Evaluation of Haptic Virtual Fixtures with Real-Time Sensors

Kevin Huang

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

Doctor of Philosophy

University of Washington

2017

Reading Committee:

Howard Jay Chizeck, Chair

Blake Hannaford

Joshua R. Smith

Program Authorized to Offer Degree:
Electrical Engineering

University of Washington

Abstract

Evaluation of Haptic Virtual Fixtures
with Real-Time Sensors

Kevin Huang

Chair of the Supervisory Committee:
Professor Howard Jay Chizeck
Electrical Engineering

Haptic virtual fixtures and haptic display can provide critical operator feedback and aid in completing teleoperated tasks. This is particularly true for cases in which visual information is not sufficient for the specific task, e.g. attempting to observe and manipulate a transparent object. This work improves upon one particular aspect of user feedback: conveying information to the user through the operator's sense of touch. Not only is feedback transparency of interest, but so too is the use of intelligent use of force feedback in the form of haptic virtual fixtures. Virtual fixtures help to modify the user's performance by encouraging or limiting their motions. In practice, providing touch or haptic feedback can be achieved with *a priori* or real-time information. The former presents problems in that virtual fixtures designed this way are useful only in predefined, well-known and unchanging scenarios. One promising implementation of the latter utilizes real-time streaming point clouds provided by commercially available RGB-Depth (RGB-D) cameras. Problems persist, however. For example, missing or gaps in point cloud data can be caused by material properties, lighting conditions, distance and other factors [1]. Surface geometries are a useful but not exhaustive set of information for teleoperation tasks. One solution is to augment the sparse/missing point cloud with additional geometry information provided by auxiliary sensors, or incorporate additional sensory information which RGB-D cameras are not suited to capture. Furthermore, feedback quali-

ties in telerobotic interaction can impact user performance. Novel visualization and haptic feedback modes are of interest, and use cases of these scenarios are demonstrated on real telerobotic platforms in this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

FORBIDDEN REGION FIXTURE: a type of virtual fixture which restricts motion into a particular spatial region.

GUIDANCE FIXTURE: a type of virtual fixture which encourages motion in a particular direction or along a desired path.

HAPTIC: of or relating to the human sense of touch. Can refer to haptic device, interface, display, rendering etc.

PRETOUCH: of or relating to non-contact proximity sensors integrated into robot manipulators.

TELEMANIPULATOR: robotic device remotely controlled by a human operator to complete a manipulation task in the remote space.

TELEOPERATION: the remote control of a robotic device, consists of a local (master) device and remote (slave) device.

VIRTUAL FIXTURE: a spatially varying constraint or function of position and its derivatives overlaid on a robotic workspace to assist or guide in the performance of a task. This can be applied through force feedback to a human operator or through direct restrictions on the remote device's motion.

VOXEL: an element of a regular grid of three-dimensional space. A 3D analog to the graphical pixel.

ACKNOWLEDGMENTS

I would like to thank my advisor Howard Jay Chizeck for his guidance and support. I would additionally like to thank members of the UW BioRobotics Laboratory. In particular, Fredrik Rydén provided a great deal of technical support and fundamental knowledge and experience.

Additionally, I want to acknowledge collaborators in the UW Sensor Systems Laboratory, Liang-Ting Jiang, Patrick Lancaster and Joshua R. Smith, for their excellent work and support. Maya Cakmak from the Human-Centered Robotics Lab also deserves recognition for her help and guidance in HRI studies. Finally, National Instruments also provided hardware pieces and robot platforms for experimentation.

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-1256082 Any opinion, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Chapter 1
INTRODUCTION

Robotic systems can offer many advantages over humans in certain tasks, including repeatability, scalability, access to small or large workspaces, strength and resistance to harmful conditions. As robotic technology advances, so too will the complexity of the tasks which humans will assign robots to perform. Current state-of-the-art allows for minimal or niche cases in which autonomous control is sufficient. When a robot must perceive or evaluate an unstructured or new environment, decision making is difficult and might be based off erroneous or incomplete data. While robots can be programmed to precisely execute certain motions, reacting to unexpected changes relevant to the purpose or environment of the task are better suited for human decision making. Fully autonomous robots are neither smart nor adaptable enough to function in real-world useful scenarios without human intervention.

With that said, human control of remote robots (telerobotics) can provide a means to bridge the high-level understanding and decision making of humans with the advantages of machines. Collaborative robots with human or human-in-the-loop interaction are needed to tackle the variable and complex nature of real-world tasks. The challenge then lies in providing sufficient feedback to the user for intuitive and safe control of the remote device. Visual and auditory channels are the easiest and most widely used, but drawbacks exist. For one, spatial distortion and lack of depth perception can result in improper positioning and control. Secondly, in more complex environments, the visual pathway may be overloaded when attempting to convey all relevant information. Another user feedback channel is necessary.

Leveraging the human haptic pathway can be a means to accommodate a more intuitive spatial feedback as well as incorporate more information in telerobotics. Already, haptic feedback and assistance modes called virtual fixtures can be built to either prevent contact with objects or guide a user along a desired path. Such assistance modes have been shown with quantitative results to be beneficial in scenarios when the environment is known *a priori* or in entirely virtual environments − no variability is introduced. The reality is, real-life implementations of haptic virtual fixtures involve force assistance modes developed only on one type of sensor and one form of guidance, limiting the potential use cases for haptic virtual fixtures in telerobotics.

The broad scope of this work is twofold: firstly technical implementation on real-life examples of combining multiple sensors and virtual fixtures in a telerobotic task, and second the evaluation of real-time haptic virtual fixtures from multiple sensor information. In the former, haptic virtual fixtures are implemented in a total of three different telerobotic tasks on a total of two different state-of-the-art robotic platforms. The scenarios lend themselves to the use of multiple sensors and multiple haptic virtual fixtures. In one of the robotic platforms, kinematics and joint limit calculations are performed from scratch. In the latter, the platforms are then evaluated via user studies to validate the efficacy of the haptic virtual fixtures.

1.1 Problem Statement

The problem addressed in this thesis involves the difficulty of synthesizing haptic virtual fixtures from real-time sensor information, and is twofold:

Firstly, current methods of real-time haptic interaction are dependent on a single sensor source. In state-of the art implementations, haptic interaction with streaming point clouds provides force feedback and guidance modes to the teleoperator based on geometry information in real-time. However the utilization point cloud information captured from RGB-D

cameras falls victim to drawbacks common to other imaging devices, namely occlusions and poorly observed objects due to material optical properties. RGB-D cameras may also be ill-suited for a particular task environment, e.g. maneuverability and size constraints. Another issue is that haptic assistance based on depth information assume a teleoperated task with execution constrained only by geometry of the environment. The reality is, however, that many teleoperation procedures rely on information sources beyond depth information. Finally, current real-time haptic rendering methods from RGB-D data are not amenable to physical interaction with the target objects.

Secondly, in order to deploy haptic virtual fixtures created from real-time sensor data, benefits and safety must first be validated. To do so, an ensemble of quantitative and qualitative metrics must be explored using these new proposed methods. Doing so with multiple sensor information sources as well as types of haptic virtual fixtures is ideal.

1.2 Proposed Solution

To address the first problem, i.e. the use of a single sensor source for building haptic virtual fixtures, different types of sensors fused with RGB-D point cloud data will be implemented on real robotic platforms. These implementations will be naive in nature, but will demonstrate the increased utility of haptic virtual fixtures in scenarios in which RGB-D data alone cannot provide enough information for the telerobotic task. For the intelligence and applicability sought for real-time sensor based haptic assistance, the use of multiple sensor sources is needed. Using the real-time data in a way to allow for haptic virtual fixtures to assist in physical manipulation of task-space objects will further extend the use of non-contact, real-time sensors. These implementations will demonstrate and afford a stepping stone towards haptic virtual fixtures from multiple sensors.

To validate the use of real-time sensor data for haptic virtual fixture generation, an extensive user-study will explore and evaluate relevant performance metrics. In critical teleoperated

tasks, quantitative measures of interest can include: time to completion, path smoothness, path length and collisions. Furthermore, an effective user interface should not burden the operator's situational awareness or task load. These quantitative and qualitative metrics will aid in assessing the viability of using haptic virtual fixtures from real-time sensors in real-world telerobotic applications.

1.3 Contributions

Real-time haptic interaction and haptic virtual fixtures rely on reliable sensor information. There are several cases when useful sensor data is unavailable. For example, when teleoperating a robot, the robot itself may occlude objects of interest. View angles, material properties and sensor limitations can also result in missing information and limit the capability of real-time haptic rendering. Interpolation can patch missing data, but makes assumptions that may not be reflected in reality. Other sensor based methods may rely on physical contact. The use of auxiliary sensors and teleoperation can provide non-contact means of gathering reliable data.

The main contributions of this thesis relate to the synthesis of haptic virtual fixtures from multiple sensor data in real-time for teleoperation, and can be summarized below and elaborated in Chapter 3. The work presented:

1. provides tangible examples on real robotic platforms of incorporating auxiliary depth information to RGB-D captured point clouds as well as multiple haptic virtual fixtures types for haptic interaction in practical teleoperation scenarios.
    - Non-contact methods for telerobotic acquisition of depth data for the purposes of haptic interaction is presented. Previous techniques for acquiring missing geometry information exist, but require contacts [2] or additional bulky, RGB-D cameras [3, 4]. These methods are thus unsuitable for delicate objects or con-

strained physical spaces.

2. provides a method for using real-time RGB-D data for the use of haptic-guidance in telemanipulation and physical interaction in a practical telemanipulation scenario.

    - Haptic guidance virtual fixtures have been shown to improve performance [5, 6], but these methods defined reference trajectories prior to operation. Using haptic guidance with real-time sensor data extends the use of fixtures to real world, tangible application. Furthermore, the real-time sensor data is allocated in an efficient, voxel representation which preserves observed information, and thus is amenable to temporary occlusions, dynamic interaction and displacement of the environment, aspects relevant to telemanipulation.

3. evaluates the efficacy and performance the above generated haptic virtual fixture assistance modes and conducts an extensive user study to validate feedback modes in telemanipulation.

    - Results show quantitative improvements by using teleoperation, pretouch sensing and haptic virtual fixtures for active surface reconstruction. The results are indeed promising for these approaches.

    - The effects of feedback modalities on teleoperation are crucial for designing useful and intuitive telerobotic interfaces. Effects of visualization and visual feedback have been explored for telerobot navigation [7], but factors for telemanipulation are not yet explored. This work provides quantitative and qualitative measures of the impact of visual and haptic feedback modes in telemanipulation.

Chapter 2
BACKGROUND

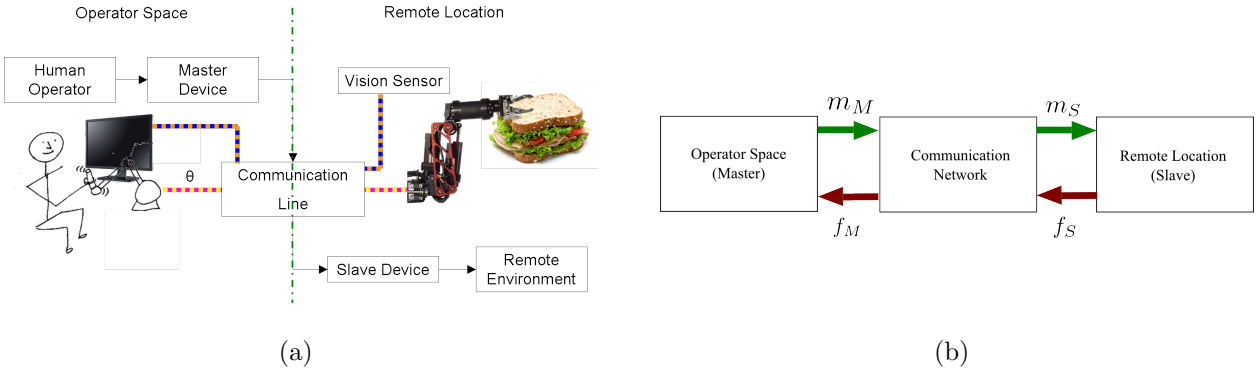
2.1 Teleoperation

Teleoperation entails the control of one or more remote manipulators (slave) through one or more local devices (master). The etymology of "teleoperation" naturally leads to a simplistic definition: operating at a distance. Teleoperation of robotic devices, or telerobotics, can be found in many applications in science, medicine, education, and military applications to name a few [8]. Several intriguing application are described by [9, 10].

2.1.1 Basic Architectures

The use of a teleoperation approach allows a human to manipulate objects remotely by attempting to relay conditions of the remote location to the operator [11]. Simultaneously, the operator can send motion commands via a master device through a communication network to the slave device in the remote location, which physically performs the teleoperated task. Many teleoperation systems have very similar underlying structure, making interoperability possible. Typically the operator is presented with visual feedback of the remote location, the manipulators (both in the operator space and remote location) have up to six degrees of freedom and kinematic functions are used to convert Cartesian information into the robot control workspace [12]. In bilateral teleoperation, in addition to recording motion commands of the user the master device can also apply forces which are sensed at the remote location, thus providing the user with tactile and force feedback and ideally increasing the operator's telepresence. Figure 2.1.1 illustrates the basic setup for teleoperation.

2.1(a) shows a basic unilateral teleoperation system. A human operator, master device and



(a) (b)

Figure 2.1: Basic Teleoperation Architectures

visual display are shown in the operator space. It should be noted that the control media on the operator side can come in various forms, including joysticks, wheels, pedals, cellular phones etc. [13–15]. In the remote location, the slave device interacts with the remote environment, in this case a sandwich, and a vision sensor is used to capture visual data of the surroundings. Motion commands are sent from the master device to the remote manipulator and vision data to the visual display through the communication line.

2.1(b) is a block diagram depicting a teleoperation system. m_M indicates motion commands (e.g. position, velocity) from the master device, which are translated into robot control workspace commands m_S at the slave device. Feedback information (e.g. visual feedback) from the remote location f_S are sent back to the operator space as f_M to be displayed. f_S and f_M containing force feedback indicate a bilateral teleoperation system.

2.1.2 Advantages

A teleoperation architecture is useful in cases where autonomy is not reasonable or feasible. This is true for complicated and highly dexterous robotic tasks, such as fine manipulation in a dynamic environment or even grasping unknown transparent objects.

In dangerous or delicate situations involving highly valuable assets, it is desirable for a

human with understanding of the task and situation to provide critical reaction and decision making where an autonomous robot cannot. When the task space is inaccessible for that human, teleoperation is a suitable solution. Augmenting human intention and task comprehension with telerobots is beneficial because:

- The remote location may be dangerous, and the task is better completed without risking a human life.
- Robots can be scaled to reach areas too small, too large or otherwise inaccessible to the human operator, e.g. underwater for long periods of time.
- Robots can be machined for high precision far beyond the capabilities of human dexterity, e.g. on the nanoscale.

Teleoperation combines the decision-making, situational understanding and experience of a human with the precision, repeatability and scalability of machines.

2.1.3 User Interface

Because of the physical separation between the operator space and the remote location, teleoperation can suffer from reduced operator situational awareness. Compromised awareness and other human performance issues can reduce effectiveness of teleoperated task completion.

Operator performance issues in teleoperation can stem from deficiencies in [16]:

1. remote perception
2. remote manipulation

Remote perception deals with conveying to the user information about the remote environment and the slave device's state through user feedback channels to increase situational awareness and immersion. When issues in remote perception occur, such as scale ambiguity,

degraded depth perception etc, the operator may have mis-perceptions about the affordances in the remote environment. Factors that can contribute to these include mismatching viewpoints in visual feedback, restricted field of view, and unimodal visual feedback to name a few [17]. These types of issues may be resolved by refining the quality of the user feedback data, incorporating additional data as well as utilizing multi-modal channels of perception, such as haptic feedback.

Remote manipulation concerns the type and mode of the operator input and affecting change in the remote environment or robot state. Ideally a natural and intuitive method of controlling the remote device is employed. Input devices and modes can include keyboards, computer mice, joysticks, voice and gesture commands, foot pedals and can incorporate bioelectric signals. In addition to these input devices, the master robotic device kinematics may conflict with those of the remote device. Providing sufficient input modalities and complexity of the master device to control the remote slave in a seamless manner without overwhelming the operator is a difficult balance to achieve. Some solutions include providing the operator with a modular software and hardware interface with which one can choose and change input hardware, command and control architecture (i.e. bilateral, unidirectional, supervisory control) [15]. Other factors that can degrade remote manipulation effectiveness include command latency, motion at the operator space, and feedback latency [17]. Supervisory control schemes that provide some level of autonomy to the slave device can help mitigate or mask the effects of degraded remote manipulation [11].

2.1.4 Model-Based Teleoperation

An extension of supervisory control involves the use of computer based virtual objects. These were initially created to compensate for lack of visibility from the remote device's cameras [18]. In the case of space robotics, virtual environments and model-based teleoperation with predictive display were used to overcome large time delays in the communication loop between the operator space and remote location [11]. Even in more localized teleoperation

procedures, lag from controllers or communication can be mitigated by the use of model-mediated teleoperation. However, when using model based or virtual reality based techniques for teleoperation, geometrical and physical modeling of the virtual world can have errors. The model can be updated to adjust for geometrical or physical simulation errors [19]. Another possible avenue is to update the virtual model based on sensor information from the remote location. In teleoperation scenarios where the remote environment is unstructured, the model can be built entirely off of sensor information which is treated as a time-varying signal. Pai et. al demonstrated a teleoperated means of collecting surface geometries of an object via contact force and torque sensors [2]. This approach is not suitable, however, in the case of delicate objects where contacts are potentially disastrous.

2.2 Haptic Display

Haptic display presents tactile information to a human user in order to convey the presence of some type of stimulus during a sensory-motor task. The most basic haptic display problems involve providing force feedback for human interaction with a purely virtual environment, a computer generated mathematical model of physical objects. Haptic interaction with a virtual environment is a bidirectional interaction, in which information is transferred both from and to the human user. As such, not only is the simulation of objects necessary (e.g. geometric modeling, collision detection and processing methods), but also an understanding of the human perception of touch. Touch can include sensations which are captured by sensors on the human skin (exteroceptors) as well as those in our muscles, joints and tendons (proprioceptors). In general, exteroceptors provide tactile sensations while proprioceptors are used for kinesthesia. In this work, we are interested in kinesthetic haptic sensation, which includes the sense of motion and forces. For realistic and effective kinesthetic haptic rendering, it has been widely accepted that a haptic update or servo rate of one kilohertz is required [20].

