sensors also cause errors in perceptions of the shape and pose of an object. In our method, it is important to study how accurate the perception of human demonstrations and the object needs to be so that the grasp is not broken by uncertainty. We conducted experiments in simulation to examine how robust the resulting grasps of the proposed method are to resist small perception errors on object geometry and the relative thumb position to the target object. Five objects, including a sphere, a box, a cylinder, a torus and a cone, were tested in the experiment. We only tested precision grasp because perception error is of higher concern for precision grasp than power grasp.

The introduction of object geometry errors can be implemented by perturbing the object size. Because the result of every size error was the same for every execution in the simulation, each size error was executed once. Figure 49 shows the quality measures of the five object models with slightly changing sizes from 0 to 10%. Since a grasp is considered to be a stable grasp when the epsilon quality is

0.1 or greater, although there was some error in the grasp qualities, the stability of the grasp was not changed.

Thumb position relative to an object is easily shifted in simulation, thereby making it easy to simulate position errors. For example, in Figure 50, the thumb is shifted slightly to the left of the real position, with a visible position error of $20\ mm$. Figure 51 illustrates how the epsilon quality is changed at different thumb position errors from 0 to $10\ mm$. According to the results, the resulting grasp presented some robustness to small position errors.

6.5. Discussions

Precision grasps are vital in dexterous manipulation. Generally, people would first apply a precision grasp with only the fingertips in contact with an object when they pick it up. Then, they perform in-hand manipulation to get better closure around the object for the subsequent manipulation.

Figure 52 shows the quality measure of a precision grasp optimization involving grasping an ashtray with respect to relative thumb pitch and thumb flexion angle. The ridge shape of the quality measure implies that the planning results in more than one optimal precision grasp quality, indicating a set of compositions of thumb pitch and thumb flexion angle. The postures of different hand compositions are shown in Figure 53. The resulting grasps indicate a rolling manipulation, where the contact points remain the same but the hand postures vary, demonstrating a manipulation flexibility of the precision grasp resulting from the proposed method.

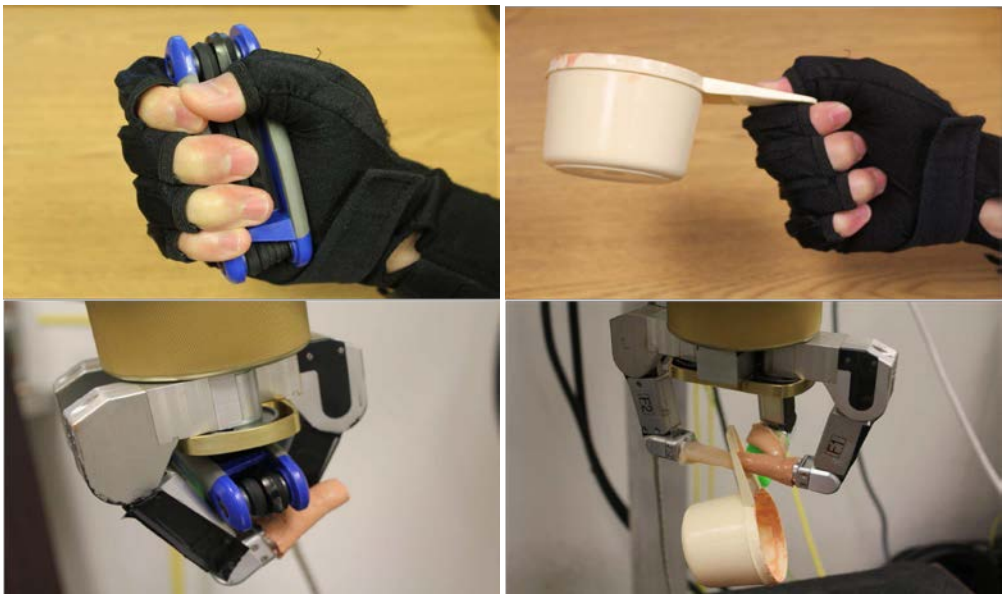


Figure 32. The corresponding small wrap and lateral pinch of the robotic hand and the human hand. They look similar to a human grasp but are different for a robotic grasp. Left: Small wrap grasps for a human hand (top) and a robotic hand (bottom). Right: Lateral pinch grasps for a human hand (top) and a robotic hand (bottom).

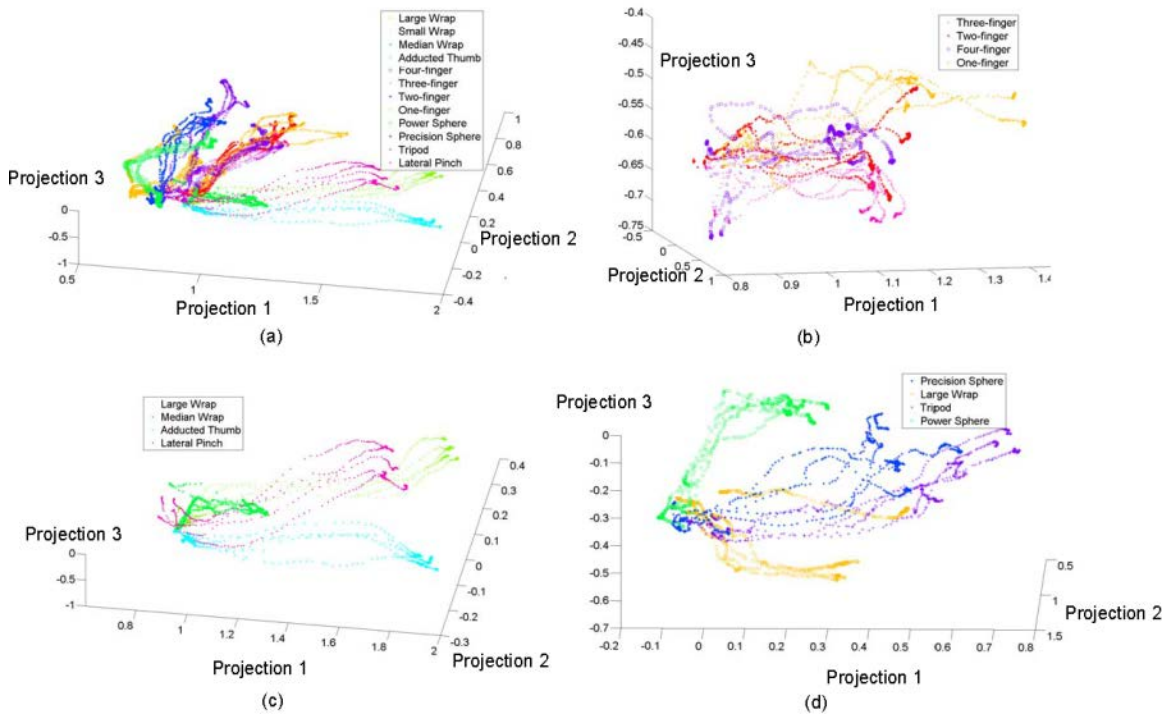


Figure 33. Three dimensional visualization of the high dimensional grasp motion data using LPP. Figures (b-d) are subfigures of Figure (a) for better visualization.

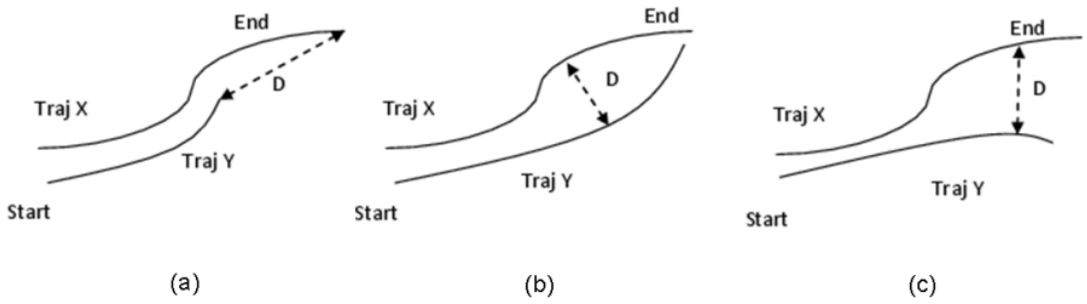


Figure 34. Three cases of Hausdorff distance between two grasp types. (a): Case 1, trajectory Y is a part of trajectory X; (b): Case 2, trajectory X and Y meet at the end but differ on the way; (c): Case 3, general case, where trajectories X and Y go further away until the end.

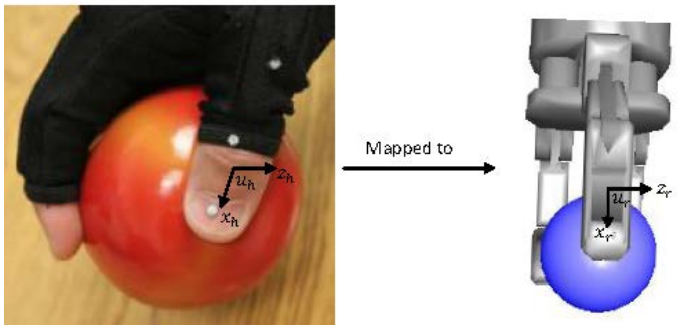


Figure 35. Translation of thumb placement from human demonstration to the robotic hand.

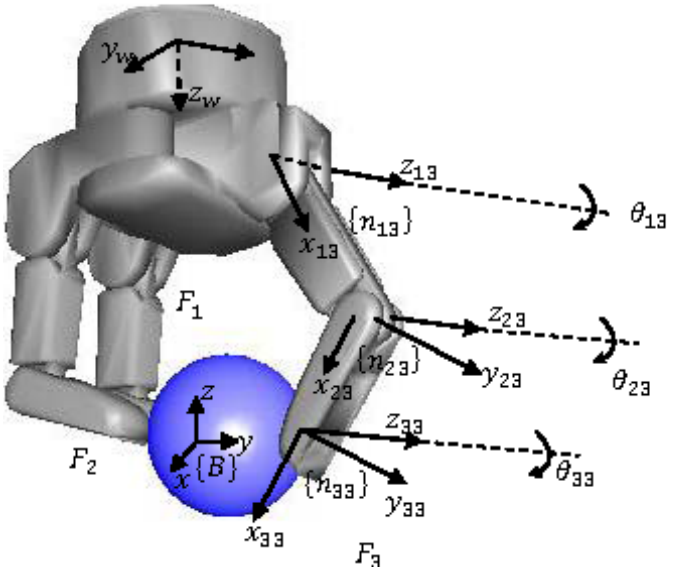
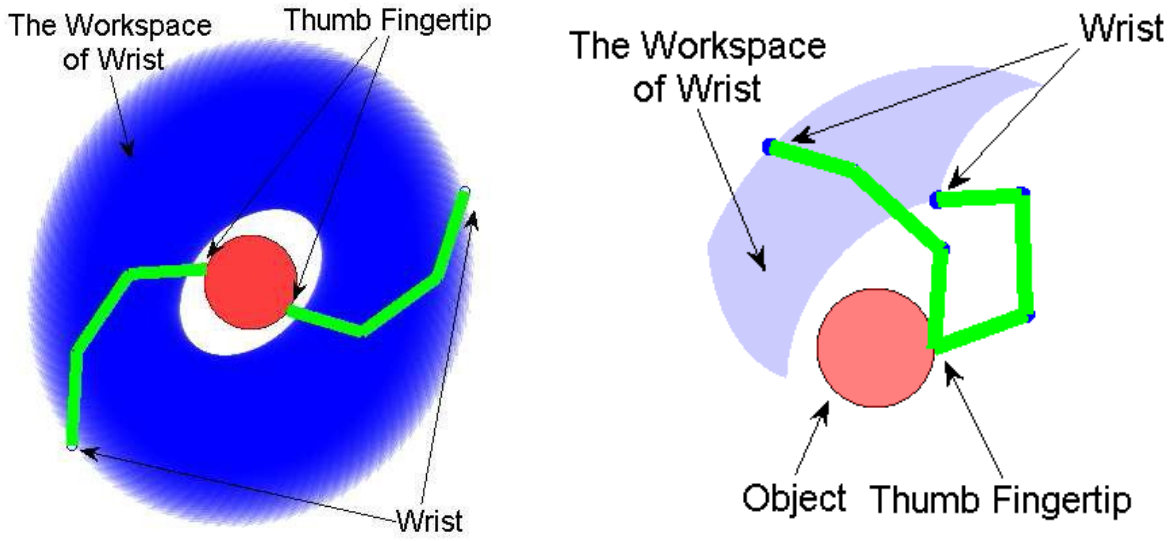


Figure 36. Barrett hand kinematics. $\boldsymbol{F1}$, $\boldsymbol{F2}$ and $\boldsymbol{F3}$are the three fingers, with $\boldsymbol{F3}$ being the thumb finger; frame $\{\boldsymbol{n_w}\}$ is attached to the wrist; frame $\{\boldsymbol{n_w}\}$ with axes $\{\boldsymbol{x_{i3}}, \boldsymbol{y_{i3}}, \boldsymbol{z_{i3}}\}$ is attached to each joint of $\boldsymbol{F3}$, where $\boldsymbol{i = 1, 2, 3}$; $\boldsymbol{\theta_{i3}}$ denotes the joint displacement, rotating around rotational axis $\boldsymbol{Z_{i3}}$.



(a) (b)

Figure 37. Workspace of the Barrett hand. (a) without constraint, and (b) with thumb constraint extracted from demonstration.

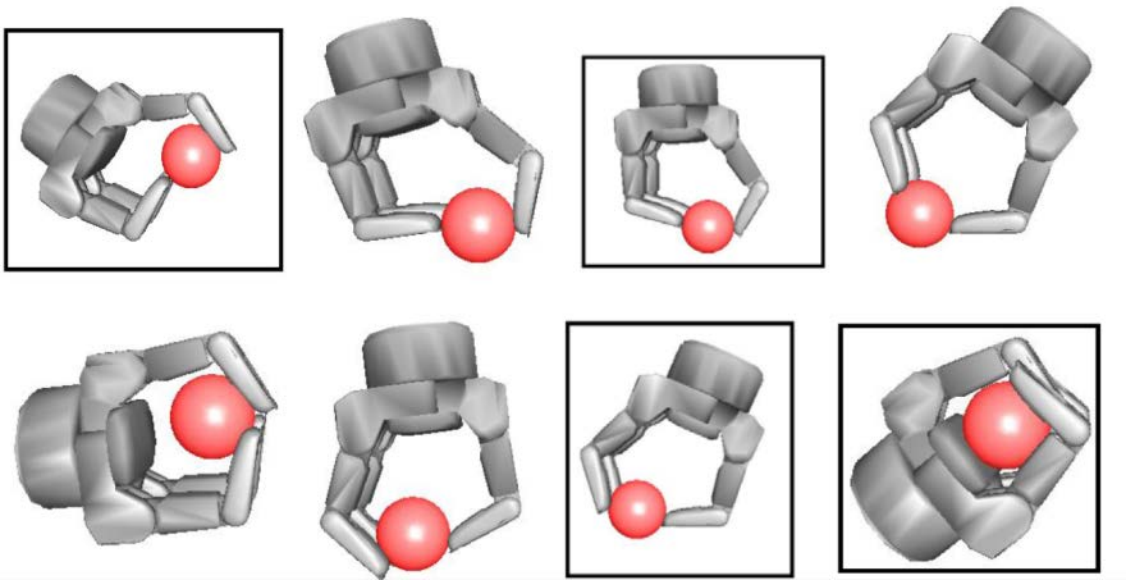


Figure 38. An example of hand posture snapshots during a searching procedure. Postures bordered by a black box are grasps with an epsilon quality larger than 0.1.

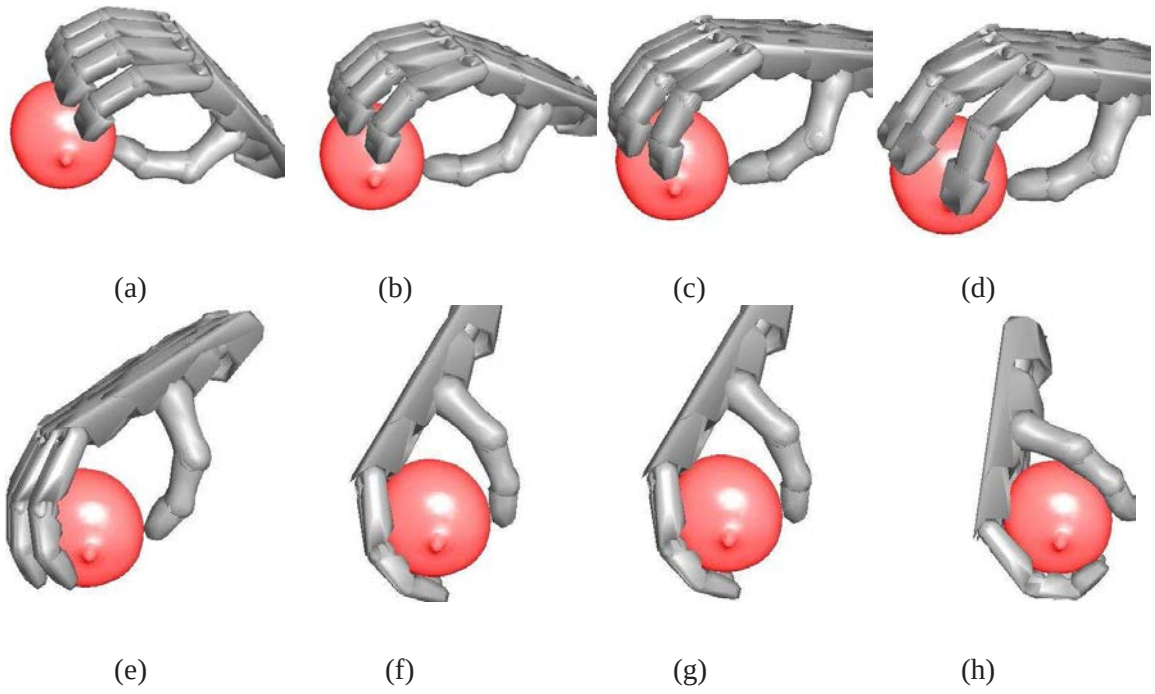


Figure 39. A procedure of the Shadow hand searching for a power sphere grasp. Figure (d) is a precision sphere grasp, which is rejected because it is not a desired grasp type; Figure (h) is a desired power sphere grasp.