2.2.1 Basic Architecture for Kinesthetic Haptic Display

Haptic interaction with a computer generated virtual environment is often a multi-modal interaction that can be broken down into three major subsystems [20]:

1. Visual Rendering
2. Computer Simulation
3. Haptic Rendering

Visual rendering simply displays via graphics to the user the state of the computer simulation. The computer simulation provides the virtual environment with which the user can interact. This includes the virtual objects as well as the location of the user's cursor or haptic interaction point (HIP). The state of the simulation can also be updated by the haptic rendering subsystem.

In the haptic rendering subsystem, a user's position is sensed from encoders on the haptic device, which updates the position of the HIP in the simulation environment. Collision detection algorithms are used to determine points of contact between the HIP and any objects in the simulation, and appropriate interaction forces are computed, whether they be from rigid objects or compliant surfaces. Control algorithms are used to display the computed force and apply to the user via the haptic device while ensuring stable behavior. In this way, forces that a user would feel by interacting with real objects can be simulated and recreated.

2.2.2 Proxy Method

In a simple geometry-dependent force rendering algorithm in which force is proportional to the deviation of the HIP to the geometric surface, several problems exist. For example, passing the HIP through the corner of an object can result in jumps in force direction from

one face of the corner to the other. Pop-through is also an inexorable problem (particularly with thin or multiple objects). To combat this Zilles et al. first proposed a virtual coupling method, in which force feedback was calculated as a mass-spring-damper system between the HIP and a moving point called a god-object [21]. This god-object did not violate any surface constraints of the polygonal models through basic collision detection such as binary space partitioning. Rendered in the virtual environment, the god-object followed the HIP until limited by surface constraints, at which point force feedback was modeled as a spring between god-object and HIP. Later, Ruspini et al. extended the notion of the god-object method to the proxy method, in which the god-object point was replaced with a sphere [22]. This prevented errors when small holes were present in the virtual model geometry. The proxy method can be extended further to geometrical shapes and objects that represent the user's "tool" through which the user interacts with the simulation, much like a stylus. With proper specifications and tuning, such force feedback can significantly enrich and affect the sense of immersion and the ability of interaction with a virtual environment [23]. The challenge can lie in generating realistic and believable proxy motion that follows physical constraints of the simulation.

2.2.3 Higher Degrees of Freedom

The above descriptions of haptic rendering methods involve the user moving the cursor or HIP as though it were a point in space. As such, haptic feedback is presented in only three degrees of freedom (DOF). In some situations, it is useful to understand force interactions not just in a point-object manner, but object-object. In this case, a user manipulates not just a point in the virtual simulation, but rather an object (like a stylus or pen). This incorporates a rotational or torque component to the force feedback, resulting in six DOF rendering.

The main challenge with 6-DOF haptic rendering is supporting a greater complexity of collision detection while simultaneously maintaining a haptic update rate of 1000 Hz [24]. Several methods have been proposed, including voxel based methods [25, 26], algorithms

based on generalized penetration depth [27], and constraint based proxy methods [28] which have been used for rendering with streaming point clouds [29].

2.3 Haptic Virtual Fixtures

The above haptic display methods can be extended beyond virtual interaction and also to assist a human user. As described previously, certain situations are amenable to human control of a robotic device, either through hands-on interaction or teleoperation. By using a haptic virtual fixture, a robot can be used to intelligently modify and regulate the motion commands of the human user. These fixtures can be used to limit tool motion to aid in task execution, and are enforced in software. During a task, the system monitors the user's motion commands, analyzes against software bounds and constraints, and finally modifies the user's motion based on violation of virtual fixtures.

In this way, given a specific telerobotic task, haptic virtual fixtures can be used to improve the efficacy of its execution. Virtual fixtures augment a human's intelligence of a dexterous task with assistive force feedback and the precision of a robotic system, and in the case of telemanipulation the presence of the slave device in a remote environment. For example, when the remote manipulator's desired path (reference curve) or destination (reference target) is determined, a virtual fixture can ensure that the operator can command the manipulator only along such a path or towards the end goal. Figure 2.3 shows a manipulator being guided with a virtual fixture either to an end target or along a target path.

With this fixture, oftentimes a guidance force is applied to the operator when he or she provides movement commands away from the desired path or direction (these directions are shown above in Figure 2.3 with arrows). However it should be noted that guidance in the master robot workspace is not a necessity, and a virtual force feedback is feasible. Abbott et al. developed a method for guidance virtual fixtures in telemanipulation, and coined the term pseudo admittance guidance fixture to describe a proxy moving with admittance

(a) (b)

Figure 2.2: Reference Target and Reference Curve Guidance Virtual Fixtures [30]

type control with the master side robot as the input force. The slave or remote device then servos to the corresponding proxy position [31]. Haptic guidance on the master side, however, can provide intuition to the operator about the slave response to master commands.

While guidance virtual fixtures help the telemanipulated device reach a desired location, forbidden region fixtures are used to prevent either the master or slave from entering a specified region, typically in the vicinity of a sensitive structure. These can be thought of as virtual force fields. In some teleoperation tasks, often regions to be protected are physical objects and thus consistent with surface geometries.

Much work in virtual fixtures deal with constraints and bounds defined a priori, which can be done with or without a human supervisor. Stability of the environment is assumed such that static virtual fixtures or predictable fixtures are suitable. In [32], the user is presented with a micro-injection task. Prior to execution, the user is given a two-dimensional image of the task, and is allowed to manually define forbidden regions prior to execution. In [33], an a priori computer-aided tomography (CT) scan of a nasal cavity was used for Ears, Nose, and Throat (ENT) surgery procedures. Haptic virtual fixtures were based on geometries captured from the *a priori* scan. Similar approaches were used by [34] for robotic cardiac catheter navigation. [35] demonstrated the use of a predictive model for haptic virtual fixtures with an oscillating platform. This assumed a very predictable motion and was claimed to simulate

motions of a beating heart. In [36], preoperative magnetic resonance imaging (MRI) and CT scans were used to build a heart model at different stages during the heart beat. This model could be interacted with haptically with forbidden region virtual fixtures, and the model instances were matched and chosen by periodic ultrasound scans and electrocardiogram signals. While these methods provide insight into the process and factors involved in generating haptic virtual fixtures for certain tasks, they are not amenable to unstructured and unpredictable environments. Furthermore, any interaction with the environment that might violate these assumptions would render such haptic virtual fixtures invalid.

Because these methods are not amenable to dynamic environments, efforts have been made towards building virtual fixtures based on unstructured, real-time data. Navkar et al. used multiple sections of real-time MRI scans of a beating heart to construct a beating heart model with which one could haptically interact [37]. While this work is novel, it is limited by the *a priori* selection of only three MR slices to depict the anatomical structures of interest, at 50ms per slice. Haptic interaction was performed offline and in simulation only. In application, the presence of an interventional tool distorts the MRI. Another attempt at haptic virtual fixtures based on information gathered during operation is that based off of geometry data in the form of point clouds. Streaming point cloud data can be collected from an RGB+Depth (RGB-D) camera (e.g. XBox Kinect 30Hz at 640x480 resolution) in real time. However, representing objects and contacts for model-based haptic interaction can result in burdensome computational load requirements. Intuitive solutions, such as interpolation to construct polygons prove too computationally heavy to satisfy the haptic update rate requirement. Instead, Ryden et al. extended the Ruspini proxy method to include three separate radii extending from the proxy center, designating states of entrenchment, contact, and free motion [38]. Using only neighboring points to the proxy, a surface normal, if it exists, can be calculated in real-time. Proxy movement is then based on the number of points within the three radii, and force is rendered via Hooke's law. In this way, the haptic rendering method can compute real-time force feedback from streaming point clouds without

exhaustive computation. This is aided even further by performing repeatable, routine tasks in parallel on a graphics processor.

To build virtual forbidden regions, the above haptic display method was implemented after registering the position and orientation of the Raven II robotic telesurgical device with streaming point cloud data.

(a) (b)

Figure 2.3: Forbidden Region Virtual Fixture From Point Clouds [39]

The point cloud data provides a software based representation and model of the remote environment with which the operator can interact using a haptic interface device. Concurrently, the remote robot device follows, within the remote environment, the user manipulated proxy in the virtual representation. The forbidden region virtual fixtures were then rendered by defining sphere objects of a given radius around each point within the forbidden region. In Figure 2.3(a), the user moves the HIP in the virtual model while the telerobot follows the proxy location in the remote environment as represented in the virtual representation. A force F is provided to the user. In Figure 2.3(b), the distance of the protective force field is determined by $r_{f,i}$ around each point to be protected. The objective was to provide assistance to the user by restricting the robot device's motion and applying a force to the operator when entering the forbidden region [39]. This method was used to protect arbitrarily moving objects in an unstructured environment using forbidden region virtual fixtures and makes no assumptions about the objects in the environment.

2.4 Sensor Limitations in Bilateral Teleoperation

Oftentimes in a teleoperation system, the user is presented with visual feedback of the remote environment as a means to increase user immersion. It is common for just a video stream from a camera on the robot to be provided. However, studies have shown that merely monocular video streams can lead to many collisions [40], long completion times [41], and high workloads [42]. In certain teleoperation navigation tasks, it was shown that providing depth information in the form of a point cloud (captured from RGB-D cameras), could improve operator performance and reduce perceived task load. Such vision techniques were also shown to aid in the predictive nature of model-based teleoperation [43]. In situations with much clutter and complicated obstacles, it was shown through user studies that 3D representations of the remote environment reduced path length, number of collisions, perceived workload and completion time (the latter two were not true when situational awareness was degraded with just monocular video streams) [7]. While the use of point cloud depth information can be useful, there are issues that need to be resolved.

2.4.1 Occlusions and Poorly Visible Geometries

While modern commodity RGB-D cameras provide real-time geometry information of an observed scene, the costs of this real-time representation include sparse scenes littered with 'holes' and incomplete geometries occurring from occlusions, optical properties, glancing incidence angles and other factors [1,44]. This is undesirable for applications in which accurate and dense 3D information retrieval is essential for the task at hand, such as grasping.

Several approaches have been taken to produce denser point clouds from these sensors and to increase coverage of point cloud data. In [44], a global implicit surface representation of an indoor scene was constructed and built up in real-time from multiple scans using the same Kinect sensor. While this method provides a dense reconstruction of a room-sized surface, localization of scans can be problematic at different scales.

In another method, it was found that the simple addition of another RGB-D sensor can effectively extend coverage, fill in shadows and occlusion that would be present with a single camera. In [1], [45], multiple Kinects were used to reconstruct an entire instrumented room. The key to using identical cameras using similar structured light pattern sensing methods was eliminating interference, which was achieved with motion. [1] used separate high frequency motion patterns for each camera, while [45] utilized a simpler technique consisting of offset-weight vibration motors. In [46], it was shown that different point sets with varying sensor characteristics (for example, different RGB-D technologies) were registered with each other using enhanced iterative closest point algorithms for the purposes of metrology. There are drawbacks, however, to the "hole-filling" methods using an additional RGB-D camera for the purposes of telerobotic manipulation. Take, for example, a teleoperated robot exploring a new, unknown environment. In such a scenario, it may be impossible to strategically place a second depth sensor. Even if the additional sensor is mounted to the robot, the added mechanical interference could reduce precision in kinematic calculations and could furthermore damage the robot.

RGB-D cameras also suffer from a well-known problem in the field of computer vision: the inability to accurately identify the shape or configuration of transparent objects. Many household objects are transparent, e.g. bottles, glasses, windows etc. For the purposes of robots in the everyday household, teleoperated or not, being able to perceive and localize transparent objects precedes safe interaction or grasping. Several automated methods have been used to address the issue of grasping transparent objects, some of which require the object to be known beforehand [47] and others may require multiple scans and produce less than ideal results [48].

2.4.2 Insufficeint Information for Task

In addition to problem of gaps and incomplete information from depth cameras, sometimes the type of data (surface geometries) is not sufficient for the teleoperated task. Incorporating information beyond what can be captured from an RGB-D camera is needed to achieve this. Additional data might include, for example, the use of medical scan images, higher resolution imaging, thermal scans or bio-sensors. Incorporating different sensor modalities might be achieved by switching haptic rendering states. Bettini et al. did this with two dimensional imaging based guidance virtual fixtures [30]. In this method, both a macro-scale and micro-scale vision sensor were used to track the reference target and end effector location. The algorithm employed a control law which switched between a path following and positioning problem depending on distance between tool-tip and desired location. When the tool was more than a certain distance away from the desired end location, the user was guided in a direction tangent to the closest point on the reference path, as well as slightly towards the reference target. When within a predefined radius, the HIP was controlled with a simple positioning algorithm to converge to the desired end point and the micro scale camera was utilized. This method could be extended to building guidance fixtures for teleoperation from image based sensor information beyond vision information. Simultaneous integration may require handling of conflicting information and management of heterogeneous information. In order to provide intelligent force feedback in real-world telerobotic applications, multiple sensor sources and information may be relevant. Therein lies the problem, for these different sources may either complement one another, support one another or even contradict each other.

Chapter 3
RESEARCH AIMS

In a broader sense, this research will contribute towards enhancing haptic user interfaces for teleoperation. Because of the inherent physical separation between the remote and proximal locations, teleoperators must estimate remote environment properties, often on visual sensor data alone. Ideally, additional sensor information could be intelligently and seamlessly integrated into the feedback provided to operators. While some technology has provided seminal steps towards this vision, problems persist:

1. The addition of depth information from RGB-D cameras can provide a certain degree of spatial awareness not available with monocular RGB streams. However, when isolating features of interest solely from image segmentation (color and depth), two fundamental issues arise. Firstly, occlusions, shadows or regions unseen by the RGB-D camera are not rendered and present holes and incomplete representations in the virtual environment. Secondly, in the case where a virtual fixture is desired around a region or object not readily identifiable by its color or depth, an RGB-D camera is insufficient. The problem lies in incorporating information beyond depth and color that helps to identify regions of interest and thus virtual fixtures. Methods for this exist, but ultimately rely on contact-type sensors [2]. The goal is to address these issues by demonstrating several cases where haptic virtual fixtures in combination with non-contact pretouch sensors can aid in an unstructured, teleoperated task.

2. Another problem lies in the use of RGB-D cameras for generating real-time haptic virtual fixtures for physical telerobotic interaction. When interacting with an object, oftentimes a manipulator will both occlude and displace the object of interest. This can

result in missing information. The goal is to provide a method for preserving previously observed data, which is essential to enabling the use of haptic virtual fixtures with real-time data in telemanipulation.

The following tangible contributions work towards solving the aforementioned issues regarding haptic virtual fixtures in telerobotics.

3.1 Demonstrate Real Telerobotic Examples of Augmenting Streaming Point Clouds with Additional Sensors

To address holes and missing surface geometry information which is potentially critical for the telemanipulation task, a methodology is proposed for inserting new points into the point cloud from end-effector mounted sensors robust to optical transparency and simultaneously providing haptic virtual fixtures with working results. End-effector mounted sensors provides more maneuverability of the sensor. This same technique can be employed to interrogate geometries of other poorly observed surfaces due to a variety of reasons, e.g. occlusion, glancing incident angles, etc.

In another scenario, surface geometry data alone may not be sufficient for completing a telerobotic task. For example, perhaps a region or object with a particular bio-marker or heat signature is sought. Geometry data and relevant sensor measurands need to be fused for effective haptic feedback. In our implementation, the position of the object of interest is tracked generally but without surface geometry information. An array of end-effector depth sensors are fused with the tracked measurand and haptic virtual fixtures employed to build a 3D voxel representation of the object of interest.

3.1.1 Anticipated Outcomes

In unstructured environments and high-value teleoperation tasks, building an accurate and useful model of the remote location and reflecting this information to the user is both difficult and important. Providing intelligent regulation of the user's action can help to ensure safe

and efficient execution in unknown scenarios. Physical demonstration of use cases with real robots and tool-tip proximity sensors provides a tangible example of working results. Furthermore, showing feasibility of augmenting point-cloud geometry information with tool-tip sensors can be extended to a variety of sensors which can potentially fill in missing point cloud data. This also provides potential verification that improvements to non-contact haptic rendering for teleoperation are achievable with auxiliary sensors in unstructured or unknown scenarios.

3.2 Demonstrate Haptic Virtual Fixtures for Telemanipulation on Real Platforms

While real-time geometry data can be used to simulate virtual contact events with real objects, actual physical interaction with the task space presents several problems. Firstly, when interacting with a target, the telemanipulator itself can occlude the object. This can make fine motions and positioning difficult for the operator. Moreover, once contact is made, objects can be displaced. During interaction and use of a haptic guidance fixture, the teleoperator needs to be able to ascertain the condition or quality of physical interaction, for example to intervene or continue with the intelligent haptic feedback. Thus, a method for observing and retaining geometry information, perhaps visually, during interaction is needed.

3.2.1 Anticipated Outcomes

A method for preserving and displaying RGB-D data for telemanipulation provides useful visual aids for physically interacting with objects. More importantly, this enables an operator to monitor and use a haptic guidance fixture which involves physical contact and perturbation of the task space.

3.3 Evaluate Haptic Virtual Fixtures in Telerobotics

Haptic virtual fixtures have been quantitatively shown to improve performance in simulations and cases not involving real-time sensor data. For ubiquitous use and deployment, simple

experiments to evaluate baseline performance effects of haptic virtual fixtures in telerobotics as well as seminal evaluation of novel sensor fusion based haptic virtual fixtures are needed. In this section, we evaluate through extensive user studies the efficacy of a guidance virtual fixture, 3D voxel representation and standard RGB stream for a simple telemanipulation task. Furthermore, we evaluate two novel methods for haptic virtual fixture generation based on real-time data. These tests are executed on multiple robotic platforms.

3.3.1 Anticipated Outcomes

The operator performance effects of haptic virtual fixtures and different visual feedback modes is explored, and quantitative results provide insight into the benefits and the drawbacks of force feedback in telemanipulation. Furthermore, the implementations of haptic feedback from multiple sensor information and haptic virtual fixture fusion are examined in terms of efficacy towards the specific task in question. These results will hopefully show that even with more naive approaches to multi-sensor haptic feedback, improvements are observed.

Chapter 4

SENSOR AND VIRTUAL FIXTURE FUSION FOR TELEROBOTICS

In this chapter, two cases are presented of virtual fixtures or sensors combined in a real telerobotic architecture, employed on a tangible robotic platform. The combination of sensors or virtual fixtures enables the completion of a telerobotic task that is not easily achieved with conventional methods that employ one type of sensor or virtual fixture. The two examples cases are as follows:

1. *Sensor-Aided Teleoperated Grasping of Transparent Objects*

   Several types of objects (due to their material properties) are not observable by an infrared structured light depth camera. One type of object property that is subject to poor visibility is one of optical transparency. These objects result in missing data when viewed through a standard RGB-D camera. To overcome this, we propose a methodology for inserting new points into the point cloud from end-effector mounted sensors robust to optical transparency and provide haptic virtual fixtures with working results. End-effector mounted sensors provides more maneuverability of the sensor. This same technique can be employed to interrogate geometries of other poorly observed surfaces due to a variety of reasons, e.g. occlusion, glancing incident angles, etc.

2. *Fusing Non-Geometric Tracking with Depth Data*

   In another scenario, surface geometry data alone may not be sufficient for completing a telerobotic task. For example, perhaps a region or object with a particular

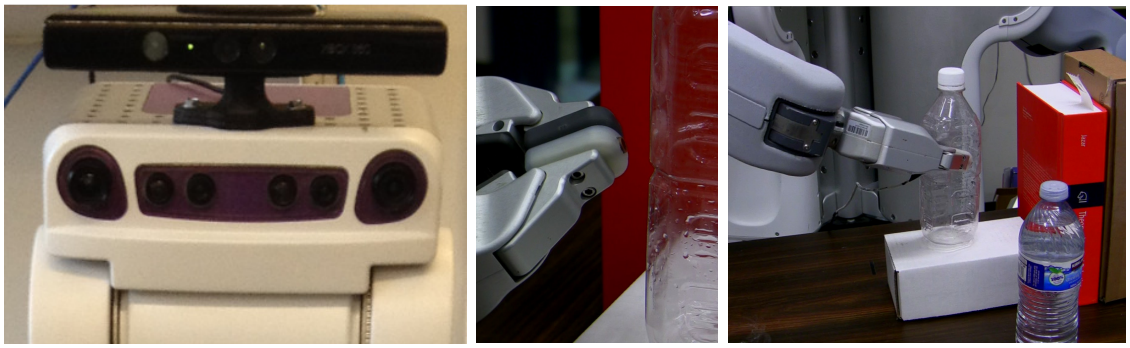
bio-marker or heat signature is sought. Geometry data and relevant sensor measurands need to be fused for effective haptic feedback. In our implementation, the position of the object of interest is tracked generally but without surface geometry information. An array of end-effector depth sensors are fused with the tracked measurand and haptic virtual fixtures (fused guidance and forbidden region) employed to build a 3D voxel representation of the object of interest.

The above two telerobotic scenario motivations, setups and algorithms are described in detail in the below sections.

4.1 Sensor-Aided Teleoperated Grasping of Transparent Objects

To address the incomplete point cloud data and unobserved transparent target, 'pretouch' depth information gathered from the robot tool-tip can be used to populate sparse areas of interest. Pretouch sensing [49, 50] refers to non-contact proximity sensors integrated into robot manipulators. Because it is a non-contact mechanism, the manipulator does not displace the object. As additional geometric data from the pretouch channel are incorporated into the point cloud representation of the grasp target, haptic rendering methods for streaming point clouds [51], [38] as well as forbidden-region virtual fixtures [39] are used to enforce a minimum distance between the robot end effector and sensed object surfaces. This prevents undesired contact with the object.

4.1.1 System Description



(a) (b) (c)

Figure 4.1: Robotic task overview. (a) PR2 headmounted kinect, (b) end effector seashell sensor, (c) PR2 grasping object.

The setup includes a bilateral teleoperation arrangement, in which the slave robot is a Willow Garage PR2 robot with a head-mounted RGB-D camera, the Microsoft Kinect. A seashell effect (i.e. based on frequency response change in an acoustic resonator [49]) pretouch sensor is integrated into the PR2 gripper to sense geometries undetected by the Kinect. These features are shown in Fig 4.1. The sensor has very reliable sensing range of $[1, 5]$ mm. On the master console side, the teleoperator manipulates a master haptic device, the Sensable ® PHANToM™Omni. This master device both sends position and orientation commands to one of the PR2's arms as well as receives and displays 3DOF haptic force feedback commands. With this setup, the teleoperator's goal is to explore a transparent, potentially sensitive object that is absent in the point cloud gathered from the RGB-D camera. With this setup, the system can integrate pretouch sensing and haptic feedback into teleoperation for exploration of otherwise poorly observed geometries. This is useful, for example, prior to attempting to grasp a transparent object.

4.1.2 Grasping Transparent Objects

Transparent materials and objects present a glaring problem in robotic grasping. Namely, for a robot, identifying and recognizing transparent objects using computer vision techniques is difficult. The visibility of such objects may change heavily based on background and lighting conditions, edges may be indistinct and there may be no distinguishable texture to isolate the object. For these reasons, it is troublesome to precisely identify configurations of or even segment transparent objects using computer vision alone, as seen in Figure 4.2 [47]. Commerically available RGB-D cameras provide a means to obtain geometrical point cloud data of an observed scene. However, a method of reconstructing unseen transparent objects without prior knowledge is still needed [52, 53], as these sensors cannot reliably measure depth of such objects.

(a) Kinect RGB image (b) Kinect depth image

Figure 4.2: Kinect RGB vs Depth Image of Transparent Objects [47]

Several approaches have been proposed to reconstruct transparent objects in point cloud data. Lysenkov et al. presented a robust method for identifying transparent objects and used 3D models that were fit to segmented, transparent regions of depth images [47]. While success rates of over 80% percent are achieved, this method had several limitations. Most notably, this required prior knowledge of the shape and size of the object, and used an as-

sumption that the object is supported on a flat surface from below. These assumptions and requirements are not reasonable in teleoperation scenarios in unknown environments.

In [48], another method for identifying transparent surfaces was presented that did not require any prior library of potential objects. Using a TOF camera, Klank et al. relied on a consistent misreading of depth information of transparent objects. Namely, distance measurements of transparent objects were in accordance to opaque surfaces behind the object, thus creating a shadow effect. By moving the TOF camera, two views and shadows are observed while matching and triangulation are performed to reconstruct a point cloud of the transparent object. This method was tested on five different transparent objects. The objects were successfully reconstructed 55% of the time, with overall 23% successful grasps [48]. While the algorithm is intriguing in its ability to reconstruct transparent objects without prior training, the relatively high miss rate can result in undesired collisions with transparent objects.

The automatic reconstruction of transparent objects based only on depth camera information is not amenable to robotic manipulation in unknown, sensitive environments. Therefore, we are more interested in methods using additional sensors. In conjunction, pose estimation of a teleoperated robotic arm is used to register the additional sensor information, effectively augmenting point cloud information with data collected in sparse regions of the point cloud.

4.1.3 Patching Sparse Point Clouds

Transparent objects present similar problems as occlusions and other poorly visible materials for current commodity RGB-D cameras in reconstructing geometries of an observed scene. Several approaches have been taken to produce denser point clouds to alleviate the effects of occlusions and shadows, increasing overall coverage of point cloud data. In [44], a global implicit surface representation of an indoor scene was constructed in real-time from multiple scans using the same Kinect sensor. In [1], [45], multiple Kinects were used to reconstruct

an entire instrumented room. The key to using identical cameras using similar structured light pattern sensing methods was eliminating interference, which was achieved with motion. While these methods provide a dense reconstruction of a room-sized surface, localization of scans can be problematic at different scales. Moreover, these approaches require the movement or placement RGB-D cameras around the target surface, a stipulation that may not be feasible in telerobotics or constrained environments. More critically, these method do not address transparent objects.

In [46], it was shown that different point sets with varying sensor characteristics (for example, different RGB-D technologies) were registered with each other using enhanced iterative closest point algorithms for the purposes of metrology. While this method provides a way to integrate homogeneous scans of the same static measurand from different types of sensors, strategically placing and registering two or more separate RGB-D cameras may prove to be troublesome, because of space requirements or simply by the nature of the task at hand. Furthermore, in the case of telerobotics, oftentimes the region of particular interest is located within the direct vicinity of the robot end effector. In this case, the end effector itself may occlude the region of interest from RGB-D cameras resulting in missing data. This is particularly troublesome for grasping tasks, where the robot end effector is in close proximity of the surfaces of interest.

End effector mounted sensors are thus more amenable to point-cloud reconstruction for grasping tasks than 'arena' sensors such as RGB-D cameras. This is partially due to the small scale reconstruction task (grasp object) relative to the entire environment. The use of tool-tip fixed sensors solves two problems. Firstly, the end effector itself will not occlude measurements. Secondly, the issue of registration is left to sensor precision and accuracy, assuming the robot kinematics are well known and pose estimation is reliable. Furthermore, there are several extant, real-time depth sensors capable of sensing transparent surfaces.

To explore sparse regions of the point cloud, the end effector sensor may either be tactile or non-contact. Tactile sensing may require several probes or palpations in order to acquire an adequate geometrical representation of an object. One benefit of this method is that the reliability of the sensor measurements are left to kinematics. The disadvantage, however, is that contact is required for sensing [54]. Thus, tactile sensing could potentially displace the object of interest, and as more data are acquired, the obtained point cloud becomes increasingly distorted as objects move [55]. Moreover, collisions in sensitive environments could be disastrous. Non-contact sensors are thus preferred as a means to gather geometries of sparse point cloud regions.

In this work, a head-mounted Kinect on the PR2 robot is used to capture RGB-D information and a seashell effect pretouch sensor for local sensing. Because of this, pretouch sensor information and point cloud registration can be done simply through the PR2's kinematics, since both sensors are mounted to known robot frames. Jiang et al. showed that a seashell sensor possesses many desirable traits and several advantages over electrical or optical approaches (namely sensing is not affected by transparency) [49]. Using this sensor, reliable depth information within the vicinity of the end effector are easily collected prior to grasp. As the robot end effector explores, the point cloud representation becomes denser in the explored regions proximal to the grasp target without contact. In this way, geometries of previously unseen regions can be measured, leading to more efficient grasping.

4.1.4 Haptic Virtual Fixtures for Grasping

For the case of automation or semi-automation, heuristic grasp planning algorithms can be implemented to handle the uncertainties of a grasp object [49], [56]. Jiang et al. showed an iterative grasp re-planning and exploration framework with a pretouch sensor to reduce object shape uncertainty prior to grasping. However, the method is time-consuming and does not prevent collisions [57]. In [58], a method for constraint-aware-teleoperation is proposed, whereby motion planning is implemented adhering to kinematic and collision constraints.

In this method, the user provides an end pose goal and the algorithm generates a motion path under constraints. This includes collision constraints, however, that are captured from a Kinect camera and is thus prone to miss features of transparent objects. Occlusions are also problematic, and local features are thus unaccounted for.

An integral part to successful implementation of haptic rendering is collision prevention, which is particularly difficult with sparse point sets. For this, a tool-tip mounted depth sensor (seashell effect pretouch sensor) is used to sense geometries of sparsely populated regions of the point cloud. This sensor is an open-ended pipe attached to a microphone, and acts as an acoustic resonator whose resonant frequency varies with depth.

While the seashell sensor can sense a one dimensional depth reading, the haptic rendering algorithm relies on a minimum density and number of points within the vicinity of any point along a surface. To ensure a sufficient density of sensed points along the surface, a forbidden region virtual fixture is used around seashell sensor detected points, and the specific geometries of this virtual fixture must be taken into account while exploring a sparse region.

The grasping target is a convex geometrical surface without acute angles. A concave object introduces potential undesired collisions during exploration. Imagine scanning with an end effector sensor the inside of a bowl versus the outside without prior knowledge of its curvature; exploring the inside is much more prone to contact. The seashell sensor explores the unseen object with sensor direction near normal to the surface of interest (assuming exploration starts with known areas of the object). Exploration and search for local features of the grasping target is investigated here, and thus the master position commands are scaled down in the virtual environment. Consequently, the maximum robot end effector acceleration and velocity exceeds that of the haptic interaction point within the virtual rendering. In the following section, analysis is performed in a 2D simplification for visual purposes. However the results can be extended directly to 3D.

Bounds on tracking error by sensing range

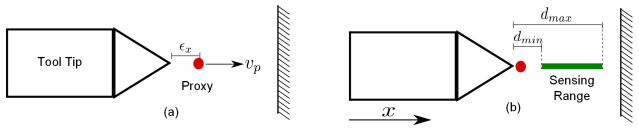


Figure 4.3: (a) Basic seashell sensor exploration as the robot end-effector approaches a surface. The seashell sensor follows the location of the proxy with tracking error $\vec{\epsilon}$, (b) sensing range defined by d_{min} and d_{max} and the surface is not yet seen.

Within the robot's physical task space, there is a finite expected maximum, steady-state position tracking error. Given a static position command within the task space, let the maximum deviation of the converged robot position from the commanded location be called the steady state position tracking error, $\vec{\epsilon_t}$. This deviation can be a result of several factors, including the robot controller, kinematics limits, and registration errors.

The robot end effector is commanded to follow a point called the proxy. The pretouch sensor is used to sense distances in front of the robot end effector, and provides reliable measurements only within a finite range. It is thus critical to consider compatibility of the sensing range and robot's position tracking, i.e. we must ensure that the sensor is able to sense a surface before the robot's actual or commanded position (proxy) penetrates it. To that end, consider Fig 4.3, which shows a basic exploration case: approaching a flat object normal to the surface.

To establish limits on the tracking error based on sensing range, assume a latency, τ, between commanded movement and robot actuation. Furthermore, let the proxy speed be limited by an upper bound, denoted v_{max}. From this, a worst case net position tracking error, $\vec{\epsilon}$, is given as

$$\vec{\epsilon} = v_{max} \left[\frac{\vec{v_p}}{|\vec{v_p}|} \right] \tau + \vec{\epsilon_t} \tag{4.1}$$

where $\vec{v_p}$ is the instantaneous velocity of the proxy and $\vec{\epsilon_t}$ is the steady state tracking error previously described. The goal of the haptic virtual fixtures is to prevent the proxy from

penetrating the object surface. From Fig 4.3, it is clear that

$$\epsilon_x < d_{max} \tag{4.2}$$

must be enforced in order for the surface to be sensed prior to the proxy penetrating it, where ϵ_x is the net position tracking error in direction x normal to the surface.

Bounds on virtual fixture radius

Once the surface is sensed, we would like to use virtual fixtures to prevent collisions during exploration around the sensed point. Establishing bounds on virtual fixture radius, call it r_v, requires further analysis. Again, first consider motion in the x direction, and suppose now that $\epsilon_x < 0$. Assume furthermore that the seashell sensor has sensed a point on the surface, as shown below in Fig 4.4(a). It is clear that to prevent the tool-tip from touching the surface, the following inequality must be satisfied

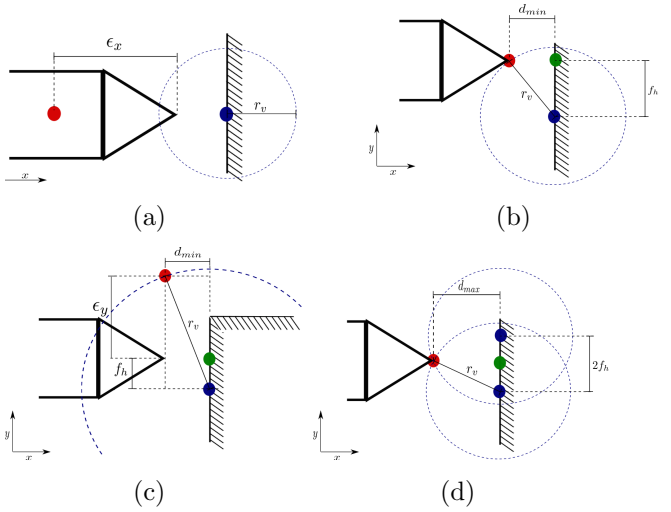


Figure 4.4: (a) lower bound on r_v, where $\epsilon_x < 0$, (b) lower bound on r_v while moving parallel to the surface to sense the next point (green), (c) virtual fixture forces end effector to travel f_h in y to sense (green) before it can travel in x to within d_{min} of surface, (d) upper bound on r_v to allow point spacing of f_h.

$$r_v > -\epsilon_x \tag{4.3}$$

and when $\epsilon_x > 0$, the lower bound on r_v is 0 ($r_v \geq 0$).

Suppose now that we would like to find additional points on the surface at a desired spacing of f_h. To that end, suppose the tool-tip may also move in a direction perpendicular to x. Without loss of generality, let that direction be y (shown in Fig 4.4(b)(c)(d)). Then, the

virtual fixture must now also prevent the tool-tip from reaching within d_{min} of the surface before a new viable point is sensed by the seashell sensor. Let $\vec{\epsilon} = \vec{0}$ as shown in Fig 4.4(b). From this, a bound on r_v is found as

$$r_v \geq \sqrt{{d_{min}}^2 + (f_h)^2} \tag{4.4}$$

Now consider both movement and error in a direction perpendicular to x, as shown in 4.4(c). The danger is this situation again is the tool-tip contacting the physical object despite the proxy being constrained by the virtual fixtures. Note that in this failure case, the proxy is free to move in the x direction, and subsequently so may the tool-tip. Several factors must be considered:

1. the virtual fixture radius must prevent the proxy from moving in the x direction more than would bring the tool-tip to d_{min} within the surface.

2. the proxy must move in the y direction enough so that the seashell sensor can detect another point which is at least f_h away from the originally sensed point.