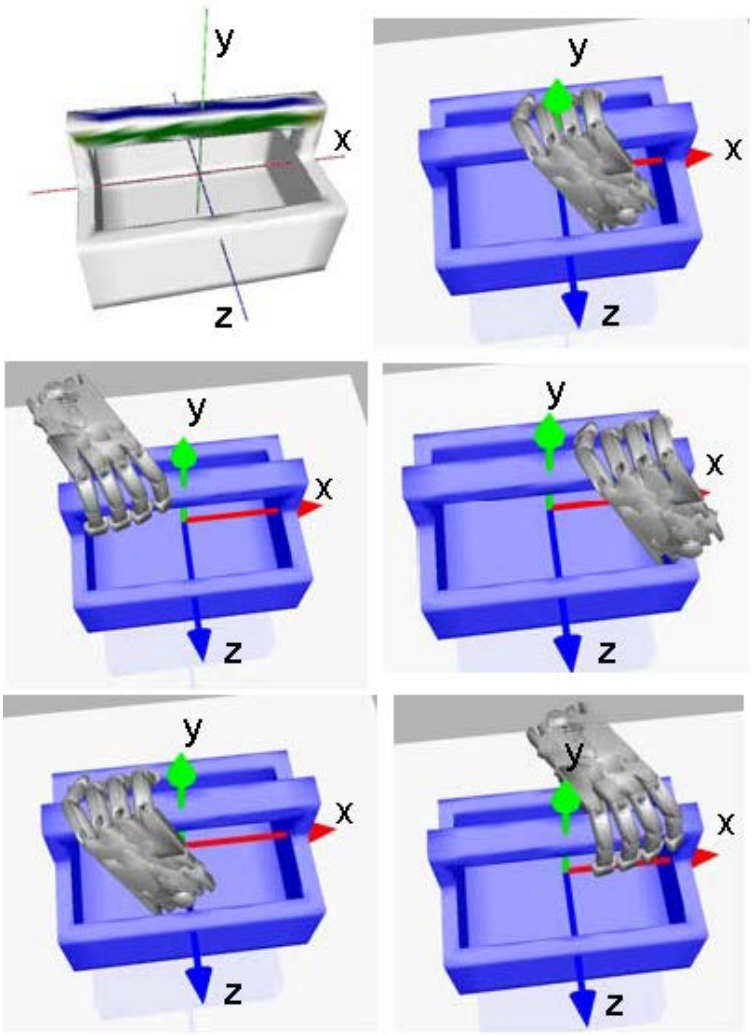


Figure 40. Illustration of a searching procedure constrained by thumb place and direction. The colored area in the first figure is the area where the thumb is allowed to be placed. Thumb placement in the green-colored area can be pointed only to axis y, while thumb placement in the blue-colored area can be pointed only to axis z.

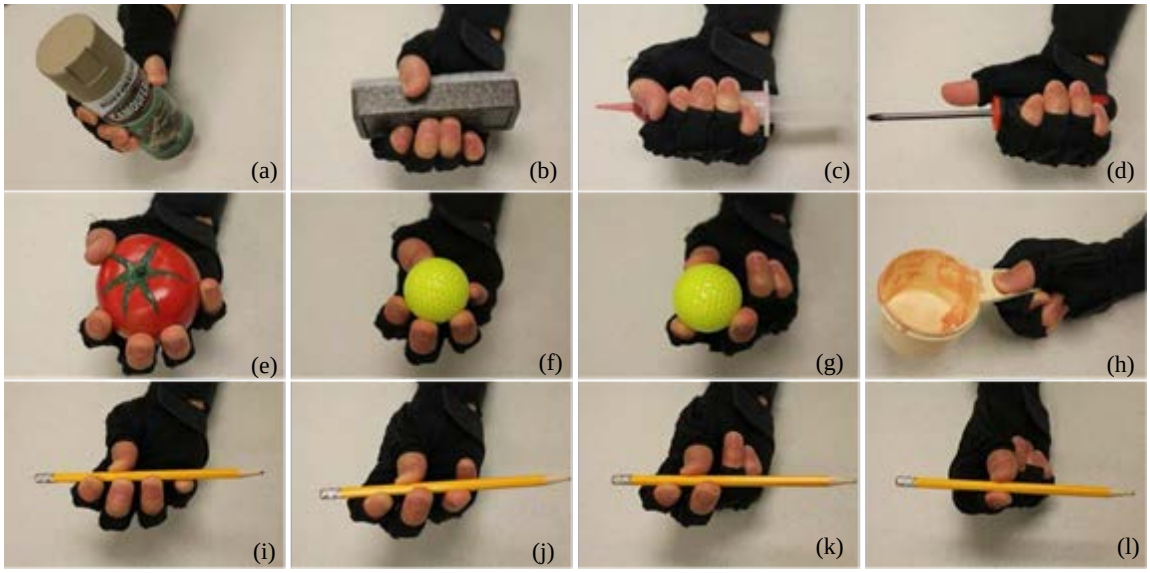


Figure 41. Twelve human grasp types used for training. (a) Large wrap; (b) Medium wrap; (c) Small wrap; (d) Adducted thumb; (e) Power sphere; (f) Precision sphere; (g) Tripod; (h) Lateral pinch; (i) Four-finger- thumb precision; (j) Three-finger-thumb precision; (k) Two-finger-thumb precision; (l) One-finger-thumb precision.

Figure 42. Daily grasping objects used in the experiment.

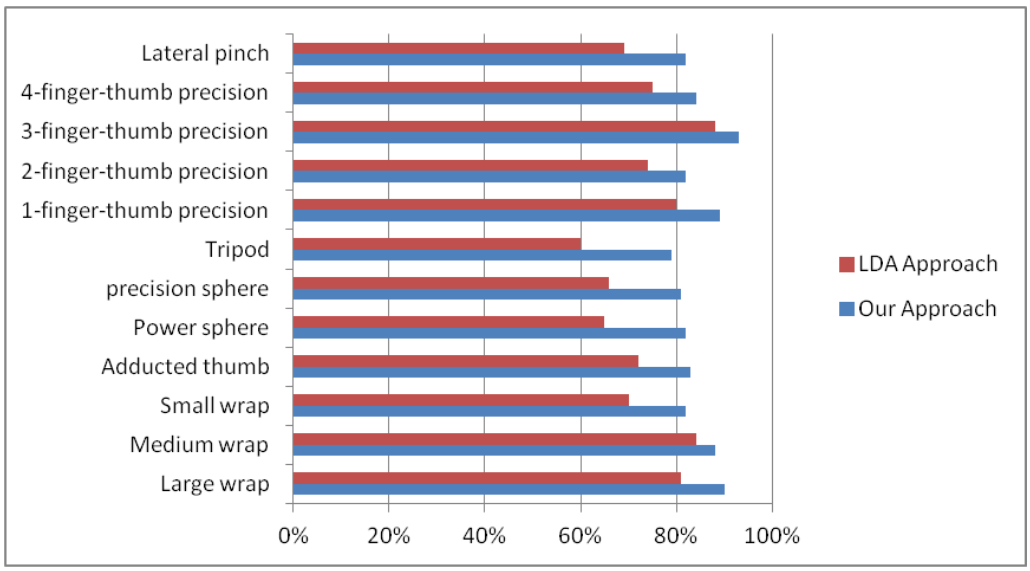


Figure 43. Recognition rates of our approach and LDA approach for unknown objects.