These conditions are shown in Fig 4.4(c) before inserting a newly sensed point (green). Another lower bound on r_v is thus found as

$$r_v \geq \left| \sqrt{{d_{min}}^2 + (\epsilon_y + f_h)^2} \right| \tag{4.5}$$

With the addition of tracking error in the x direction, equation (5) may be extended to equation (6) as

$$r_v \geq \left| \sqrt{(d_{min} - \epsilon_x)^2 + (\epsilon_y + f_h)^2} \right| \tag{4.6}$$

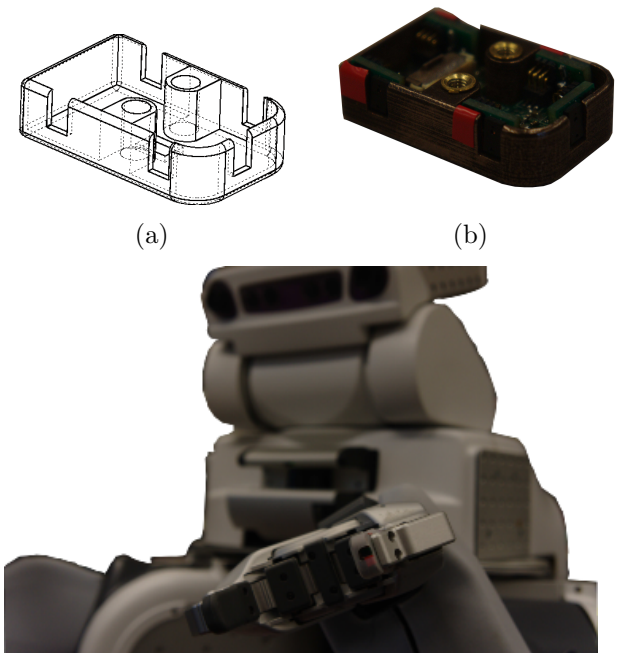
Now consider the task of allowing densely sensed points on the surface at spacing interval of f_h. Suppose that two points, spaced $2f_h$ apart have already been sensed on the surface, as shown in Fig 4.4(d). The aim is to sense a point in between these two at a spacing of f_h, as shown in green. From this, an upper bound on r_v is found as

$$r_v \leq \sqrt{{d_{max}}^2 + {f_h}^2} \tag{4.7}$$

In application, a maximum spacing of $\sup[0, 2f_h)$ is possible. With these bounds, forbidden region virtual fixtures were implemented with the PR2 bilateral teleoperation setup. The pretouch sensor was telemanevuered towards an unseen object to explore. As new points are observed, haptic virtual fixtures are built upon newly acquired data as well as the original point cloud to aid in avoiding collisions.

4.2 Fusing Non-Geometric Tracking with Depth Data

Now consider scenarios of teleoperation in dynamic environments in which the objects to be manipulated are moving, and vision may be degraded. Example motivating application scenarios include (1) robotic surgery, in which vision may be limited inside the body, and manipulation targets may move due to breathing, heart beats, and other biological motion, and (2) underwater robotic telemanipulation, in which visual sensing is degraded due to murky water, and objects may move due to currents. In these applications, target/environment geometry must be well understood prior to interaction, but contact should not be used as an information gathering mechanism (because of the potential for damage or unwanted target displacement), and long range visual sensors cannot be relied upon.



(a) (b) (c)

Figure 4.5: The utilized optical pretouch sensor. (a) Computer generated prototype of housing. (b) Sensor casing and circuitry. It is packaged into a small form factor. (c) Sensor mounted on the robot's left finger.

In contrast to the previous implementation, this work relaxes the assumption of static interaction targets, a common simplification in robotics that is not valid in less structured settings [59]. Again, various forms of pretouch sensing are highly maneuverable and are robust to poor visibility, and haptic virtual fixtures can provide intelligently informed, non-visual cues in exploration. The use of multiple pretouch sensors and sensor modalities can reduce uncertainty and increase understanding of the environment and thus the ability to safely interact with it. Here a array of pretouch sensors based on optical time of flight measurement is used, see Figure 4.5. The sensors detects and ranges target objects prior to contact; information from these sensors is presented to the user via haptic virtual fixtures. This combination of techniques allows the teleoperator to "feel" the moving object without actually creating contact between the robot and the object/environment. Thus it provides the perceptual benefits of touch interaction to the operator, without relying on the negative consequences of the robot actually contacting unknown geometrical structures.

In addition to the use of pretouch sensing, sensor-driven feedback pathways are used to intelligently inform the human operator. The haptic pathway is one of particular interest due to associated low reaction times [60] as well as intuitive nature when coupled with user motion input.

Problem Description

The goal of this work is to integrate compact pretouch sensing, object tracking and haptic feedback into teleoperative refinement of a moving object to simulate real-world scenarios. In some telerobotic applications, the robotic task can involve first identifying and recognizing the precise geometries of an object with which to interact. Sometimes it is sufficient to use modern range sensors, which can build a baseline geometry map of the remote scene in real-time. There are drawbacks, however. As previously discussed, the objects of interest may be poorly observed. This can be due to several factors, including occlusions, optical properties, glancing incidence angles and other factors [1,61]. In other scenarios, such sensors

are not suitable for the task — the remote environment may completely interfere with the sensor's mode of operation (e.g. IR sensors outdoors or in highly radioactive settings), or the environment may have physical or biological constraints. One cost of high-bandwidth geometry information is form factor, making such sensors difficult to utilize in confined spaces. Biological systems are also oftentimes spatially compact and furthermore are sensitive to the material and chemical properties of foreign bodies. A more compact sensor solution is needed.

Pretouch Sensing

Fingertip sensors are typically highly maneuverable, allowing them to explore areas that would be occluded in the relatively static view frames of long range depth sensors or cameras. Furthermore, by collecting data in closer proximity to the area or object of interest, fingertip sensors can often yield data that is more precise than sensors that measure from afar. Tactile sensors can provide precise information about object pose. In [62] and [54], tactile measurements were used to estimate the 6 degree of freedom (DOF) pose of an object. Both works described and utilized the Scaling Series algorithm as a means to integrate tactile measurements with an object model for pose estimation. One problem with this approach is that it is unlikely for a robot to have a model of the object *a priori*. In general, tactile sensing necessarily requires contact, which could displace the object.

The "pretouch" modality of sensing occurs at a distance scale intermediate to that of long-range depth sensors and tactile sensors. Many types of pretouch sensing have been previously explored, such as electric field, optical, and acoustic. Electric field sensing, however, only works well for objects that are highly capacitive or possess a dielectric constant significantly different from that of air. In [63], Mayton et.al demonstrated that servoing an arm and fingers equipped with electric field sensors can successfully gather geometric information about an object.

An optical sensor for the purpose of grasping, surface classification and slip detection was developed in [64]. In [56], optical proximity measurements were used to estimate object pose from a probabilistic model. Optical methods are affected by surface properties of the object and ambient lighting, and can completely fail when trying to sense highly specular or transparent objects. Jiang and Smith presented the first fingertip-mounted acoustic sensor in [65] and detailed a framework for using it and other pretouch sensors to perform exploration in [57]. Although acoustic sensing is well-suited for many of the failure cases of electric field and optical sensing, it does not detect open foams, rough fabrics, and fur well.

Sensor Based Exploration

Environment perception is crucial for most dexterous tasks — a model of the real-world must first be constructed, and oftentimes this model is constructed from information gathered from sensors. Saxena et al. used a model-based approach, where fixed objects and obstacles were partially identified and then fitted to a known model [66]. This group, along with [67] and [68], use a library of 3D models for actual interaction.

Oftentimes, *a priori* models are not available, especially in the case of new environments and objects. As stated by Aleotti et al. without actively exploring the environment, a robot can fail to build a complete 3D model because of occlusion [4]. An active exploration scheme, which utilizes maneuverable sensors, can be used to populate and refine previously unseen surfaces. In [3], an RGB-D sensor was mounted to the gripper of the robot. This information was used to maintain and update a voxel-representation of the environment which was actively explored in occluded regions. Using this active exploration approach, grasping task speed was increased and success rates were more than tripled. Similarly, in [4], a laser-range finder was mounted to the robot end-effector. Multiple viewpoints were stitched together in order to build a 3D model of an object for grasp planning, and different methods of 3D registration were compared.

Haptic Virtual Fixtures

Now consider the modifications to the haptic feedback in this implementation. Note that the haptic feedback for this task needs to now help accomplish two primary goals:

1. Guide the operator to the unobserved target

2. Prevent collisions with obstacles and the target

In order to achieve the former, a guidance virtual fixture in the form of a distance proportional attractive radius can be implemented (global attractor). This is in contrast to several path following approaches which restrict user motion along a predefined path or provide virtual force feedback in directions orthogonal to the desired path [30]. To achieve the latter, forbidden region virtual fixtures can be constructed with 3DOF proxy-method haptic rendering, as demonstrated by [39]. These forbidden regions will prevent the robot from coming within a given radius of any sensed geometry point.

While effective real-time implementations of these algorithms exist, the challenge here is twofold: firstly determining the proper parameters for the separate virtual fixtures, and secondly to resolve the two haptic cues. First consider setting up the virtual fixtures separately. To that end, first define the following variables

r_{fr}	forbidden region radius
r_g	no-guidance radius
r_o	anticipated target boundary radius
$\vec{f}_G$	guidance force vector
$\vec{f}_{fr}$	forbidden region force vector
$\vec{f}_{\text{net}}$	calculated force feedback

The determination of valid ranges for r_{fr} are the same from the previous implementation. From this, $\vec{f}_{fr}$ can be calculated and proxy motion determined, as per [51]. Examine first the potential forces generated via the target geometries and location. This aids in determining

the attractive guidance radius. On one hand, the operator should be encouraged to move toward the object if too far away. On the other, when the operator is within sensing range of the moving target, the attractive force should not produce a collision. The goal then is to determine a radius from the object at which guidance force starts and stops. With that said, bounds for both safe and efficient target refinement are sought. The following analyses are performed with a two-dimensional representation, but conclusions and implications can easily be extended to three dimensions.

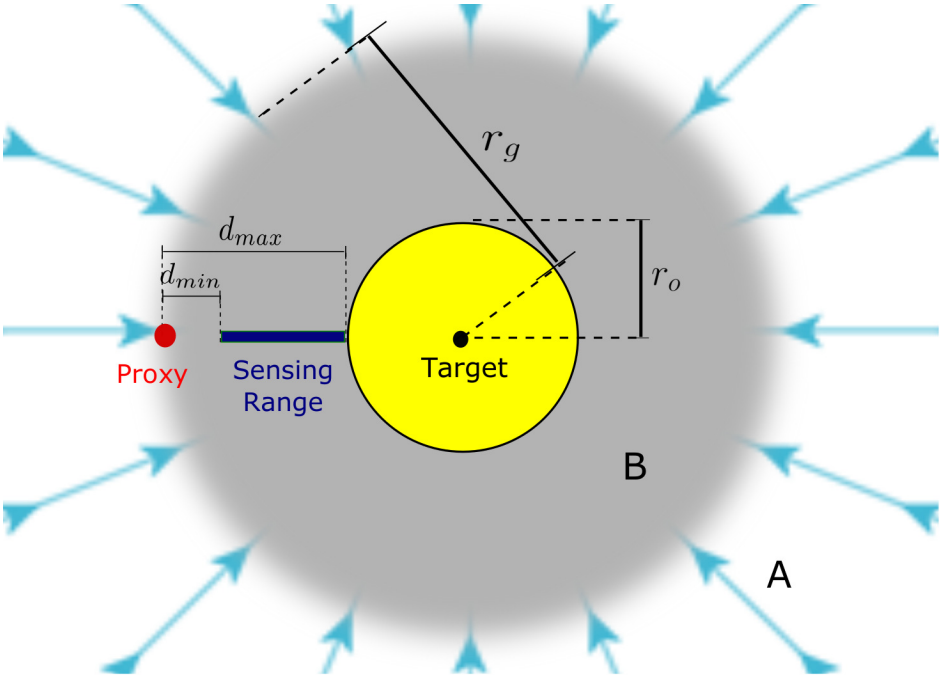


Figure 4.6: Basic end effector sensor exploration as the robot end effector approaches a surface. The robot end effector follows the location of the proxy which is attracted to within r_g of the target center, at which point the target is within sensing range. A proxy in region A would experience an attractive force towards the target, while one in B would receive no guidance feedback since it would be deemed to be within sensing range of the target.

First, consider the goal of maintaining the robot end effector within effective sensing range, and begin with the setup shown in Figure 4.6. From this figure, it is immediately clear that the attractive force must bring the proxy/end effector to within d_{max} of r_o of the target point. This can be expressed simply as

$$r_g \leq r_o + d_{max} \tag{4.8}$$

This will ensure that the operator and robot will be guided to within sensing distance of object surface. Note that the end effector may be configured in a way such that none of the pretouch sensors within the sensor array are facing exactly normal to the anticipated target surface. This upper bound on r_g encompasses all other cases.

Similarly, a maximum lower bound on r_g to enable the end effector to detect the surface at r_o is easily found as

$$r_g \geq r_o + d_{min} \tag{4.9}$$

This bound does not ensure safe guidance force radius; rather it is the smallest radius at which the robot may be able to observe the surface at r_o. In practice, the geometry and arrangement of the sensor array with respect to the proxy and r_o are needed to determine a lower bound on r_g to ensure safe guidance.

Now suppose that r_{fr} is determined. Let it be depicted as a positive scalar $r_{fr} \in \mathbb{R}^+$ such that

$$d_{max} \geq r_{fr} \geq d_{min} \tag{4.10}$$

Observe that this radius then helps define an effective value of r_g. This can be shown by assuming that a point has been sensed on the object surface, and consider a radius of

$$r_i = r_o + r_{fr}$$

as is shown in Figure 4.7.

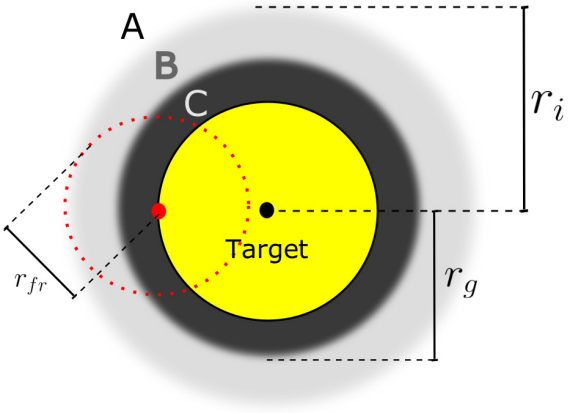


Figure 4.7: $r_g \leq r_i$ (when they are equal, region B does not exist).

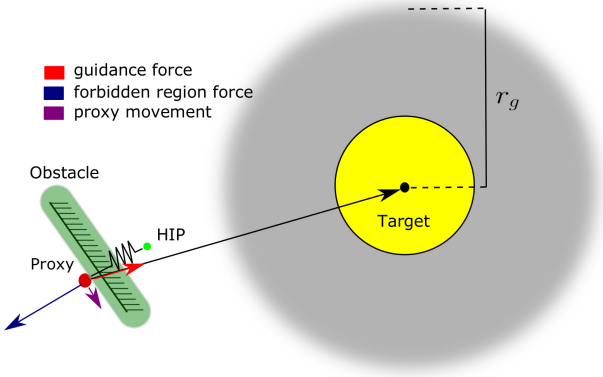


Figure 4.8: Obstacle with forbidden region virtual fixture obstructing guidance virtual fixture.

In Figure 4.7, region C corresponds to forbidden region feedback only, A represents only guidance feedback, and in region B, little to no net force is experienced due to virtual fixtures associated with the target. When inequalities 4.8, 2 and 3 are satisfied, force feedback from virtual fixtures associated with the target is as described above. Now consider forbidden region virtual fixtures due to obstacles. Forbidden region virtual fixtures from objects described as point cloud geometries are amenable to monolithic processing [39]. Consider resolving forbidden region force feedback from obstacles with haptic guidance towards the target. Suppose an obstacle obstructs a direct path to the target, as shown in Figure 4.8.

The danger occurs if the HIP moves far enough away from the proxy such that once the proxy has a clear, direct path to the HIP, it moves a great distance in a short period of time. This is a problem since the robot may move too fast for the pretouch sensor to detect an obstacle in time. While limiting proxy or robot velocity may mitigate this problem, it can reduce responsiveness to moving objects or anything else that requires fast motion. Instead, the proposed method can overcome the problem by accelerating the rate of navigation around obstacles as well as limiting the haptic guidance force to magnitude M. (M should be chosen such that the majority of force feedback is due to forbidden regions when the HIP

is a certain distance from the proxy). This is outlined as pseudo-code here.

Algorithm 1 Guidance Force Component

for each haptic update **do**
  Listen for latest published target frame
  Compute translation vector $\vec{t}_T^{\mathcal{P}}$ proxy to target
  if $\left|\vec{t}_T^{\mathcal{P}}\right| > r_g$ **then**
    Let $\vec{f}_G = k_G \vec{t}_T^{\mathcal{P}} \left(1 - \frac{r_g}{|\vec{t}_T^{\mathcal{P}}|}\right)$
    with fixed $k_G \in \mathbb{R}^+$
  else
    Let $\vec{f}_G = \vec{0}$
  end if
  if $|\vec{f}_G| \geq M$ **then**
    Set $\vec{f}_G = \vec{f}_G \frac{M}{|\vec{f}_G|}$
  end if
end for

The initial guidance force vector is thus acquired. Next, the initial force vector, $\vec{f}_{fr}$, generated from the forbidden region virtual fixtures can be obtained as shown in [39]. With these force vectors, the problem then becomes resolving the two signals to determine $\vec{f}_{\text{net}}$.

Algorithm 2 Net Force Determination

for each haptic message **do**
  Compute a plane Pl normal to $\vec{f}_G$
  Project $\vec{f}_{fr}$ onto Pl, call it $\vec{f}_{Pl}$
  $\vec{f}_{\text{net}} = \vec{f}_{fr} + \vec{f}_G + \vec{f}_{Pl}$
  Publish $\vec{f}_{\text{net}}$ to haptic device
end for

The effect of this method is twofold:

1. The user is guided towards the target as desired in free motion - when in conflict with

forbidden region, the forbidden region effects are dominant.

2. Proxy movement is expedited along convex surfaces towards the target — the HIP is encouraged to move around obstacles obstructing a clear path to the target.

The proxy in this method is still susceptible to concave surfaces or crevices in which the proxy may get stuck. With a reduced guidance force, the user need only overcome a slight force if an alternative route is readily apparent.

Target Tracking and Occupancy Grid

Methods to achieve object tracking can take advantage of many different types of information. Vision systems such as the Kinect can leverage geometric and color information to segment an object from the background, and then track it using the resultant reference model. However, there are many situations in which objects are tracked without real-time geometric or color information because it is either not available or not as robust as other methods. For example, in disaster situations, heat may be used to track humans or objects when rubble or inclement conditions make vision difficult or impossible. Also, radioactive tracers can be ingested in order to highlight certain types of tissues in the body. These are two instances where cameras may be too large, static, slow, or occluded to recover additional geometric information, but small, highly maneuverable proximity sensors could quickly build a map instead. In this work, an augmented reality (AR) tag indicates an area of interest with little to no prior geometric information.

Voxel grid representations of space efficiently represent occupied regions and are user friendly. In this work, Octomap [69], an open source voxel grid that is based on the OcTree data structure, was used. Deeper levels in the OcTree represent finer-grained regions of space, and nodes are added as space is explored and designated as either occupied or unoccupied. An OcTree with n nodes can be searched with $O(\log\ n)$ operations, and minimizes space by only explicitly representing areas that have been explored. Furthermore, a voxel grid

presents the user with a clear indication of whether a region is occupied, unoccupied, or unknown. These states are updated with any incoming sensor data from the pretouch sensors. If the user were instead viewing a point cloud, he or she may spend an excessive amount of time exploring a region, and data storage can become redundant.

Chapter 5

HAPTIC GUIDANCE FOR TELEMANIPULATION

The past two implementations demonstrated methods for fusing disparate haptic virtual fixtures or sensor types to achieve working results for telerobotic exploration tasks involving 3D geometric information. These experiments provide concrete examples of the performance benefits of the haptic virtual fixtures in terms of exploring otherwise unobserved data, but no physical interaction with objects occurred. Haptic feedback is amenable to aiding in guiding operator motion, but since objects collide and are displaced by the teleoperation, additional feedback may be necessary. In other words, a teleoperator ideally has feedback to monitor and evaluate the physical interaction process. This can be achieved via force sensors and haptic feedback, however limited fidelity, conflicting haptic signals, cost and compatibility issues are limitations in these methods [70]. Instead, enhanced real-time visual feedback can assist in verification during telemanipulation, and haptic guidance can potentially assist the operator even further.

In this implementation, a RGB-Depth (RGB-D) camera is used to gather color and geometric data of a remote scene. From this information, the operator is presented with a monocular color video stream, an enhanced 3D-mapping voxel representation of the remote scene, or the ability to place a haptic guidance virtual fixture to help complete a known telemanipulation task. The efficacy of this approach of task-specific guidance fixtures as well as the different visual feedback modes is then validated experimentally through a user study, and the different visual and haptic feedback modes are comparatively evaluated on the basis of objective and subjective metrics.

5.1 Introduction

In teleoperation involving high-value or delicate structures, it is imperative that the operator is able to perceive as much information about the remote environment as possible during physical interaction. This will aid in collision avoidance and monitoring of objects and task flow. The information should be perceived in a way that is as intuitive as possible, so that the operator's focus is not distracted from the remote task at hand. The standard feedback modality in many teleoperated schemes involves streaming RGB video. However, a combination of feedback quantities beyond simple monocular vision can assist the operator, for example by overlaying sensory information over data captured from the remote scene using haptic virtual fixtures [71]. This is particularly useful when the teleoperated task is known in advance. With an effective operator interface, the user can make the best decisions to efficiently complete the task with minimal physical and mental effort. This study aims to explore the effects of user-placed haptic guidance virtual fixtures and 3D-mapping methods for a telemanipulation interface. In particular, the following teleoperation user interface feedback types were examined:

1. Visual feedback
    - 2D monocular RGB
    - 3D voxel representation
2. Task-specific haptic guidance

This work investigates these feedback modes in a bilateral telemanipulation task through user-study experiments. Insight about their effects on both quantitative and qualitative performance factors, such as completion time and operator focus, is gathered, and results are indeed encouraging to the use of 3D mapping methods and haptic virtual fixtures for telemanipulation.

System Description

The setup for this project includes a bilateral teleoperation arrangement, in which the remote robot includes a KUKA youBot robot with a Primesense Carmine RGB-D camera. The youBot has an omnidirectional base and a 5 degree-of-freedom (DOF) manipulator. In this implementation, these joints are controlled by National Instruments' Compact RIO real-time controller, and commands back to the master console are transmitted via an Asus AC router. These features are shown in Figure 5.1.

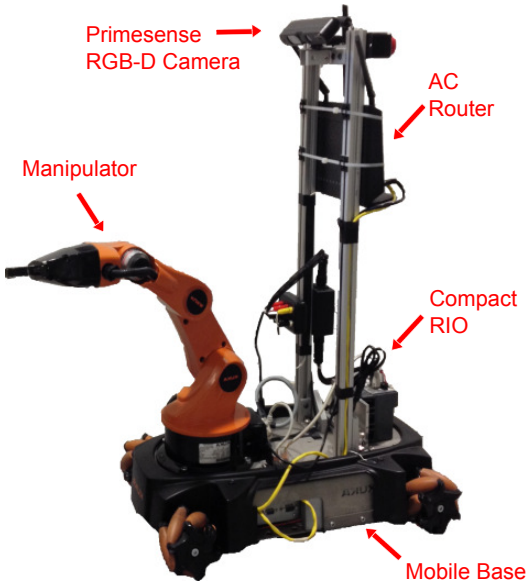


Figure 5.1: Teleoperated platform based on KUKA youBot.

On the master console station, the teleoperator manipulates a haptic device, the Sensable ® PHANToM™Omni. This device sends 3DOF position commands to control the end effector location of the youBot and it also receives and displays 3DOF haptic force feedback commands from the virtual fixture software. In addition, the user is presented with visual feedback on a LCD monitor. The teleoperator's goal is inspired from a search and rescue perspective, and is to manipulate the robot arm to turn a gas valve. The master console setup is illustrated above in Figure 5.2.

Figure 5.2: Teleoperation master console station. Includes 3DOF haptic device as well as a LCD monitor to display visual feedback.

5.2 Background

5.2.1 Visual Feedback in Telerobotics

Inexpensive commodity RGB-D cameras have provided non-contact means of collecting geometrical information of an observed environment. More specifically, point cloud data are obtained in real-time and provide boundary information about a physical scene. It has been shown that providing the teleoperator with depth information in the form of a real-time point cloud for certain navigation tasks can improve performance when compared to monocular RGB-streams [7]. However, with depth information, several factors can deteriorate the quality of feedback and increase confusion to the viewer. For example, physically interacting with the environment can result in occlusions due to the robot manipulator itself. Sensor resolution may be an issue when resolving smaller manipulation targets — the visual data may only provide a general localization of the object. Furthermore, depending on material properties, depth information may be noisy or arbitrary (e.g. transparent materials, glancing angles, or light absorbent materials) [1, 44] . Other variables such as measurement distance have also been shown to affect measurement noise and density [72]. Because of this variability, it is not certain that 3D-mapping techniques can improve operator performance in telemanipulation tasks.

5.2.2 Haptic Feedback

A haptic guidance virtual fixture pushes, prods or otherwise guides the operator's hand in a desired direction. This is useful for maintaining a predefined trajectory [5, 30, 39, 73–75]. In the case where depth perception is difficult, avoiding contacts and maintaining a safe, desired path can be assisted with such virtual fixtures. Moreover, since the virtual environment and force feedback are calculated in software, the actual robot end effector can be locked out of deviating from the desired path while a guiding force is applied to the user. In [65], a flexible guidance fixture was demonstrated where computer vision was used to identify obstacles obstructing a predefined virtual guidance trajectory. A modified trajectory was calculated that

avoided obstacles. The above types of guidance fixtures deal with fixed, predefined paths. In the case of telemanipulation in an unknown environment, while the task may be predefined, its ideal configuration in the remote location is difficult to determine. However, when enough information about the physical task space is obtained, it is feasible and desirable for the teleoperator to place the desired trajectory.

Wang et al. [6] showed an increase in user performance from the combination of both the visual and haptic display of a guidance virtual fixture in a computer simulation, i.e. the user could see the desired trajectory as well as feel guidance forces. The guidance virtual fixture was generated and placed by the user using a computer mouse on a virtual surface [6]. While this method limited the user-defined virtual guidance fixture to the face of an object, it is extendable to trajectories in three-dimensional space.

5.3 Experimental Goals

To evaluate the effectiveness of the described feedback modes and user-placed guidance fixtures, teleoperator performances during the valve turn task are compared under the following different user feedback conditions:

1. Visual only, monocular RGB stream (**R**)

2. Visual only, 3D-mapping voxel method (**V**)

3. Visual and haptic guidance virtual fixture (**VF**)

Scenario 1 (**R**) represents a particularly simple baseline case, RGB streaming video, which is still employed in certain teleoperated tasks.

Scenario 2 (**V**) provides a baseline for 3D visual representation. The user is able to rotate and translate his or her view within the 3D representation as well. This representation includes a 3D voxel map of a volume enclosing the youBot's task space, which is updated

with RGB-D sensor information based on a Bayesian statistical model to determine binary occupancy state. Similar methods were explored in [44, 69], however in this study, the voxel allocation and update is hardware accelerated to ensure real-time acquisition and fast response to motion while explicit data is retained.

Scenario 3 (**VF**) provides the operator with the option of haptic feedback in the form of a guidance virtual fixture of proper shape for the task in addition to the 3D-mapping method from Scenario 2. This fixture will prevent the operator and robot from deviating from a path known to successfully complete the valve turn and avoid undesired contacts, and overcome confusion caused by occlusion of the valve itself due to the manipulator itself. The trajectory is a series of finely sampled, ordered points. Because the environment is rendered as a 3D voxel grid, it is simple and quick for the operator to place the visualized desired trajectory properly.

It is of interest to determine whether or not, in this telemanipulation task, 3D-mapping techniques will improve operator performance, even if displayed to the user with 2D visual display. Furthermore, the efficacy of a user-placed guidance trajectory is desired. A comparison is sought between 2D monocular RGB stream (**R**), 3D voxel mapping techniques (**V**), and visual+haptic feedback (**VF**). Two questions are being evaluated:

1. Does the addition of 3D-mapping techniques improve user performance, decrease workload or increase awareness?
2. Do manually-placed haptic virtual fixtures provide additional improvements?

Mast et al. [7] determined that the usefulness of voxel based 3D-mapping in navigation varied between environments. Here, the utility of such 3D representations when applied to a manipulation task is explored. In such a task, 3D information may not be conveyed in a useful way, and could in fact be detrimental when dealing with movement and small objects. Additionally, occlusions could result in lack of information and heightened interpretation

effort, while a monocular RGB stream is intuitive and familiar to most users.

It has been established that haptic virtual fixtures can be useful for object avoidance and following a predefined path [5]. In this work, the user is given the ability manually set the virtual fixture. It is therefore imperative that the user be provided with the 3D-mapping representation to properly place the virtual fixture, as well as monitor the actual valve turn during operation. Depending on several spatial and sensor-limited factors (e.g. the user may have a difficult time placing the virtual fixture), it is not clear whether such a feedback option is beneficial.

Because of the variability of the above factors, the nature of this experiment is exploratory. This work may lead to interesting findings in the suitability of 3D- mapping methods and manually placed virtual fixtures for telemanipulation feedback.

5.4 Experimental Methods

5.4.1 Feedback Modalities

Recall that three main feedback modes are compared. Each are described here.

*Monocular RGB, (**R**)*

Monocular RGB feedback, (**R**) is a simple and widely available baseline case. For this feedback mode, the user was presented only with streaming RGB video feedback displayed via the LCD monitor. Figure 5.3 shows a typical screenshot of the visual feedback from this mode. The RGB data was captured at an acquisition rate of 30 Hz, and

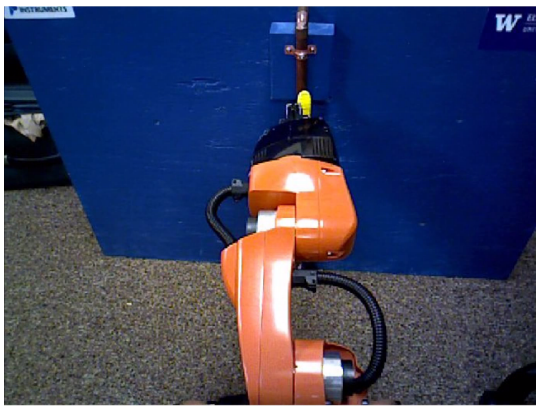


Figure 5.3: Monocular RGB visual feedback.

the visual feedback was rendered in OpenGL using components found in RViz.

*Voxel-based 3D-mapping, (**V**)*

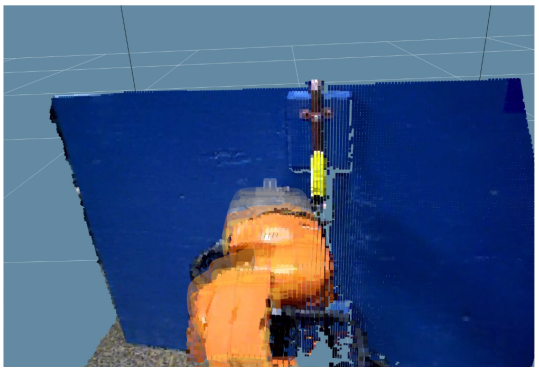


Figure 5.4: Voxel-based 3D-mapping Feedback

In the 3D-mapping mode (**V**), a voxelized cube with side length of one meter and resolution of 5mm was graphically rendered in front of the youBot, and is depicted in Figure 5.4. A simple Bayesian update method with heuristically tuned update parameters was used to determine voxel occupancy. In this way, occupancy state can be preserved even with occlusion.

In each RGB-D frame, for each voxel, project the voxel onto the depth image. The voxel is represented in the depth image with pixel location p, and depth value of v. In the measured depth image, p also has a camera measured depth representing real-world data, call it s. The two depths, v and s, represent the voxel depth and the sensed surface depth respectively. If $v \leq s$ (i.e. the voxel is closer than the surface sensed by the camera), determine the voxel occupancy state, O, via a Bayesian update rule. This can be summarized as pseudo code below:

Algorithm 3 Voxel Update Rule

for each RGB-D frame **do**
  for each voxel **do**
    Project onto depth image
    Obtain: p – pixel
      v – voxel depth
      s – measured depth at p
    if $(v \leq s)$ **then**
      Define observed occupancy weight, W
        $W = e^{-k(s-v)}$ k constant
      Calculate posterior occupancy weight, $\hat{W}$
        $\hat{W} = \max(i\hat{W} + jW, 1)$ i, j constant
    end if
    if $(\hat{W} \geq \tau)$ **then** τ constant threshold
      $O = 1 \implies$ voxel is occupied
    else
      $O = 0 \implies$ voxel is unoccupied
    end if
  end for
end for

In this way, the occupancy was updated every RGB-D frame while preserving occupancy states of now occluded voxels. i, j, k, τ were all heuristically tuned. The algorithm is highly parallelizable, and was hardware accelerated to ensure real-time acquisition and fast response to motion. In mode **V**, the user could view the occupancy grid from various angles using a computer mouse. The RGB-D data was captured at an acquisition rate of 30 Hz, and again the visual feedback was rendered in OpenGL using components found in RViz.

*Guidance Virtual Fixture, (**VF**)*

In the haptic virtual guidance fixture feedback mode (**VF**), the user was provided with the same 3D visual feedback described for mode **V**. In addition to this, the user was able to place a visualized path (a green colored arc) on the ball valve structure, as shown in Figure 5.7.

This path provided haptic feedback once placed, and ideally passed through occupied voxels representing objects for interaction. For the valve turn in particular, it was desired that the path pass through the voxels representing the handle of the ball valve.

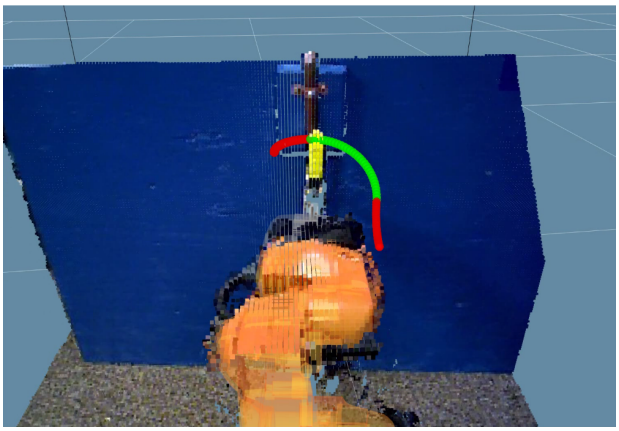


Figure 5.5: Manually-placed haptic guidance virtual fixture feedback. The red denotes entry and exits points of the trajectory, while the green signifies the valve-turning portion of the path.

In order to render the haptic feedback, the path is first sampled as a set of spatially ordered points. As the operator approaches the guidance path to within X of any sampled point, an attractive haptic well is generated around that point. The force profile of this haptic well is defined by a piecewise cubic polynomial (see eq 5.4.1), whose shape is shown here in Figure 5.6. The effect of this force profile is twofold. Firstly, the user can remove themselves from the guidance fixture by moving beyond X of the guidance point. Secondly, the user is strongly encouraged to stay within $\frac{X}{2}$ of the guidance point while receiving force feedback.

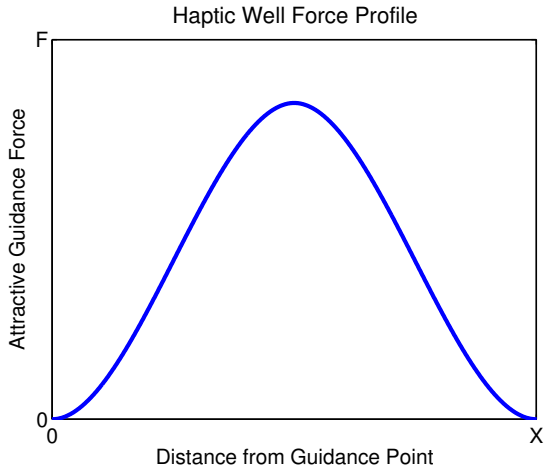


Figure 5.6: Haptic well force profile shape.

$$f(x) = \begin{cases} a_2x^2 + a_3x^3 & 0 \leq x < \frac{X}{2} \\ a_2(X-x)^2 + a_3(X-x)^3 & \frac{X}{2} \leq x \leq X \\ 0 & \text{else} \end{cases} \tag{5.1}$$

To move along the path, the current closest point and directly adjacent ordered points are considered. When the user moves and an adjacent point is now closer, the haptic well around the current point is attenuated while a new haptic well is enforced at the new closest point. This process is repeated on the subsequent points, guiding the user along the ordered points. If the user leaves the guidance fixture, the entire procedure is repeated.

5.4.2 Experimental Task

The operator is asked to complete a valve turn task. Such a task is motivated from a disaster recovery perspective. In the case of a gas leak during a natural disaster, teleoperation is attractive because it reduces risk to human responders. Moreover, a teleoperated device may be better designed to reach constrained physical scenarios than a human being. In this study, the valve to be turned consists of a ball valve structure with 90° dynamic range. The task can be broken down into two subtasks and turns:

Task A—turning the ball valve from the 12 o'clock position to the 3 o'clock position.

Task B—turning the ball valve from the 3 o'clock position to the 12 o'clock position.

The task is depicted in Fig 5.8, and it is with this setup that the user study for the project was conducted.