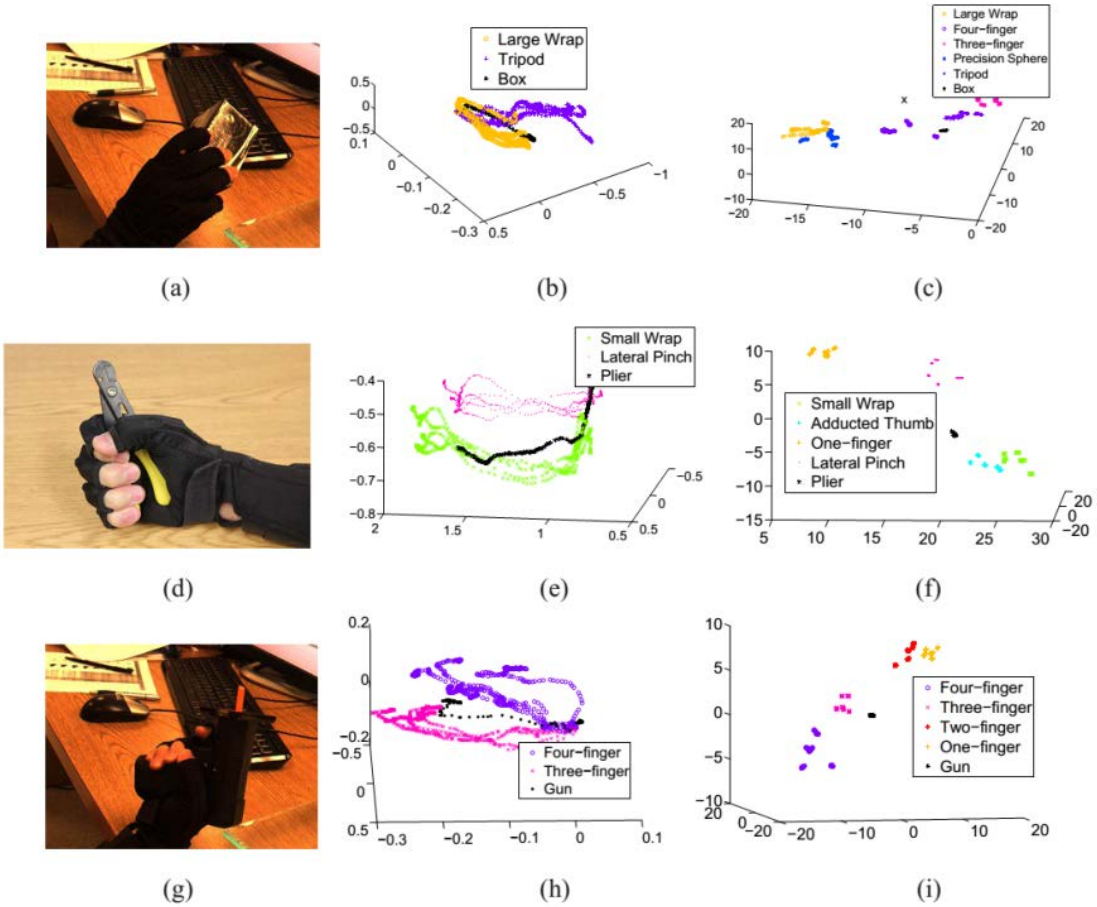
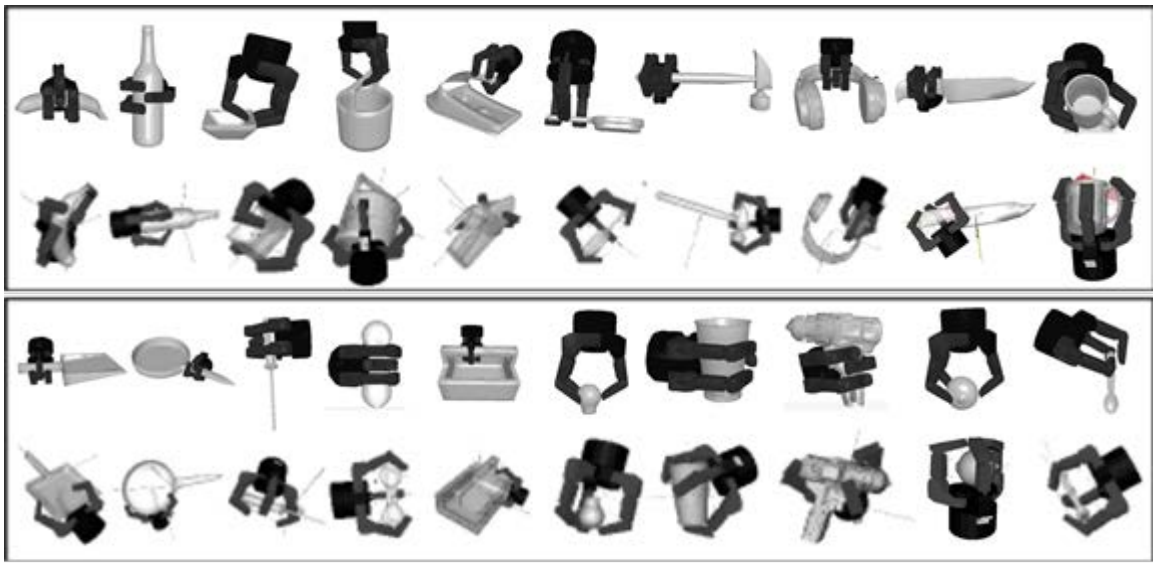
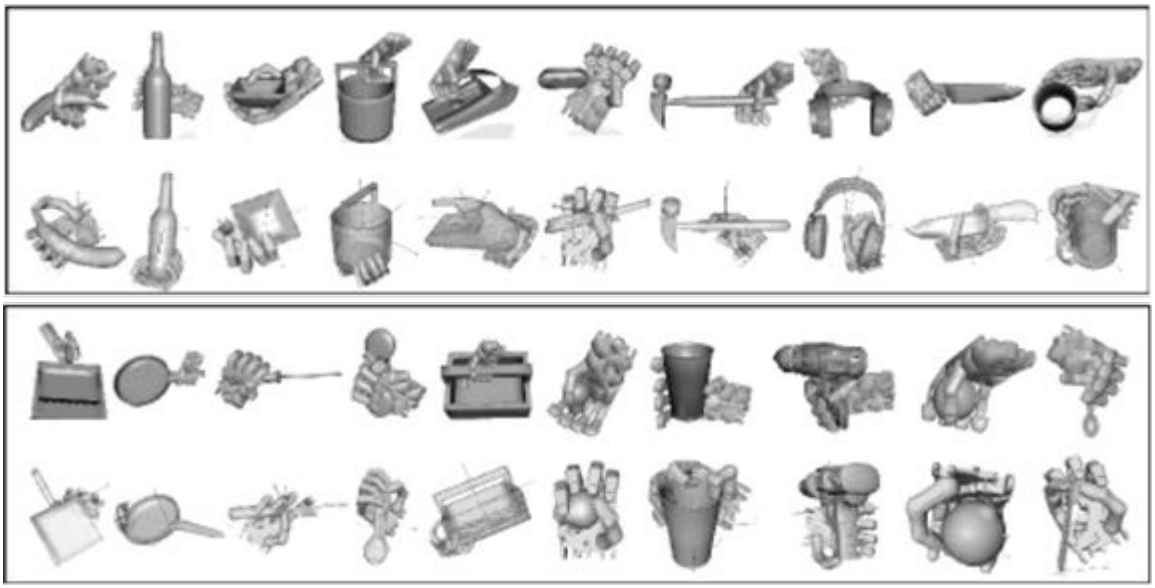


Figure 44. Three grasp examples. Column 1, human demonstration; Column 2, 3-D representation of 14-D hand motion trajectories using LPP; Column 3, 3D representation of 14D static hand pose using LDA.



(a)



(b)

Figure 45. Comparison between the top ranking grasps selected by our approach and GraspIt!. Figure (a): Result of the Barrett hand model; (b): results of the Shadow hand model.

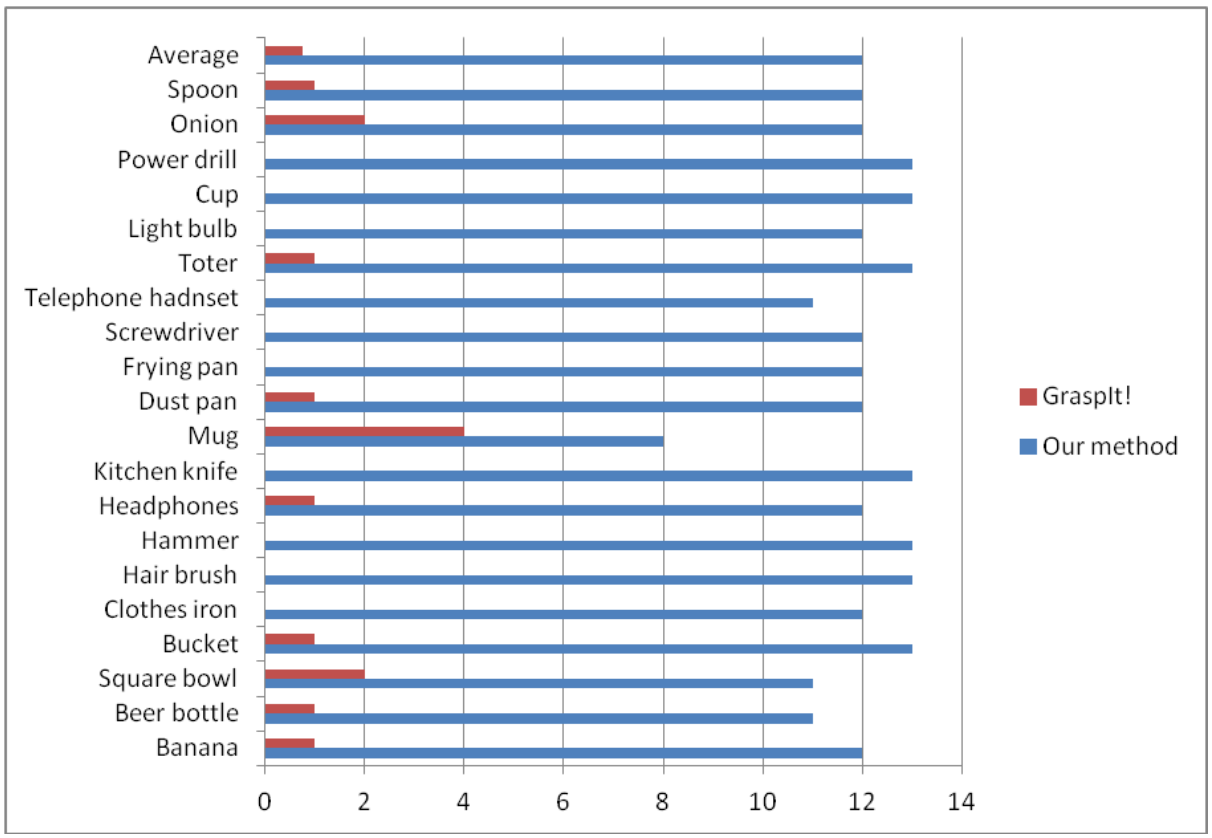


Figure 46. This figure reports the choice of thirteen human participants in their preferred grasps for the Barrett hand model, concerning the functionality and shape of the target object. Multiple choices or no choice can be made. On average, 92.31% considered the grasps selected by our approach as close to their intuitive ones, compared with 5.77% of GraspIt!.