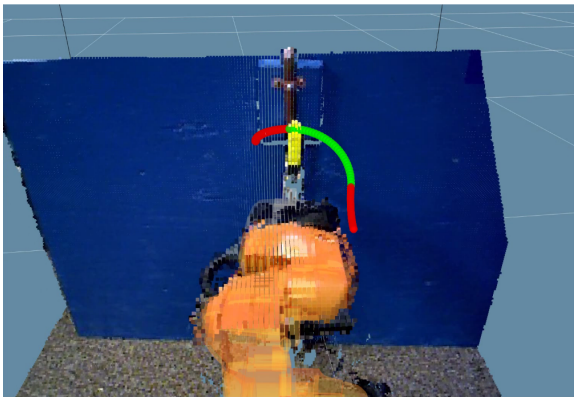


Figure 5.7: Manually-placed haptic guidance virtual fixture. Red denotes entry and exits points of the trajectory, while the green signifies the valve-turning portion.

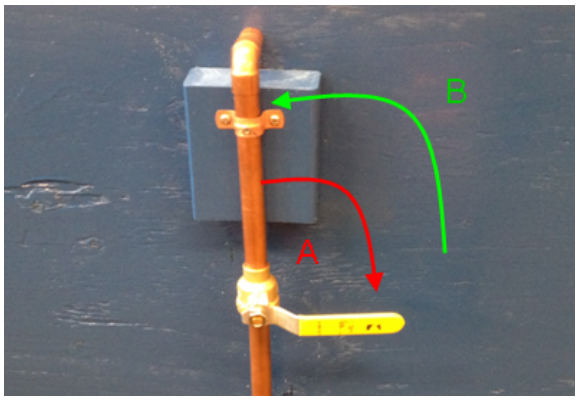


Figure 5.8: Teleoperation valve turn task. Task A (red) 12 o'clock to the 3 o'clock position, Task B (green) 3 o'clock to the 12 o'clock position

Chapter 6

TESTBED EVALUATION: HAPTIC VIRTUAL FIXTURES IN TELEROBOTICS

6.1 Sensor-Aided Teleoperated Grasping of Transparent Objects

The completion of a successful teleoperated grasp consists of two main components: gathering sufficient geometrical information of the grasp target, and the executed grasp itself. The experimental portion of this work focuses on the former, and is composed of two parts: (1) exploration of an occluded face of a flat cardboard box, (2) validation through exploration and grasp of transparent grasp target. Recall that in this setup, the operator is using the seashell effect pretouch sensor to explore an unseen object (e.g. due to transparency), and haptic virtual fixtures prevent collision during this task. The test scenario consists of an opaque box surface, and is amenable to repeatable quantitative analysis of the algorithms described in 4.1 because:

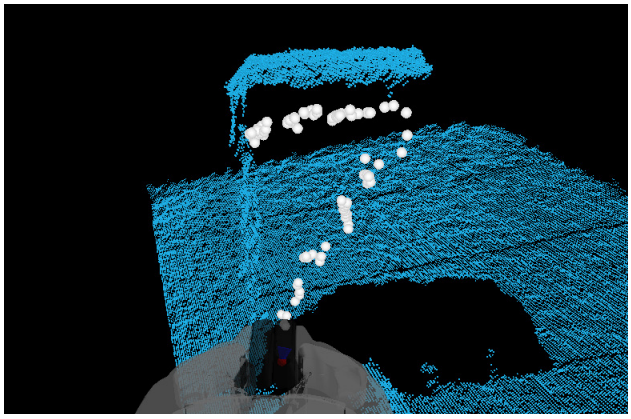


Figure 6.1: Desired exploration trajectory for unknown flat box surface

1. The unseen, occluded faces of a cardboard box simulate the unseen geometries of a transparent object.

2. The faces of the cardboard box which are successfully measured by the RGB-D camera provide:

   (a) a physical reference frame for repeatable exploration trajectories.

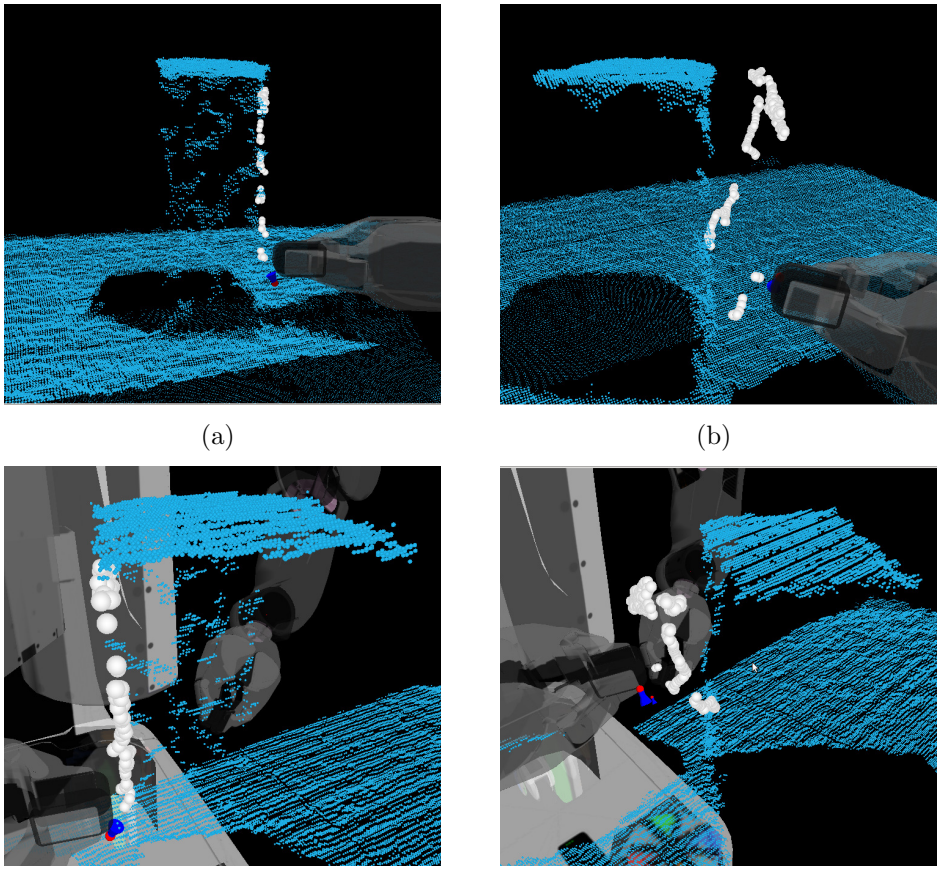
(b) a clear visualization of displacement during exploration.

Teleoperation experiments are consistently repeated and displacement is easily visualized with the opaque box. Point spacing and virtual fixture radius were chosen as described in 4.1, and haptic virtual fixtures were used to safely explore an occluded face of a cardboard box. In particular, the exploration was performed in a specific trajectory as shown in Fig 6.1. The teleoperator was asked to discover an unknown surface of the box while attempting to avoid contact by exploring along the top edge, and then returning to the bottom corner across the diagonal of the box face. During teleoperation, the user received visual feedback in the form of the virtual environment rendering (includes proxy location, HIP location, surface normal, PR2 model and registered point cloud) as well as an RGB image stream from the Kinect. When using haptic rendering, the user received force feedback via the Sensable ® PHANToM^{TM}Omni device according to the forbidden region virtual fixtures. Finally, safe exploration and grasp of a transparent grasp target are used to verify feasibility of this method.

6.1.1 Results

The teleoperated exploration task as described above was performed five times both with and without haptic virtual fixtures. During the trials, the points sensed by the seashell pretouch sensor were both rendered graphically and logged for post-trial quantitative analysis. At the end of each trial, a visual inspection of the point clouds was performed, and a sample of this is shown in Fig 6.2

The visual results provided a qualitative measure of performance. Ideally, the sensed point cloud should resemble a flat planar surface, as describes a static box surface. With haptic virtual fixtures, the robot end effector was prevented from colliding with the surface, and the teleoperator's hand was pushed away from the sensed surface. With visual feedback only, displacement and less structured measurements are expected due to the difficulty of teleoperating the end effector in the sub-centimeter range. This can be seen in Fig 6.2.



(a) (b)

(c) (d)

Figure 6.2: Seashell sensor point clouds (a) with haptic virtual fixture, (b) visual feedback only, (c) close-up with haptic virtual fixture, (d) close-up visual feedback only.

A more quantitative comparison was performed. In particular, a two dimensional plane was fit in a constrained least squares sense to each of the point clouds (if the object was not displaced, the seashell-sensed points should all lie on a planar surface). The enforced constraint ensured that the fitted plane was perpendicular to plane represented by the robot base z axis (box surface is approximately perpendicular to this plane). Figure 6.3 shows plane fitting results.

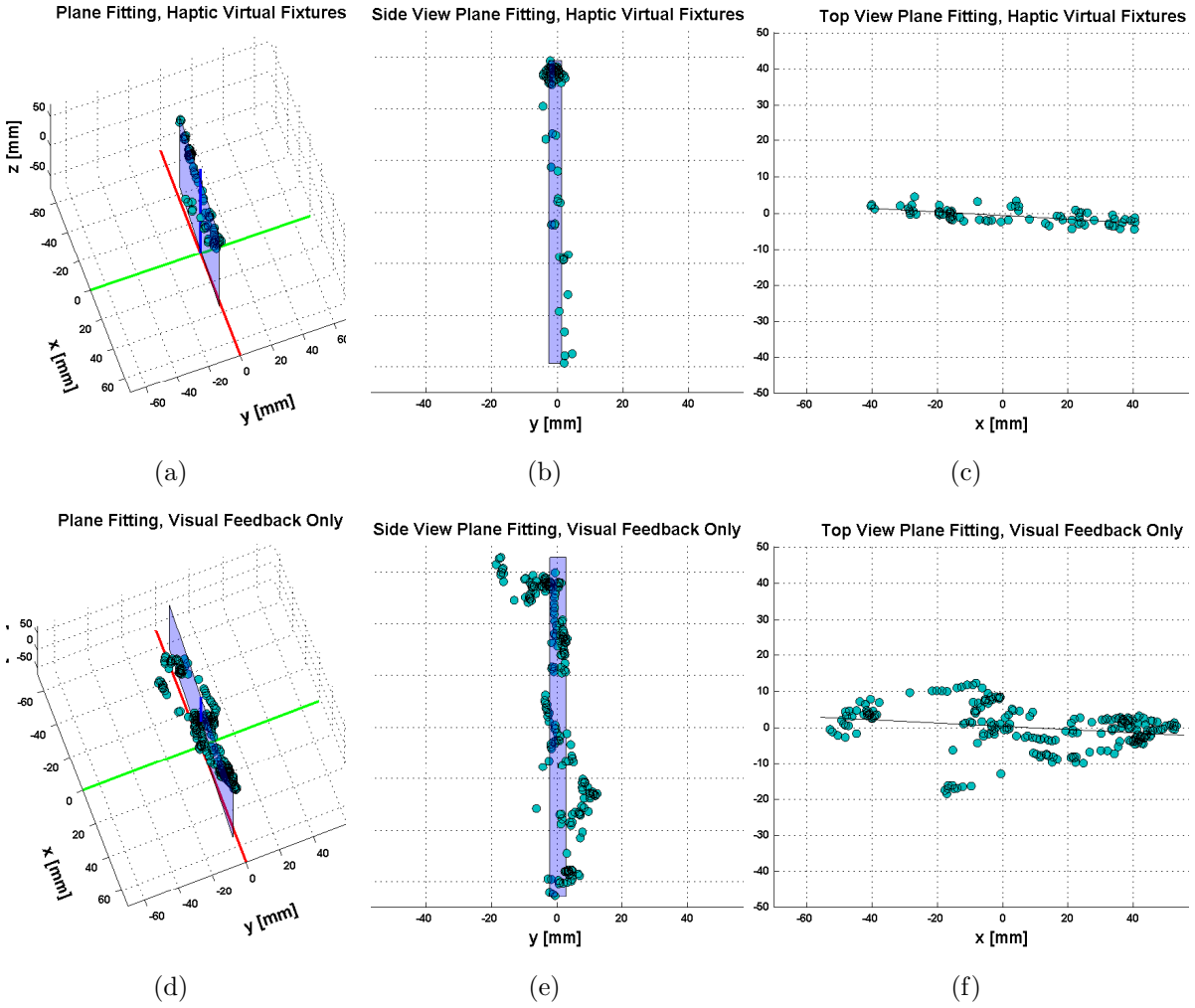


Figure 6.3: Plane fitting visual results {(a),(b),(c)} haptic virtual fixture, {(d),(e),(f)} visual feedback only.

The deviation from the fitted planes was calculated for each dataset. Additionally, the number of collisions, time to completion, as well as object displacement were measured for each test. Unless stated otherwise (i.e. preceded with 'total' or 'RMS'), the data in Table 6.1 represent the mean value across ten total trials.

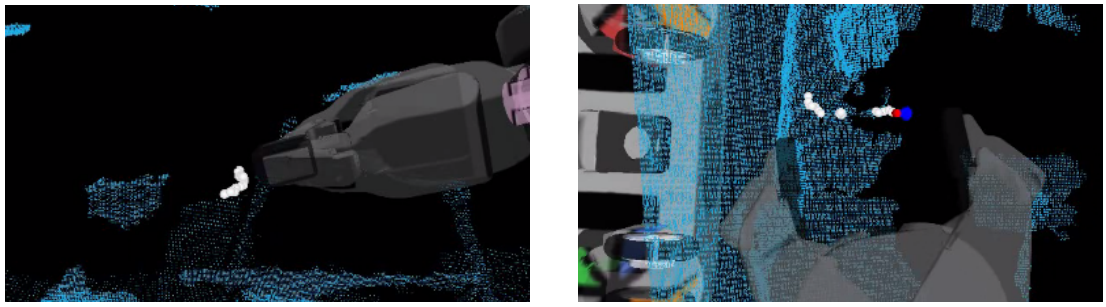
Table 6.1: Experimental results of the exploration tasks

	Haptic+Visual	**Visual Only**
Number of sensed points	295	1217
Task completion time [s]	185.4 ± 27.7	108.6 ± 17.3
Total collisions	1	71
Collisions	0.2± 0.2	14.2±1.6
Displacement [mm]	N/A	17.6± 7.1
Deviation from plane [mm]	1.468 ± 1.171	3.797 ± 3.033
RMS deviation from plane [mm]	1.878	4.860

The experimental results show that haptic virtual fixtures around pretouch sensed points can significantly reduce the number of collisions and object displacement, at the cost of less point density and longer completion time. This is an expected trade-off, since the haptic virtual fixtures restrict the user's motion (longer time), and limit the robot end effector motion near the surface (less points). This combination of techniques can be used in grasping tasks to improve the teleoperator's performance. It also allows for reaction to moving objects (e.g. to avoid unintended contact).

In addition to exploring the opaque box surface, the same methods were used to gather geometric information about a plastic, transparent bottle in a cluttered environment. This information allowed for intuitive and successful grasp. The main difference between this implementation and the experimental opaque box case was the lack of reference geometries, which were previously provided by faces of the box observed by the Kinect. The empty regions of the point cloud of the transparent object provided enough cues for initial approach

without collision. Once a point was sensed by the seashell sensor, the exploration proceeded similarly to the experimental implementation, with the addition of grasping the object following sensing. The sensing of the transparent bottle surface for teleoperated grasp is shown below in Fig 6.4.



(a) (b)

Figure 6.4: Transparent grasping, (a) exploring the transparent surface (b) newly sensed points provide critical geometries prior to successful grasp.

6.2 Fusing Non-Geometric Tracking with Depth Data

6.2.1 System Description

The setup includes a bilateral teleoperation architecture with a master console and telerobotic slave device interacting with a remote task environment. Software components were developed and executed in ROS.

Telerobotic Platform

The slave robotic platform is the PR2 from Willow Garage. The end effector of the PR2 was outfitted with a pretouch sensor array.

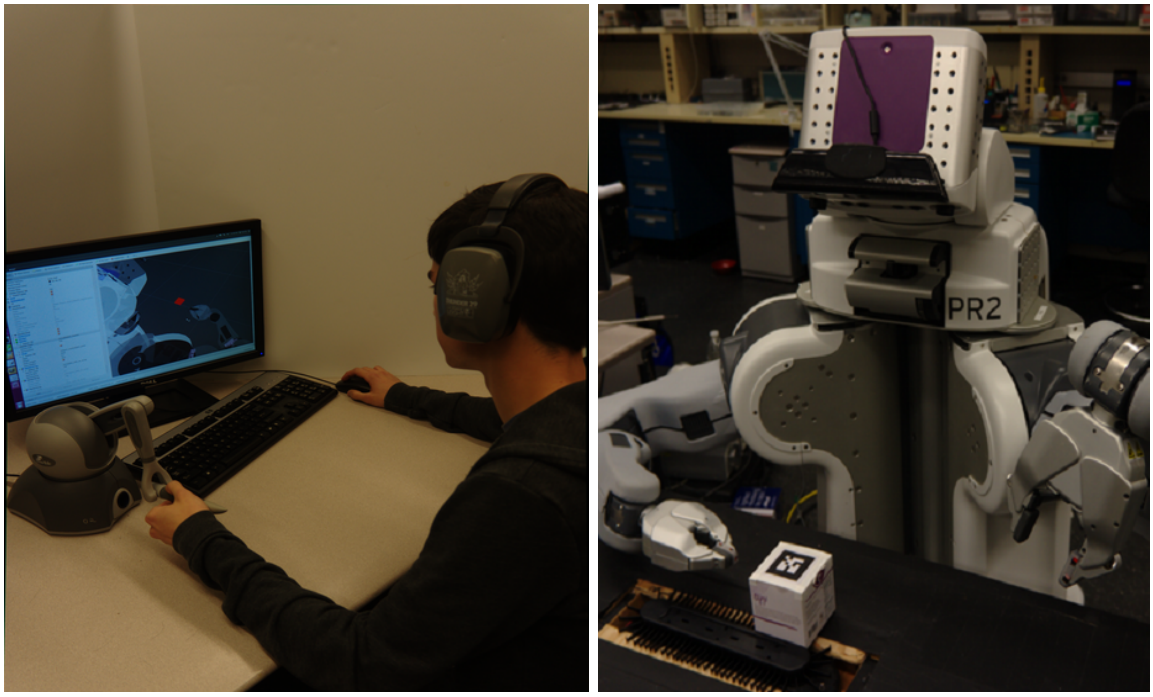
Pretouch Element

The robot is equipped with an optical pretouch sensor on one of its fingertips in order to facilitate exploration. It consists of six STMicroelectronics VL6180X [76] proximity sensing

modules, each of which can sense distance within 1 to 10 cm at a rate of 33 Hz. There are two sensing modules on each side of the finger, one at the front, and one on the pad of the finger. The sensor is completely integrated into the robot's gripper.

Master Console

The master console both presents feedback to the human operator as well as receives user input motion commands, as shown in Figure 6.5(a). Visual feedback (a voxel representation of the target object) is rendered on an LED monitor while haptic feedback is presented through a 3DOF haptic device, the Sensable ® PHANToM™Omni. The feedback reflects forces calculated to both prevent unwanted collisions (forbidden region virtual fixtures) as well as guide the user towards the exploration target (guidance virtual fixtures).



(a) (b)

Figure 6.5: The experimental setup. (a) The user operates a 6-DOF input device to control the robot. A location to search and explored occupied voxels will be displayed in RVIZ. (b) The robot carries out the teleoperator's commands as a conveyor belt changes the pose of the object.

Telerobotic Task

A controllable circular conveyor belt, as shown in Figure 6.5(b), was designed to enable granularity in movement speed for experiments. An object could be translated and rotated with the conveyor belt, which sits flush to a tabletop. With this setup, a moving exploration target will be refined with the pretouch optical sensors on the slave device.

6.2.2 Experiment Workflow

This work evaluates the effect of haptic feedback on operator performance in exploring a completely unseen object. Two sets of 10 one-minute object refinement trials were conducted, one with haptic feedback and one without. Visual feedback was provided by RVIZ. At the start of each trial, the user is shown the kinematic model of the robot and a marked location to be explored. This location is tracked using an AR tag, which simulates a location tracked through non-geometric means.

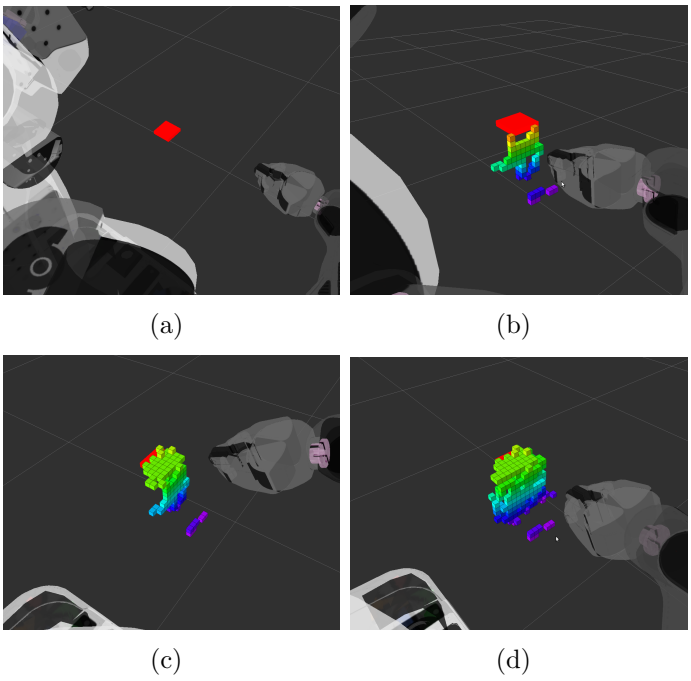


Figure 6.6: The marker and explored occupied voxels are shown to the user during the experiment. (a) Initially, the user is only shown the marker of the location to be explored. (b) Additional voxels are discovered as the user explores one side of the object. (c) Voxels on top of the object are explored. (d) Voxels on the front and back of the object are found as the user continues to explore until the end of the experiment.

In each trial, the object moves back and forth at a speed of 0.5 $\frac{\text{cm}}{\text{s}}$ in front of the robot, with pauses in motion rang-

ing from 0 to 4 seconds. During the trial the user attempts to explore an unseen box with a width, height and length of 9 cm, and any occupied voxels that he or she discovers are displayed and recorded. An example of the visual feedback that the user receives is illustrated in Figure 6.6. The distance between the end effector and the marked location was also recorded. Note that the red marker in Figure 6.6 does not represent any surface geometrical information about the unseen target. It reflects the general target location.

6.2.3 Results

Throughout the 10 trials with haptic feedback, the user explored more than twice the number of occupied voxels when compared to visual feedback only. In order to compare the fit of the discovered voxels to the ground truth, the iterative closest points algorithm was performed between the collected occupied voxel centers and the true voxel centers of the explored cube. In Figure 6.7(a) the discovered voxels across all 10 haptic feedback trials have been fitted to the ground truth model and plotted from an isometric perspective in red. Ground truth is plotted in blue. Figure 6.7(c) offers a side view of the same data. Figures 6.7(b) and 6.7(d) offer the corresponding information for the non-haptic feedback trials.

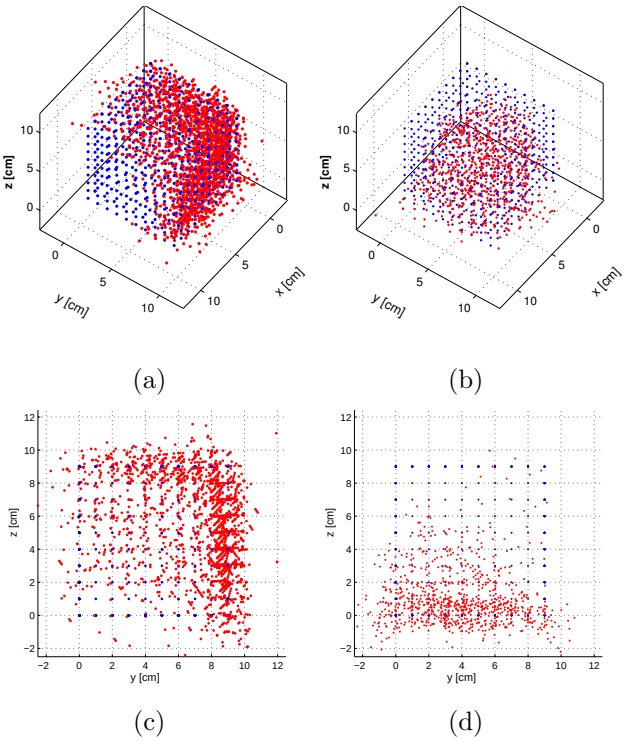


Figure 6.7: Cumulative discovered occupied voxels gathered across all trials in red with the true voxel centers in blue: (a) and (b) show an isometric view of the results with and without haptic virtual fixtures respectively. (c) and (d) present side views of the data shown in (a) and (b) respectively.

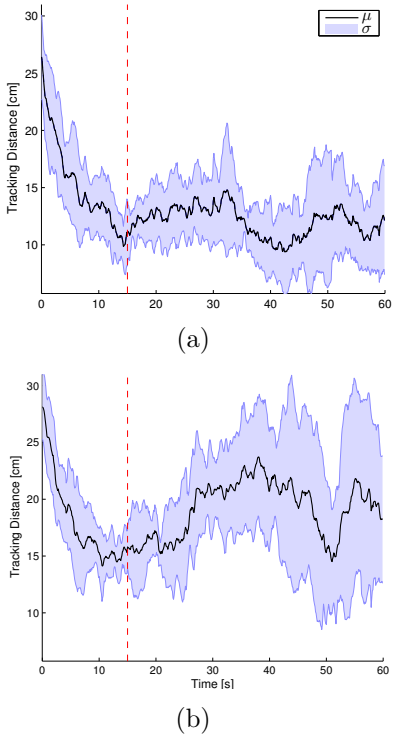


Figure 6.8: Distance between the end effector and the tracked object during the 10 trials. Blue envelope illustrates the standard deviation of the distance.

The experiment also showed that haptic feedback allowed the user to bring the end effector closer to the object. In Figure 6.8, the distance between the fingertip end effector and the center of the upper surface of the target box is plotted. Specifically, the average distance and standard deviation across all trials is plotted for each point in time. The red dashed line denotes the 15 second mark, the typical amount of time that the user spent initially approaching the object. Figure 6.8(a) shows the results with haptic feedback. For the trials that did not use haptic feedback, the corresponding plot is shown in Figure 6.8(b). Table 6.2 summarizes the findings of the experiment.

Table 6.2: Experimental results of the exploration tasks

	Marker Only	**Haptic Feedback**
Total Trials	10	10
Avg Sensed Voxels	88.90±21.46	192.5±24.38
Avg Distance [cm]	18.70±9.621	12.60± 6.121
Avg ICP RMS [cm]	0.9418 ±0.3396	0.7779 ± 8.801e-04
Total Collisions	7	3

6.3 Haptic Guidance for Telemanipulation

Subject Recruitment

In this study, recruitment was performed on the University of Washington campus and subjects consisted solely of undergraduate and graduate students. As described in section 5.1, a total of three test conditions exist. In this project, a between-user study was employed. 21 male subjects participated in this study (seven in each test group). Their age ranged from 18 to 35 years of age (mean age group **R**: 25.143; **V**: 23.000; **VF**: 26.143). Participants were chosen to be male to avoid any effects due to possible differences between human males and females in spatial problem solving as described by [77]. Each of the participants used computers at least 10 hours per week. In each group (seven participants total), six of the participants played less than two hours per week of video games, while exactly one participant played more than 10 hours per week (mean video game usage per week **R**: 2.214; **V**: 1.857; **VF**: 2.000). Lastly, all participants had or were pursuing a university degree.

6.3.1 Experimental Protocol

Metrics

In this project, both objective and subjective metrics were employed for comparison. In particular, objective performance metrics included:

- time to complete the valve turn task [s]
- path length of the end effector [mm]
- number of undesired collisions
- jerk of the end effector $\left[\frac{m}{s^3}\right]$

After the completion of the task, subjective measures were assessed via post task questionnaires. The following two subjective categories were measured:

- Perceived workload
- Situational awareness

Perceived workload was measured using the unweighted NASA Task Load Index (TLX), and situational awareness was gauged using the three-dimensional Situation Awareness Rating Technique (SART) scaled to be within (0, 120).

Procedure

The experiments were conducted in an office and the hallway corridor outside. The participant teleoperated from the master console within the office, while the simulated remote environment was in the hallway outside of view from the subject. In the hallway, the youBot and the valve structure were placed in the same location for each experiment. Prior to the experiment, the users were allowed to see the valve and robot position, and were further

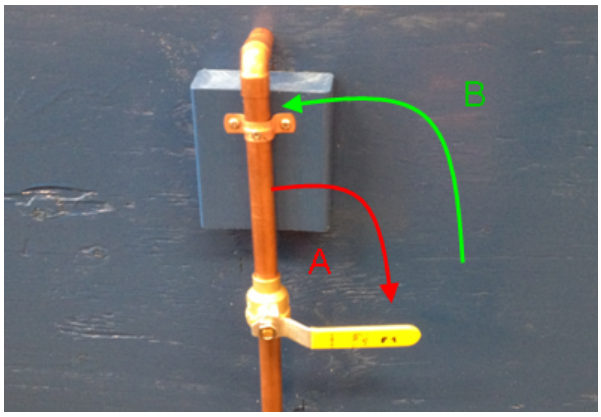
allowed to turn the valve manually to obtain a sense of the range of motion as well as the torque needed to turn the valve.

After viewing the valve structure and youBot, the subjects underwent a training period which lasted for 20 minutes or when the user was satisfied, whichever happened first (in all cases in this study, the user was satisfied with the training prior to the 20 minutes). The training session occurred with the youBot in the office space within view of the operator. For the training task, the operator was asked to knock over a set of stacked boxes. During the training session the user was only allowed to teleoperate with their given feedback mode only. For the monocular RGB mode (**R**), the user was presented with a 640x480 video stream of the manipulator in well lit conditions, for (**V**) a voxelized representation and for (**VF**) the operator received the voxel map visual feedback and could furthermore place a haptic guidance fixture. During practice and the experiment, data was acquired at 50Hz and visual feedback updated at 30Hz, the data acquisition rate of the Primesense Carmine camera. The haptic update rate was set at 1200 Hz to maintain realistic force feedback.

For each subject, once the experiment began, noise-isolating ear protection was placed on the participant's ears. Each subject was asked to perform ten tasks, and they performed in order:

- Task A five times
- Task B five times

These tasks are depicted by Fig 5.8, and shown again here.



Teleoperation valve turn task. Task A (red) 12 o'clock to the 3 o'clock position, Task B (green) 3 o'clock to the 12 o'clock position

In between trials, the robot was homed to a fixed starting configuration. The user was timed from movement from this home position until the valve was turned completely. In mode **VF**, the user needed to place the guidance virtual fixture during each trial, i.e. ten times per subject.

6.3.2 Results

Completion Time

Consider the mean completion time across the various feedback modes (**R**: monocular video stream, **V**: 3D-voxel mapping, **VF**: 3D-voxel mapping and guidance virtual fixture), as shown here in Fig 6.9 and Table 6.3. In these bar graphs, the error bars represent standard deviations within each mode.

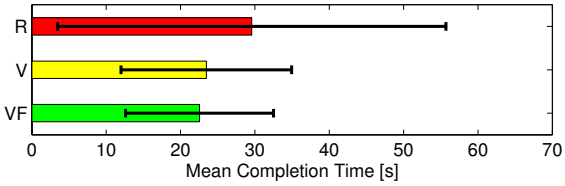


Figure 6.9: Mean Completion Time

	R	**V**	**VF**
Mean Completion Time [s]	29.573	23.468	22.547

Table 6.3: Mean Completion Time

	R-V	**R-VF**	**V-VF**
Total Time p**-value**	0.0757	0.0373	0.6129

*denotes significance

Table 6.4: Completion Time Statistics

Table 6.4 shows no significance observed between feedback modes for completion time. In all tables describing statistical significance, an asterisk * denotes statistical significance based on two-sample t-tests assuming roughly normal distributions. Significance was determined via the Holm-Bonferroni method, described later.

Now consider the mean number of unwanted collisions (this includes any contact not with the valve handle or contacts pushing the valve handle in the incorrect direction) across the various feedback modes, shown in Fig 6.10 and Table 6.5.

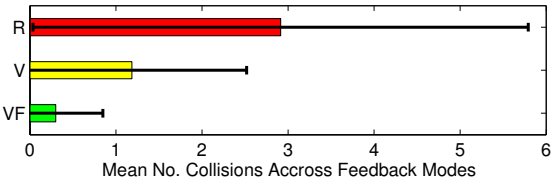


Figure 6.10: Mean Number of Collisions

	R	V	VF
Mean Number of Collisions	2.913	1.186	0.300

Table 6.5: Mean Number of Collisions

	R-V	R-VF	V-VF
Collisions p-value	1.118e-5*	8.459e-12*	9.110e-7*

*denotes significance

Table 6.6: Collisions Statistics

For the number of unwanted collisions, there was observed statistical significance between all the feedback modes **R**, **V** and **VF**, see Table 6.6.

Overall task path length was determined from recorded encoder values and forward kinematics. The collected data from mean path length across the three different feedback modes are shown in Fig 6.11 and Table 6.7. Statistical significance for path length was not found between the feedback modes, see Table 6.8.

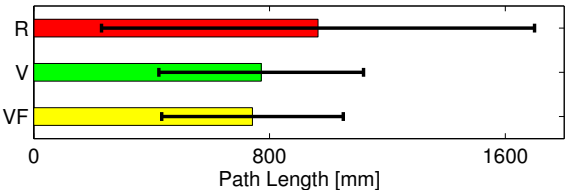


Figure 6.11: Mean Path Length

	R	V	VF
Path Length [mm]	964.68	771.81	742.06

Table 6.7: Mean Path Length

	R-V	R-VF	V-VF
Path Length p-value	0.0499	0.0212	0.5927

*denotes significance

Table 6.8: Path Length Statistics

Differentials of the sampled position were used to approximate jerk. A low-pass filter removed high-frequency components introduced through discrete differentiation. The mean values are displayed below in Fig 6.12 and Table 6.9.

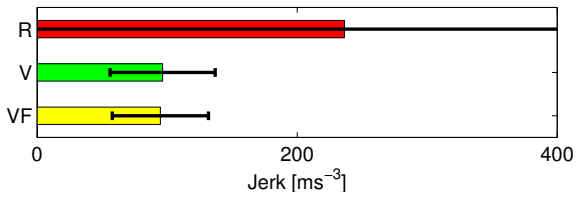


Figure 6.12: Mean Jerk

	R	V	VF
Mean Jerk $\left[\frac{m}{s^3}\right]$	236.375	96.5122	94.8093

Table 6.9: Mean Jerk

	R-V	R-VF	V-VF
Jerk p**-value**	0.0023*	0.0020*	0.7853

*denotes significance

Table 6.10: Jerk Statistics

Statistical significance for mean jerk was found between the monocular RGB feedback mode (**R**) and the two 3D-mapping feedback modes (**V**, **VF**), see Table 6.10. The jerk is an indication of path smoothness. Finally, consider the results from the post-experiment questionnaires assessing task load and situational awareness. These results are shown below in Fig 6.13 and Table 6.11.

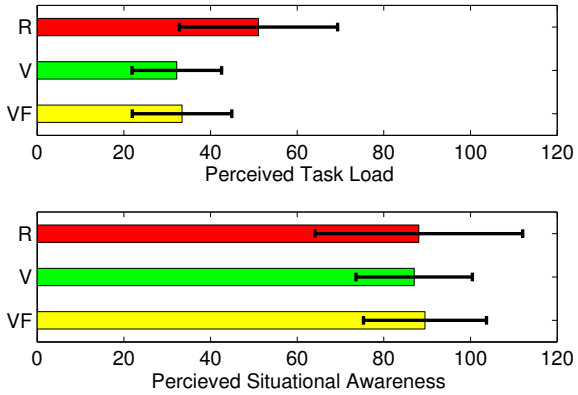


Figure 6.13: TLX and SART

	R	V	VF
NASA TLX out of 120	51.071	32.214	33.429
SART out of 120	88.071	87.000	89.500

Table 6.11: TLX and SART

	R-V	R-VF	V-VF
TLX p**-value**	0.0350	0.0515	0.8387
SART p**-value**	0.9195	0.8943	0.7407

*denotes significance

Table 6.12: TLX and SART Statistics

Statistical significance was found for neither post-experiment situational awareness (SART) nor perceived task load (TLX) between any of the feedback modes, see Table 6.12.

In summary, the metrics for which feedback mode showed a significant effect are shown in Table 6.13. In all significant comparisons involving the guidance fixture mode (**VF**), (**VF**) always yielded better performance. In all significant comparisons between and (**V**) or (**R**), (**V**) always yielded better performance.