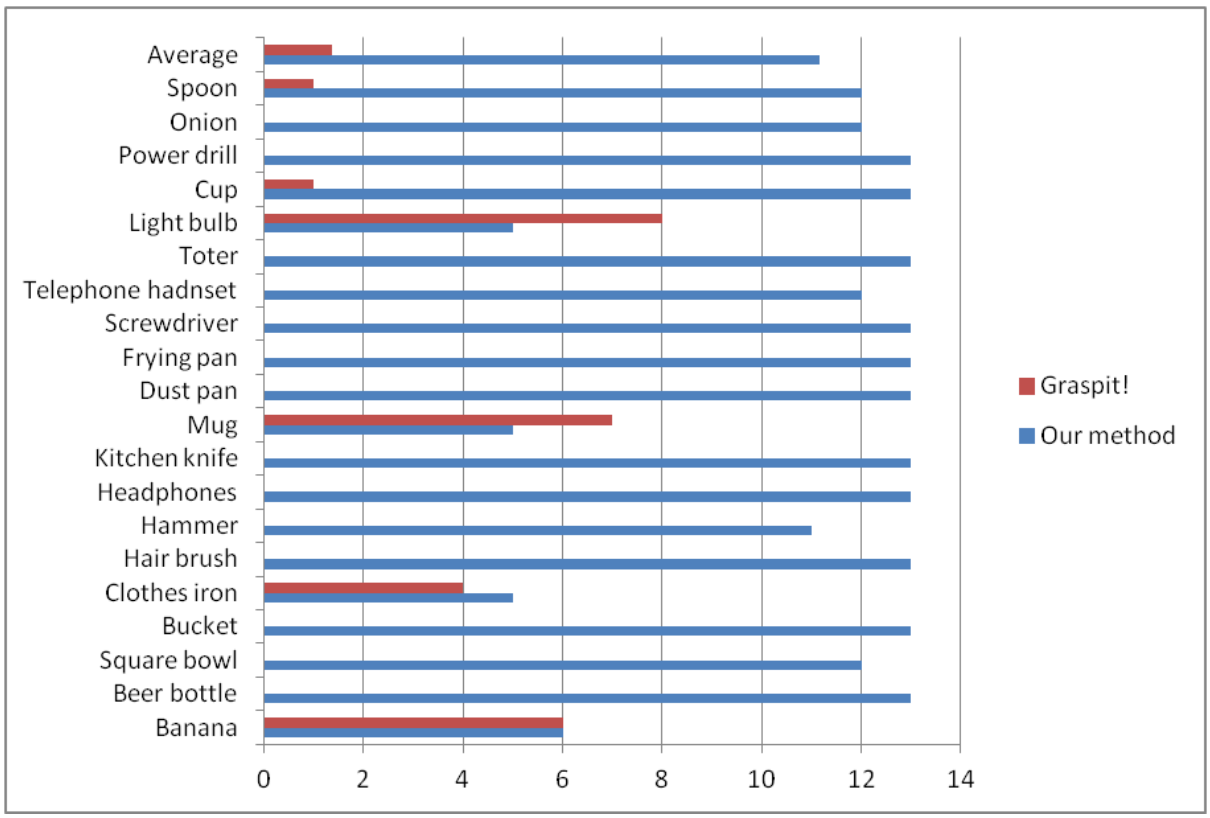


Figure 47. This figure reports the choice of thirteen human participants in their preferred grasps for the Shadow hand model, concerning the functionality and shape of the target object. Multiple choices or no choice can be made. On average, 85.77% selected the grasps resulting from our approach, compared with 10.38% of GraspIt!.

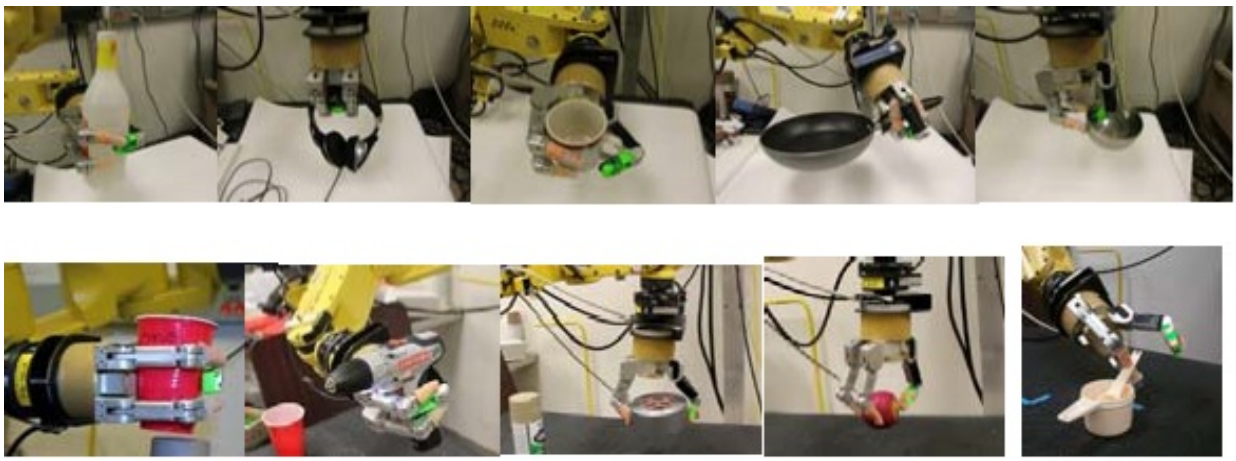


Figure 48. Experiment on real robotic platform.

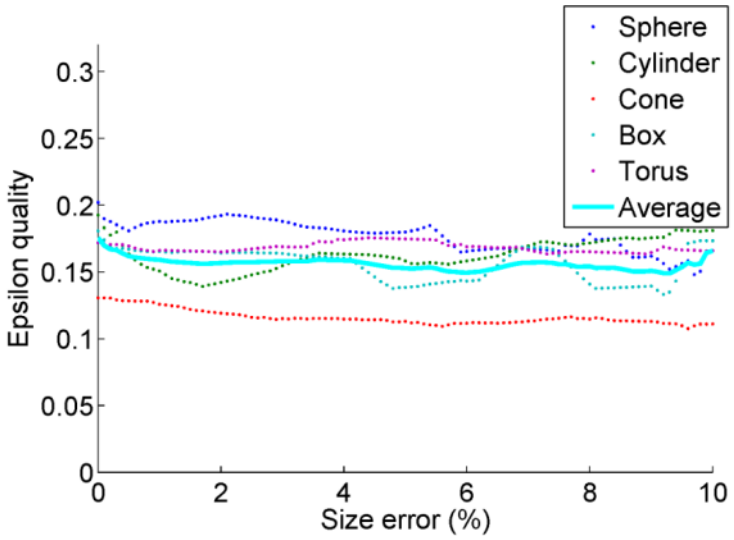
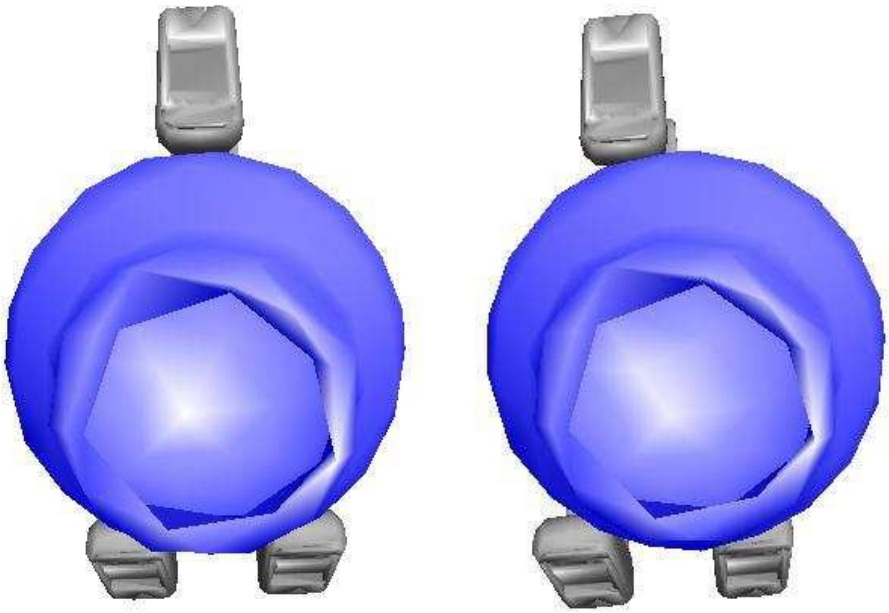


Figure 49. The epsilon quality measures at different object size errors.



(a) (b)

Figure 50. A visible thumb position error of 20mm in simulation. (a) Original thumb position; (b) thumb position with a 20mm error.

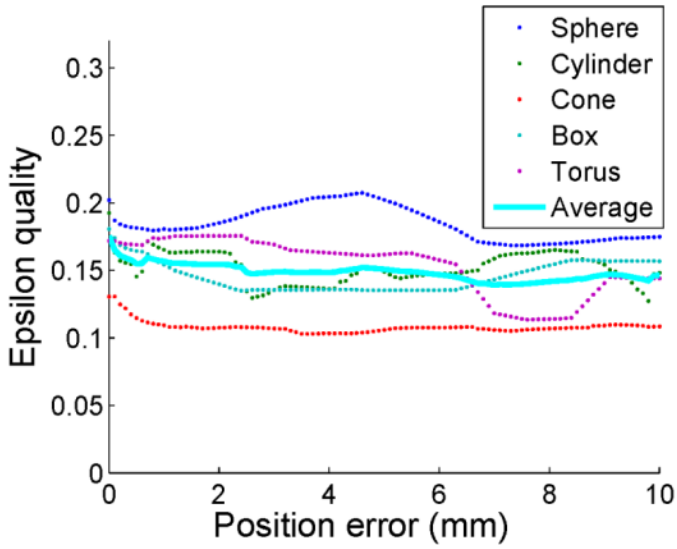
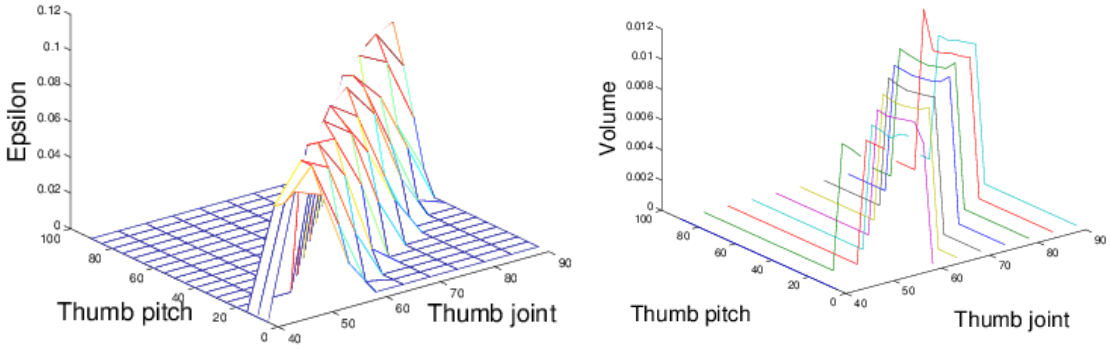


Figure 51. The epsilon quality measures at different thumb position errors.



(a) (b)

Figure 52. The grasp quality measures of a precision grasp vs. thumb pitch relative to the object and thumb joint angles. (a) Epsilon grasp quality; (b) Volume grasp quality.

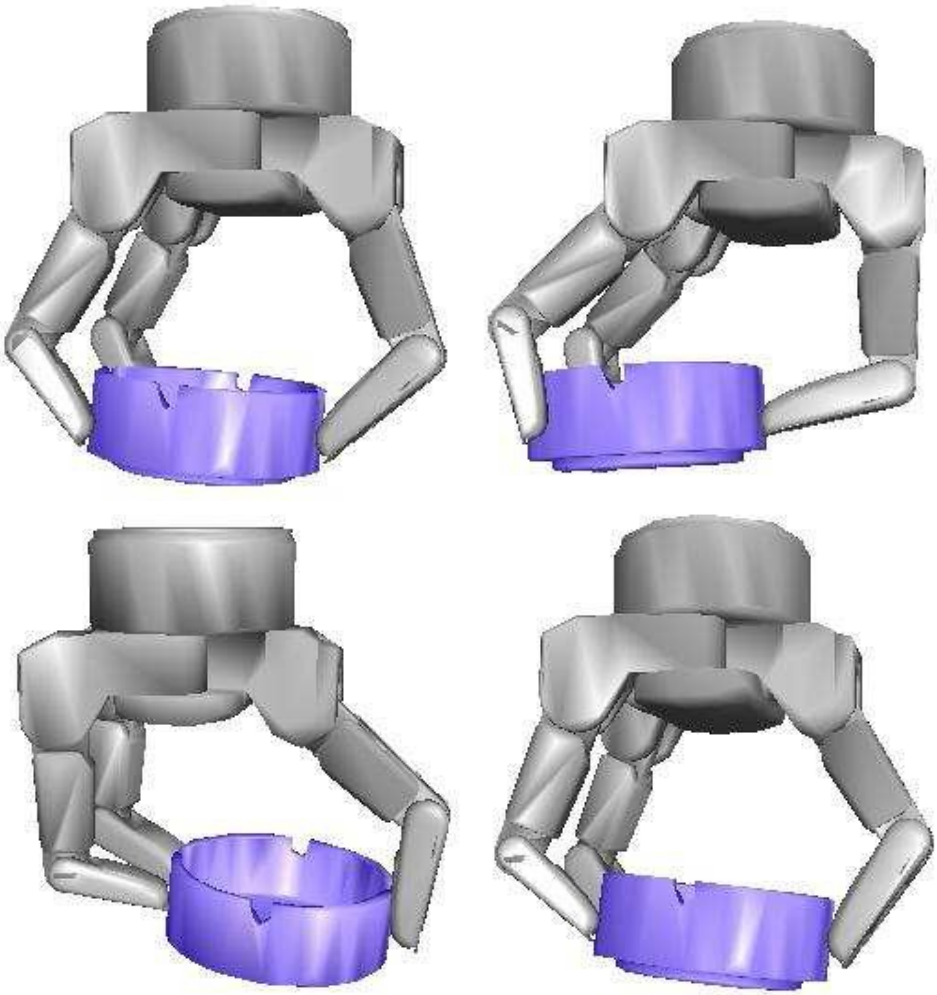


Figure 53. Four different hand postures along the ridge of the quality measure surface.

Chapter 7

Conclusions and Future Work

7.1. Discussion

In the context of this dissertation, the problem of deriving an optimal grasp for an object associated with a task to manipulate the object was addressed. For task-oriented grasp planning, manipulation tasks are known to be difficult to model. In this study, a manipulation task was modeled with the force and motion requirements, which should be efficiently satisfied by the manipulator given a grasp. The grasp should maintain a firm grip and withstand interactive wrench of the task; and the grasp should enable the manipulator to carry out the task most efficiently with minimal motion effort.

The data of task wrench cluster (TWC) and the data of task motion sequence (TMS) were captured from the human demonstration, instead of empirically approximating the task. The task-oriented grasp quality metrics quantify how well a grasp can meet the task requirement, given the wrench and motion data. Studies on the task wrench cluster indicate that the task wrench is not evenly-distributed. Instead, it is possible that interactive wrenches in some directions occur more frequently than the other areas, even if they may be smaller than wrenches that occur less frequently. In favor of grasps that are able to apply frequently-occurring forces, the task wrench coverage criterion measures the ratio of disturbance a grasp covers.

To reduce the computational complexity of the search in a high-dimensional robotic hand configuration space, as well as to avoid a correspondence problem, the human grasp strategies of grasp type and thumb placement were used to constrain the grasp. The candidate grasps are computed from a set of given thumb placements and directions, and confined by the desired grasp type. Grasp type,

thumb placement and direction are basic grasp descriptions existing in almost all grasps. They can also be commonly used for different robot models, facilitating the robot to generate a user desired grasp.

Grasp type and thumb placement provide partial constraints to hand postures and wrist positions and orientations. They cannot ensure the grasp is stable, but leave enough room and flexibility to accommodate the differences between a human hand and a robotic hand. Grasp optimization then searches for a stable grasp adaptable to a specific object shape given a robotic hand model.

The approach has been validated in simulations with a Barrett hand and a Shadow hand. Both the demonstration and the task model are independent of hand models, so they can be used for other robotic hands. We also demonstrated how robust the approach was to perception uncertainties.

The presented approach was also evaluated with a real robotic system to compare with the non-task- specific automatic grasp planning. Results verify that success was consistently achieved with the proposed disturbance-based quality metric.

The force and motion requirement are basic task descriptions that commonly exist in every task. Also, grasp type and thumb constraints are universal grasp strategies for almost all robotic hands. Nevertheless, additional task-based grasp constraints may be required for a specific task. For example, in a task of pouring water with a cup, a hand should not block the open surface of the cup; in a task of using a power drill or a firearm, a finger should be put on the trigger. These grasp features are dependent on tasks.

The static property of virtual objects in simulations also limits the grasp planning, which is the limitation of most existing grasp simulators. In simulations, the static object cannot be moved by the fingers after contact. Then, the simulated stable grasp may be fragile in reality, such as the mug example in Chapter 6. In the future, object dynamics property needs to be included in the simulator.

7.2. Future Directions

In our current research, we modeled the grasp only by contact locations without considering hand con- figurations. This raises the problem that the contact force may not be applicable by finger joints if they are in “bad” configurations. Some existing work has proposed a grasp quality metric from hand

configurations, such as manipulability – the ability of the manipulator to impart arbitrary motions at the end-effector [18]. In the future, grasp quality measures considering hand configurations can be combined with our proposed quality measure as a global grasp quality measure to give a more complete evaluation of a grasp. A global grasp quality measure combining contact forces, arm motion, and manipulability could result in good contact points, wrist position and orientation, and hand configurations. With good contact points, the grasp is not broken by the interactive wrenches from the environment; with appropriate wrist position and orientation, the robot can efficiently execute the manipulation; and with proper hand configurations, the joint torques can be easily transferred from actuators to fingertips.

When computing the task motion effort in our current experiments, we only tested the non-redundant robotic arms that have 6 DOFs, where there are only a finite number of solutions to the path tracking problem. For redundant robotic arms that have more than 6 DOFs, since there may be an infinite number of solutions, a more general approach is necessary to find the optimal solution given a grasp. Therefore, a future direction is to combine motion planning together with grasp planning. A further investigation in this direction will be crucial for our work.

We did not consider grasp planning for novel objects, which has been explored by other researchers. Example work includes [59, 60]. The proposed approach can be easily combined with other algorithms to adapt to novel objects. In the future, potential research can be conducted toward this direction. Grasp planning can be performed by matching a similar shape in the database of objects with a labeled thumb contact point and task-specific grasp type.

The hammer example in simulation implies that the resulting robotic grasps may be different from intuitive grasps of humans, who consider a combination of hand and arm motions as well as force required by a task. The existing research, including ours, only considers that the object wrenches are achieved by a grasp, but does not consider they can be achieved by arm motions. The hand and arm are treated separately in grasp analysis. Therefore, including arm motion that can also affect the object wrenches can be a direction of future work in grasp analysis.

The current work assumes the manipulation is performed with one grasp. In many situations, the manipulation cannot be fulfilled by only one grasp, but requires transition from one grasp to another grasp instead. Moreover, for multi-finger robotic hands other than simple grippers, in-hand manipulation is also possible. Therefore, another research direction is using our approach for manipulation planning. The manipulation task can be segmented into sub-tasks, and then the grasp for each sub-task as well as in-hand manipulation to transfer grasps will be planned for the task.

References

[1] J. R. Napier. Evolution of the human hand. In *Proceedings. Royal Inst. Great Britain*, volume 40, pages 544–557. Wiley Online Library, 1965.

[2] M.R. Cutkosky. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *Robotics and Automation, IEEE Transactions on*, 5(3):269–279, 1989.

[3] W. Dai, Y. Sun, and X. Qian. Functional analysis of grasping motion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

[4] D. A. Rosenbaum, R. J. Meulenbroek, J. Vaughan, and C. Jansen. Posture-based motion planning: applications to grasping. *Psychological review*, 108(4):709, 2001.

[5] M. Dornay, Y. Uno, M. Kawato, and R. Suzuki. Minimum muscle-tension change trajectories predicted by using a 17-muscle model of the monkey's arm. *Journal of Motor Behavior*, 28(2):83–100, 1996.

[6] T. Flash and N. Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *The journal of Neuroscience*, 5(7):1688–1703, 1985.

[7] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement. *Biological cybernetics*, 61(2):89–101, 1989.

[8] A. Gasparetto and V. Zanotto. A technique for time-jerk optimal planning of robot trajectories. *Robotics and Computer-Integrated Manufacturing*, 24(3):415–426, 2008.

[9] K. J. Kyriakopoulos and G. N. Saridis. Minimum jerk path generation. In *IEEE International Conference on Robotics and Automation*, pages 364–369. IEEE, 1988.

[10] D. Kirkpatrick, B. Mishra, and C. K. Yap. Quantitative steinitz's theorems with applications to multi- fingered grasping. *Discrete & Computational Geometry*, 7(1):295–318, 1992.

[11] C. Ferrari and J. Canny. Planning optimal grasps. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2290–2295, May 1992.

[12] N. S. Pollard. Parallel methods for synthesizing whole-hand grasps from generalized prototypes. *Ph.D Dissertation*, 1994.

[13] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen. Automatic grasp planning using shape primitives. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1824–1829, Sep 2003.

[14] K. Hsiao, M. Ciocarlie, P. Brook, and W. Garage. Bayesian grasp planning. In *ICRA Workshop on Mobile Manipulation: Integrating Perception and Manipulation*, 2011.

[15] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 348–353. IEEE, 2000.

[16] O. Brock, J. Kuffner, and J. Xiao. Motion for manipulation tasks. In Bruno Siciliano and Oussama Khatib, editors, *Springer Handbook of Robotics*, pages 615–645. Springer Berlin Heidelberg, 2008.

[17] Z. Li and S. S. Sastry. Task-oriented optimal grasping by multifingered robot hands. *IEEE Journal of Robotics and Automation*, 4(1):32–44, feb 1988.

[18] T. Yoshikawa. Manipulability of robotic mechanisms. *The international journal of Robotics Research*, 4(2):3–9, 1985.

[19] A. Sahbani, S. El-Khoury, and P. Bidaud. An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336, 2012.

[20] C. Borst, M. Fischer, and G. Hirzinger. Grasp planning: how to choose a suitable task wrench space. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 319–325, May 2004.

[21] L. Han, J. C. Trinkle, and Z. X. Li. Grasp analysis as linear matrix inequality problems. *IEEE Transactions on Robotics and Automation*, 16(6):663–674, 2000.

[22] R. Haschke, J. J. Steil, I. Steuwer, and H. Ritter. Task-oriented quality measures for dextrous grasping. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 689–694, June 2005.

[23] V. D. Nguyen. Constructing force-closure grasps. *The International Journal of Robotics Research*, 7(3):3–16, 1988.

[24] J. Ponce and B. Faverjon. On computing three-finger force-closure grasps of polygonal objects. *IEEE Transactions on Robotics and Automation*, 11(6):868–881, 1995.

[25] X. Zhu and J. Wang. Synthesis of force-closure grasps on 3-d objects based on the q distance. *IEEE Transactions on Robotics and Automation*, 19(4):669–679, 2003.

[26] J. Liu, G. Xu, X. Wang, and Z. Li. On quality functions for grasp synthesis, fixture planning, and coordinated manipulation. *IEEE Transactions on Automation Science and Engineering*, 1(2):146–162, 2004.

[27] M. A. Roa and R. Suárez. Finding locally optimum force-closure grasps. *Robotics and Computer-Integrated Manufacturing*, 25(3):536–544, 2009.

[28] C. Rosales, L. Ros, J. M. Porta, and R. Suárez. Synthesizing grasp configurations with specified contact regions. *The International Journal of Robotics Research*, 30(4):431–443, 2011.

[29] S. B. Kang and K. Ikeuchi. Toward automatic robot instruction from perception-mapping human grasps to manipulator grasps. *IEEE Transactions on Robotics and Automation*, 13(1):81–95, 1997.

[30] R. Zoliner, M. Pardowitz, S. Knoop, and R. Dillmann. Towards cognitive robots: Building hierarchical task representations of manipulations from human demonstration. In *IEEE International Conference on Robotics and Automation,*, pages 1535–1540. IEEE, 2005.

[31] J. Tegin, S. Ekvall, D. Kragic, J. Wikander, and B. Iliev. Demonstration-based learning and control for automatic grasping. *Intelligent Service Robotics*, 2(1):23–30, 2009.

[32] J. Romero, H. Kjellstrom, and D. Kragic. Modeling and evaluation of human-to-robot mapping of grasps. In *International Conference on Advanced Robotics*, pages 1–6. IEEE, 2009.

[33] JW Soto Martell and Giuseppina Gini. Robotic hands: Design review and proposal of new design process. *World Academy of Science, Engineering and Technology*, 26:85–90, 2007.

[34] L. Biagiotti, F. Lotti, C. Melchiorri, and G. Vassura. How far is the human hand? a review on anthropo- morphic robotic end-effectors. 2004.

[35] K. B. Shimoga. Robot grasp synthesis algorithms: A survey. *The International Journal of Robotics Research*, 15(3):230–266, 1996.

[36] M. Cutkosky and R. D. Howe. Human grasp choice and robotic grasp analysis. In *Dextrous robot hands*, pages 5–31. Springer, 1990.

[37] J. Bohg, A. Morales, T. Asfour, and D. Kragic. Data-driven grasp synthesis - a survey. *CoRR*, abs/1309.2660, 2013.

[38] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[39] C. A. Klein and B. E. Blaho. Dexterity measures for the design and control of kinematically redundant manipulators. *The International Journal of Robotics Research*, 6(2):72–83, 1987.

[40] S. L. Chiu. Task compatibility of manipulator postures. *The International Journal of Robotics Research*, 7(5):13–21, 1988.

[41] A. M. Okamura, N. Smaby, and M. R Cutkosky. An overview of dexterous manipulation. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 255–262. IEEE, 2000.

[42] R. Suárez, J. Cornella, and M.RGarzón. *Grasp quality measures*. Institut d'Organització i Control de Sistemes Industrials, 2006.

[43] M. Prats, P. J Sanz, and A. P. D. Pobil. Task-oriented grasping using hand preshapes and task frames. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 1794–1799. IEEE, 2007.

[44] H. Kruger and A. F van der Stappen. Partial closure grasps: Metrics and computation. In *IEEE International Conference on Robotics and Automation*, pages 5024–5030. IEEE, 2011.

[45] H. Kruger, E. Rimon, and A. F. van der Stappen. Local force closure. In *IEEE International Conference on Robotics and Automation*, pages 4176–4182. IEEE, 2012.

[46] H. Kruger and A. F. van der Stappen. Independent contact regions for local force closure grasps. In *IEEE International Conference on Robotics and Automation*, pages 1588–1594, May 2013.

[47] A. Billard, S. Calinon, R. Dillmann, and S. Schaal. Robot programming by demonstration. In *Hand- book of Robotics*. MIT Press, 2008.

[48] J. Romero, H. Kjellstrm, and D. Kragic. Human-to-robot mapping of grasps. In *IEEE/RSJ Interna- tional Conference on Intelligent Robots and Systems, WS on Grasp and Task Learning by Imitation*, 2008.

[49] M. Hueser and J. Zhang. Visual and contact-free imitation learning of demonstrated grasping skills with adaptive environment modelling. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, WS on Grasp and Task Learning by Imitation*, 2008.

[50] M. Fischer, P. van der Smagt, and G. Hirzinger. Learning techniques in a dataglove based telemanipulation system for the dlr hand. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1603–1608. IEEE, 1998.

[51] A. Peer, S. Einenkel, and M. Buss. Multi-fingered telemanipulation-mapping of a human hand to a three finger gripper. In *The 17th IEEE International Symposium on Robot and Human Interactive Communication*, pages 465–470. IEEE, 2008.

[52] G. Gioioso, G. Salvietti, M. Malvezzi, and D. Prattichizzo. Mapping synergies from human to robotic hands with dissimilar kinematics: an object based approach. In *IEEE International Conference on Robotics and Automation*, 2011.

[53] S. Ekvall and D. Kragic. Grasp recognition for programming by demonstration. In *IEEE International Conference on Robotics and Automation*, pages 748–753. IEEE, 2005.

[54] J. Aleotti and S. Caselli. Grasp recognition in virtual reality for robot pregrasp planning by demonstration. In *IEEE International Conference on Robotics and Automation*, pages 2801–2806. IEEE, 2006.

[55] Guido Heumer, H Ben Amor, Matthias Weber, and Bernhard Jung. Grasp recognition with uncalibrated data gloves-a comparison of classification methods. In *Virtual Reality Conference, 2007. VR'07. IEEE*, pages 19–26. IEEE, 2007.

[56] M.R. Cutkosky. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *Robotics and Automation, IEEE Transactions on*, 5(3):269–279, 1989.

[57] Q. V. Le, D. Kamm, A. F. Kara, and A. Y. Ng. Learning to grasp objects with multiple contact points. In *IEEE International Conference on Robotics and Automation*, pages 5062–5069. IEEE, 2010.

[58] N. S. Pollard. Closure and quality equivalence for efficient synthesis of grasps from examples. *The International Journal of Robotics Research*, 23(6):595–613, 2004.

[59] 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.

[60] L. Ying, J. L. Fu, and N. S. Pollard. Data-driven grasp synthesis using shape matching and task-based pruning. *Visualization and Computer Graphics, IEEE Transactions on*, 13(4):732 –747, july-aug. 2007.

[61] J. Kim. *Example-based grasp adaptation*. PhD dissertation, Massachusetts Institute of Technology, 2007.

[62] H. Ben Amor, O. Kroemer, U. Hillenbrand, G. Neumann, and J. Peters. Generalization of human grasping for multi-fingered robot hands. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2043–2050. IEEE, 2012.

[63] K. Hsiao and T. Lozano-Perez. Imitation learning of whole-body grasps. In *Intelligent Robots and Systems, IEEE/RSJ International Conference on*, pages 5657–5662. IEEE, 2006.

[64] M. Stark, P. Lies, M. Zillich, J. Wyatt, and B. Schiele. Functional object class detection based on learned affordance cues. In *Computer Vision Systems*, pages 435–444. Springer, 2008.

[65] S. El-Khoury and A. Sahbani. A new strategy combining empirical and analytical approaches for grasping unknown 3d objects. *Robotics and Autonomous Systems*, 58(5):497–507, 2010.

[66] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *Acm computing surveys (CSUR)*, 38(4):13, 2006.

[67] Y. Sun, J. M. Hollerbach, and S. A. Mascaro. Predicting fingertip forces by imaging col- oration changes in the fingernail and surrounding skin. *Biomedical Engineering, IEEE Transactions on*, 55(10):2363–2371, 2008.

[68] Y. Lin and Y. Sun. 5-d force control system for fingernail imaging calibration. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1374–1379. IEEE, 2011.

[69] A. Freivalds, Y. Kong, H. You, and S. Park. A comprehensive risk assessment model for work-related musculoskeletal disorders of the upper extremities. In *Proceedings of the Hu- man Factors and Ergonomics Society Annual Meeting*, volume 44, pages 5–728. SAGE Publications, 2000.

[70] S. Calinon and A. Billard. Incremental learning of gestures by imitation in a humanoid robot. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, pages 255– 262. ACM, 2007.

[71] A. P Dempster, N. M. Laird, D. B Rubin, et al. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal statistical Society*, 39(1):1–38, 1977.

[72] F. Conti, D. Morris, F. Barbagli, and C. Sewell. Chai 3d. *Online: http://www. chai3d. org*, 2006.

[73] M. B. Wilk and R. Gnanadesikan. Probability plotting methods for the analysis for the analysis of data. *Biometrika*, 55(1):1–17, 1968.

[74] C. B. Barber and H. Huhdanpaa. Qhull. *The Geometry Center, University of Minnesota, http://www. geom. umn. edu/software/qhull*, 1995.

[75] I. P. Corke. *Robotics, Vision & Control: Fundamental Algorithms in Matlab*. Springer, 2011.

[76] J. R. Napier. Studies of the hands of living primates. In *Proceedings of the Zoological Society of London*, volume 134, pages 647–657. Wiley Online Library, 1960.

[77] R. L. Susman. Hand function and tool behavior in early hominids. *Journal of Human Evolution*, 35(1):23–46, 1998.

[78] M. W. Marzke and R. F. Marzke. Evolution of the human hand: approaches to acquiring, analysing and interpreting the anatomical evidence. *Journal of anatomy*, 197(01):121–140, 2000.

[79] R. W. Young. Evolution of the human hand: the role of throwing and clubbing. *Journal of Anatomy*, 202(1):165–174, 2003.

[80] E. Oztop and M. A. Arbib. Schema design and implementation of the grasp-related mirror neuron system. *Biological cybernetics*, 87(2):116–140, 2002.

[81] C.S. Lovchik and M.A. Diftler. The robonaut hand: a dexterous robot hand for space. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 2, pages 907–912 vol.2, 1999.

[82] Shadow Robot Company. Shadow dextrous hand.

[83] S. C Jacobsen, J. E Wood, DF Knutti, and K. B Biggers. The utah/mit dextrous hand: Work in progress. *The International Journal of Robotics Research*, 3(4):21–50, 1984.

[84] J. Butterfass, M. Grebenstein, H. Liu, and G. Hirzinger. Dlr-hand ii: next generation of a dextrous robot hand. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 1, pages 109–114 vol.1, 2001.

[85] Y. Lin and Y. Sun. Grasp mapping using locality preserving projections and knn regression. In *Robotics and Automation, 2013. Proceedings., 2013 IEEE International Conference on*, may 2013.

[86] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

[87] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179– 188, 1936.

[88] J. B. Tenenbaum. Mapping a manifold of perceptual observations. *Advances in neural information processing systems*, pages 682–688, 1998.

[89] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[90] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, volume 14, pages 585–591, 2001.

[91] X. He and P. Niyogi. Locality preserving projections. In *NIPS*, 2003.

[92] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, volume 14, pages 585–591, 2001.

[93] M. Santello, M. Flanders, and J.F. Soechting. Postural hand synergies for tool use. *The Journal of Neuroscience*, 18(23):10105–10115, 1998.

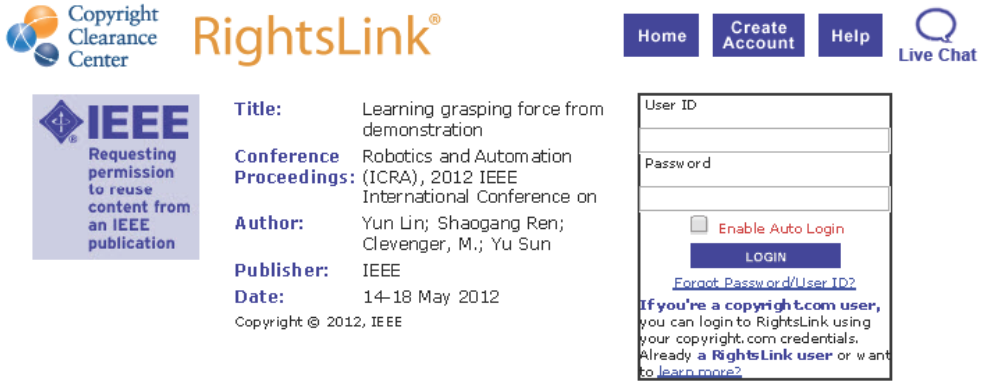
[94] M. T. Ciocarlie and P. K. Allen. Hand posture subspaces for dexterous robotic grasping. *The International Journal of Robotics Research*, 28(7):851–867, 2009.

[95] R. Balasubramanian, L. Xu, P. D. Brook, J. R. Smith, and Y. Matsuoka. Human-guided grasp measures improve grasp robustness on physical robot. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2294–2301, may 2010.

[96] Y. Lin and Y. Sun. Task-oriented grasp planning based on disturbance distribution. In *ISRR*, 2013.

[97] J. Kim, K. Iwamoto, J.J. Kuffner, Y. Ota, and N.S. Pollard. Physically based grasp quality evaluation under pose uncertainty. *IEEE Transactions on Robotics*, 29(6):1424–1439, 2013.

Appendix Copyright Permissions for Chapters 3, 4 and 6



Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

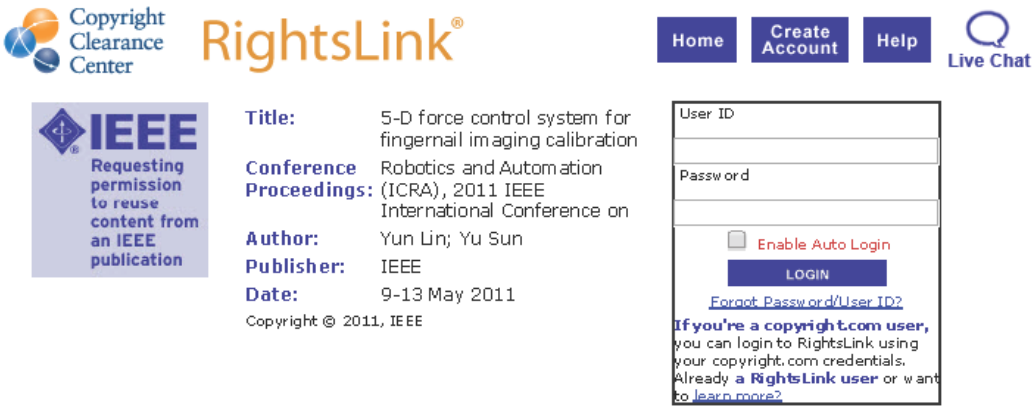
1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW



Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK | CLOSE WINDOW