	R-V	**R-VF**	**V-VF**
Time	×	○	×
Collisions	●	●	●
Path Length	○	○	×
Jerk	●	●	×
TLX	○	×	×
SART	×	×	×

● denotes significance ○ denotes $p < 0.05$ × denotes lack of significance

Table 6.13: Statistical Comparison Summary

Holm-Bonferroni Correction

Because of the exploratory nature of this work, three separate experimental groups and six different metrics were examined. The result is a total of 18 different hypotheses considered from the data set. Thus a multiplicity problem arises, and statistical analysis must account for this in order to avoid Type I errors, i.e. falsely rejecting a null hypothesis.

To begin the Holm-Bonferroni correction, first consider the $i = 18$ different p-values from the two-sample t-tests analyzing the 18 null hypotheses (three experimental groups, six metrics). Then sort these p-values in ascending order in a list with corresponding null hypotheses:

$$p_1, p_2, ..., p_i \qquad n_1, n_2, ..., n_i$$

Now take the typical analysis significance level, $\alpha = 0.05$. Via the Holm-Bonferroni method, let $j \in [1, i]$ be the least value index such that the inequality below is satisfied:

$$p_j > \frac{\alpha}{i - j + 1} = \frac{0.05}{19 - j}$$

Then the null hypotheses $\{n_1, n_2, ..., n_j\}$ are rejected, while the remaining are not. Using this analysis method yields these statistical results in Table 6.14:

j	null hypothesis	p-value	$\frac{\alpha}{i-j+1} = \frac{0.05}{19-j}$
1	Collisions (R-VF)	8.459e-12	0.0028
2	Collisions (V-VF)	9.110e-7	0.0029
3	Collisions (R-V)	1.118e-5	0.0031
4	Jerk (R-VF)	0.0020	0.0033
5	Jerk(R-V)	0.0023	0.0036
6	Path Length (R-VF)	0.0212	0.0038
7	TLX (R-V)	0.0350	0.0042
8	Time (R-VF)	0.0373	0.0045
9	Path Length (R-V)	0.0499	0.0050
10	TLX (R-VF)	0.0515	0.0056
11	Time (R-V)	0.0757	0.0063
12	Path Length (V-VF)	0.5927	0.0071
13	Time (V-VF)	0.6129	0.0083
14	SART (V-VF)	0.7407	0.0100
15	Jerk (V-VF)	0.7853	0.0125
16	TLX (V-VF)	0.8387	0.0167
17	SART (R-VF)	0.8943	0.0250
18	SART (R-V)	0.9195	0.0500

Table 6.14: Holm-Bonferroni Correction

Chapter 7

CONCLUSIONS

7.1 Summary of Studies

7.1.1 Transparent Grasping of Objects

This work provided results for the use of 3DOF haptic feedback and one dimensional pretouch sensing to assist a teleoperator in exploring unknown, static environments in a telerobotic grasping task. It has particular application in obtaining geometries of a transparent object while preventing undesired collisions. The streaming point cloud data is augmented with pretouch proximity sensor information, which is used for both haptic rendering and refining the shape of the object without physical contact. Previous methods involved either making assumptions about the missing data from spatially neighboring information, or by requiring invasive contact sensing techniques [2]. In contrast, the methods presented here allow for maneuverable interrogation of an unseen object gathering real data without collisions. Theoretical analyses were performed to design the virtual fixtures suitable for pretouch sensing; experiments with an opaque cardboard box were conducted to verify the effectiveness of this approach, while successful teleoperated grasp of a transparent object was achieved.

By fusing haptic feedback and pretouch sensing, the augmented point cloud and virtual fixtures provide the teleoperator with critical geometrical information of the grasp target while simultaneously preventing the robot end effector from colliding with it. In designing teleoperation architectures with unknown non-moving objects, sensor range and end-effector positioning repeatability need to be considered. When done effectively, results are promising.

7.1.2 Non-Geometric Tracking Data and Depth Sensor Arrays

Here we presented a system for exploring dynamic objects and regions that cannot be well observed by cameras or long range-depth sensors due to space constraints, accessibility, or occlusions. Given a region to search, the teleoperator could use pretouch sensing to recover geometric information about an object as haptic feedback both encourages more expedient exploration and prevents collisions. As the search continues, explored locations are added to the OctoMap and displayed to the user. The results of this work showed that haptic feedback allowed the user to collect significantly more occupied voxels than when only utilizing visual feedback.

Indeed, the haptic feedback trials yielded less than half of the collisions that the non-haptic feedback trials did, and the user was able to consistently get closer to the object of interest, resulting in over twice the exploration coverage while inducing fewer collisions. While the haptic feedback encouraged the user to approach the object when far away, it also allowed the user to search more aggressively and focus on exploration, rather than having to search more conservatively in order to avoid collisions. The increased exploration of voxels was likely a direct result of heightened operator confidence while operating with haptic virtual fixtures; by spending more time close to the object, the object was more likely to be within the sensor's measurement range.

The effects of object speed and predictability still need to be explored in terms of integrating this method with real world application. Furthermore, tracking delay may be present and orientation information may be lacking, furthering the difficulty of general application.

7.1.3 Haptic Feedback and 3D Vision

In this work, we examined different feedback modalities in a basic teleoperation task: a valve turn. In addition to widely available 2D monocular visual feedback, depth informa-

tion conveyed in the form of a voxel-occupancy grid, as well as a manually placed haptic guidance fixtures were tested. It was of interest to explore the effects of these modifications to telemanipulation task performance in both quantitative and qualitative metrics. While 3D voxel mapping techniques have been shown to improve operator performance in navigation tasks, its effect in telemanipulation had not been quantified. Moreover, while the use of haptic guidance fixtures has been shown to positively influence predetermined trajectory following, we evaluated the effect of manually set/positioned guidance fixtures in a known telemanipulation task with real-time data. To validate and more importantly evaluate these methods, a user study was performed.

This study showed that varying the user feedback mode did not affect the metrics of:

- Perceived situational awareness

There were no detrimental effects from using 3D-mapping methods over monocular RGB streams in any of the metrics explored. Furthermore, the results of the user study show that even in simple telemanipulation tasks:

1. Haptic guidance virtual fixtures significantly reduce the number of unwanted collisions compared to visual feedback modes.

2. 3D visualization significantly reduces the number of collisions compared to 2D.

3. 3D visualization significantly improves path smoothness compared to 2D.

and that there are trends towards

- Haptic guidance virtual fixtures reducing completion time compared to visual feedback modes.

- 3D visualization reducing path length compared to 2D.

- 3D visualization reducing perceived task load compared to 2D.

This work shows that even in simple telemanipulation cases, 3D visualization and haptic guidance can provide performance improvements over conventional monocular RGB feedback. In designing operator interfaces for telemanipulation, effort should be made to equip remote devices with accurate and proper sensing capabilities to provide enhanced 3D visual feedback. In some cases, visual clutter may hinder the positive effects of providing 3D visualization. When such complexity and visual overload is an issue, another feedback pathway is needed. While the operator-side hardware setup is an additional burden, haptic guidance can be leveraged to assist human operators in real-time with similar if not better affect, as shown by this work.

7.2 Discussion of Studies

7.2.1 Transparent Grasping of Objects

Physical structures which are difficult to observe with optical sensors (e.g. glancing angles, reflective surfaces and transparent objects) pose difficulties in terms of perception. This can be particularly troublesome when the objects in question are high-value and delicate. In scenarios where autonomy is insufficient or ill-advised, telerobotics offers a distinct advantage: a human in-the-loop. This work focused on addressing the issue of transparent objects in the workspace.

Design Implications

Combining pretouch geometry sensors which are robust to optical transparency with haptic forbidden regions showed encouraging results towards safely (without collisions) exploring an unseen surface. The novel sensor device provided a means to gather explicit geometry data, while telerobotics and haptic virtual fixtures allowed for intelligently protecting the unseen surface. The data suggests that this architecture reduces the number of unwanted collisions and displacement of the unseen surface as compared to using teleoperation with

visual feedback alone. The resultant geometry data was much better in terms of plane-fitting results. These performance benefits come at the cost of reduced number of sensed data as well as a marginally increased task completion time, see Table 6.1. The short sensing range and haptic forbidden regions account for fewer sensed data points and the increased time. This work can be generalized to arbitrary missing geometry data, perhaps from occlusions or reflective material, and is encouraging to the implementation of pretouch sensor information and real-time haptic forbidden regions. The combination of robust sensor and haptic feedback enables a telerobotic work-flow for surface data collections while preventing unwanted collisions.

Research Implications

While the results of this study are promising and provide a simple approach to gathering new point cloud data in real-time from auxiliary sensors, several factors limit the applicability of this work to specific scenarios. The haptic virtual fixtures affect only a single point, namely the HIP. For collision avoidance, this method was amenable only under the assumption of a convex grasp object and no nearby neighboring objects. In other cases, the grasper itself may collide with the grasp target or nearby surfaces - again, imagine scanning with an end effector sensor the inside of a bowl versus the outside without prior knowledge of its curvature; exploring the inside is much more prone to contact.

In this work, several assumptions about robot kinematics and sensor noise are made. Namely, we assumed that the PR2 arm controller was reliable enough within the physical task space to maintain position tracking error within a finite bound amenable to our seashell sensor. This bound was empirically and heuristically determined for the PR2 arm, and was found to satisfy this assumption. Moreover, sensor noise from the seashell sensor was assumed to be negligible within the specified reliable range ([1,5] mm). With these assumptions, the method shows working results. However, if the kinematics of a manipulator are less reliable, sensing range and tracking error effects must be resolved.

Thus, while the seashell pretouch sensor can reliably detect surfaces in great detail within the sub-centimeter range, a larger sensing range would allow for surface detection from greater distances and provide a safer distance from which to explore and avoid contact. The small coverage area of the sensor furthermore increases the time needed to explore and register a transparent or unseen area. An array of sensors would ameliorate this concern. Finally, this approach deals with non-moving environments and objects, since the sensed points are non-moving as well - an object which moves would render the pretouch acquired points invalid.

7.2.2 Non-Geometric Tracking Data and Depth Sensor Arrays

RGB-D cameras can provide a dense array of optical and depth information. However, telerobotic tasks may involve information other than optical and depth data, and in the case that depth is important, RGB-D types of sensors can be bulky or otherwise incompatible with the task space. The previous work dealt with acquiring new point-cloud data from a non-moving object leveraging a pretouch type sensor. This work explored tasks where, again, geometry information of an object of interest is unavailable, as RGB-D cameras are not suitable. However, suppose the object is moving and can be generally located via other sensing means, e.g. thermal imaging, biomarkers, chemical sensors etc. The use of a maneuverable robot serial link manipulator in combination with end-effector mounted sensors can acquire desired geometry data otherwise unsuited for RGB-D, and haptic guidance virtual fixtures help maintain a reasonable sensing distance to the object and forbidden regions protect the potentially delicate target. Voxel volumes were used to reduce data storage needs and to encourage faster exploration. This combination yielded promising results.

Design Implications

The use of an IR TOF pretouch sensor array allowed for

- greater sensing range than the seashell pretouch sensor
- mulitple spatial sensing directions

While granularity is not as fine as with an RGB-D camera, the solution provides a means for higher bandwidth depth data collection in constrained and localized task spaces, those that would be unsuitable for an RGB-D sensor. The resolution of the desired voxel volume may ameliorate this drawback. The added spatial sensing directions are helpful in avoiding collisions surrounding the entire end effector, which was useful for interrogating surface geometries of a moving object.

The data shows that the two simultaneously provided types of haptic feedback improved operator performance. The guidance virtual fixture helped the operator maintain a closer tracking distance to the moving object (12.60 cm vs. 18.70 cm), and the forbidden region virtual fixture reduced collisions, see Table 6.2. These factors combined allowed for more data acquisition and better data fitting.

In environments where target objects move unpredictably, using a guidance fixture can help with tracking and maintaining exploration within sensing distance. Furthermore, with appropriately designed forbidden regions, the object will be safer from accidental collisions and data can thus more freely be acquired via teleoperation. In other words, forbidden regions allowed the operator to more aggressively explore unexplored regions.

Research Implications

This work provides a method for exploring unseen geometries of a tracked moving target. Haptic guidance and forbidden regions are combined with pretouch sensor arrays to improve operator performance. However, the object moved at about $0.5\frac{cm}{s}$ with randomized pauses, and in a relatively predictable path. Future work should address and evaluate faster and less predictable tasks. However, despite these assumptions, the haptic feedback presented

showed marked improvements when compared to visual feedback alone, suggesting that the task was not too simple.

Additionally, it was assumed that both location and orientation information about the object of interest were available, whereas some tracking systems provide location data only. In this case, the potential use of the iterative closest point algorithm can help to identify potential orientations between old and new sensor data [44]. Otherwise, a prior or ground-truth known overall shape of the object can be fit to the sensor data. A method of correcting for potential orientation changes is needed.

7.2.3 Haptic Feedback and 3D Vision

Technologies such as commodity RGB-D cameras and haptic interfaces provide inexpensive hardware means to adopt 3D visualization and haptic assistance modes in teleoperation architectures. While it may not be surprising that 3D voxel visualization can afford better teleoperator performance over 2D methods, or that haptic assistance can further improve telemanipulation, this study provides user study data from a real telerobotic implementation and quantifies the effects of each improvement in specific performance metrics.

In particular, despite the exploratory nature of and multiple comparisons in this work, it was shown that workspace voxelization based on real-time point cloud data could help reduce collisions and jerk over traditional 2D monocular display. Mode **VF** offered the addition of a user-placed haptic guidance fixture as well as 3D voxel visualization, and was significantly better than mode **R** in the same metrics. The addition of a user-placed haptic guidance fixture yielded significantly better collision avoidance compared to 3D voxel visualization alone.

If the experiment were reimplemented but tested for a single hypothesis at a time with the same results, significance would be found in other comparisons. Particularly, the data shows that real-time workspace voxelization can potentially provide better performance in

terms of path length and perceived task load over traditional 2D monocular feedback. Also, the addition of a manually placed haptic guidance fixture does not hinder to possible improvements in path length as compared to 2D monocular vision alone, and the data suggests that only with the addition of the haptic guidance fixture is the metric of completion time potentially improved, see Table 6.13.

Design and Real World Implications

The data suggests that, in designing a telemanipulation interface, it is beneficial to use RGB-D data over a RGB video stream alone. The teleoperator performance was shown only to improve with this modification. Improvements in collision avoidance, overall path length, jerk and task load can expand accessibility and safe operation for currently limited and highly trained teleoperated tasks – indeed, the novice users in this study realized appreciable gains in these metrics from 3D visualization. An example of a highly skilled teleoperated task is underwater telemanipulation, where users need to operate tools, manipulate valves or match cables underwater. The procedure, currently accomplished with only macro-scale imaging [78–80], is challenging and expensive due to the high skill level required [81]. Designing systems with small-scale depth imaging and voxel representation, as demonstrated in this work, can enhance the telemanipulation performance of less trained individuals, thus reducing the skill and subsequent operating costs to execute such operations.

Furthermore, when delicate critical structures are involved, a collision could be disastrous. For such scenarios, this study shows that the addition of haptic virtual fixtures can further enhance telemanipulation collision avoidance. One such application area is search and rescue (SAR) robotics, a case where safe control is essential in a hazardous, unstructured environment. Robots in this application, also known as response robots, can be used for incident prevention and support, and can be a useful tool for saving human lives and accelerating the search and rescue process. From data collected by search and rescue initiative ICARUS end users, it was a general consensus that in practical SAR applications, "the robots will always

need to be tele-operated [sic] for safety and legal reasons" [82]. These types of robots have been used in trials-by-fire, including response to the Chernobyl meltdown, the Fukushima Daiichi meltdown, the Sago mine disaster, terrorist attacks of September 11th, Hurricane Katrina and La Conchita Mudslide [83, 84].

Current response robots for SAR applications, however, operate under a minimal amount of autonomy and assistance modes. This may be due to the delicate nature of incorporating novel technology into high-risk operations, where safety and legal issues may arise. The complexity and unstructured context of many crisis and SAR missions also make such missions extremely technology-unfriendly [82]. In terms of current utilization, remote robot responders are used merely to obtain geographical information and to assist human responders – human responders are able to remotely navigate and access robot sensor data [82]. In this way, most current practical applications of rescue robots have been passively assisting human responders in situation assessment.

This user study helps to expand the scope and utility of telerobots in rescue situations. The data provides encouraging results towards integrating the use of 3D visualization and haptic assistance in telemanipulation. In fact, improvements in collisions avoidance, path length and jerk indicate that these telemanipulation interface improvements offer safer, more direct and smoother operation, all of which are encouraging for designing SAR application telerobots. This contribution provides results and limitations of the described user interface features. Such information is needed in order to adopt new technologies that eventually eliminate the need for and risk of in-the-field human responders, replaced instead with intelligently controlled telerobots with capabilities to safely and efficiently execute sensitive tasks.

Research Implications

The user study provides a baseline assessment of the effects of 3D voxel representations and user-placed haptic guidance virtual fixtures on operator performance in telemanipulation. This basic study provides a comparative benchmark for evaluating the effects of additional augmentations. For example, future work can compare the effects of visual clutter, additional user-placed haptic fixtures, different feedback modalities, and even automatically generated assistance modes with results from the baseline study.

Chapter 8

NEW APPLICATION AREAS OF HAPTICS AND HAPTIC TECHNOLOGY: SEMG CONTROL

Commodity electromyography (EMG) armbands are inexpensive tools that can be integrated into user interfaces. EMG involves measuring electrical activity associated with muscle movement, and thus presents a novel input channel in human computer interaction. However, using EMG interfaces could present a large cognitive load, as the EMG muscle movements may not translate well to the computer task. Haptic sensory feedback is a promising direction in terms of augmenting performance and learning of novel EMG interfaces. A novel computer game, which is EMG-controlled, is developed, and a haptic sensory feedback loop in addition to visual feedback modes are presented. To validate and more importantly evaluate the effectiveness of the haptic feedback, extensive user studies are conducted.

8.1 Electromyography (EMG)

Electromyography (EMG) involves the detection of electrical potential signals that are created by muscle cells when electrically or neurologically activated. These types of signals are most commonly analyzed and used to detect medical abnormalities or to analyze human biomechanics. There are two types of EMG, intramuscular EMG and surface EMG (sEMG). This work deals with the use of sEMG in particular, as commodity use has expanded to limb prosthesis control, robotics control, phone or computer control, flight control and exercise and fitness monitoring. The Thalmic Lab's Myo uses eight surface electrodes around the forearm to read particular muscle actuated gestures, i.e. fist, wave left, wave right, double tap, and fingers spread gestures. Typical signals are shown below in Fig 8.1

These poses can be detected and used to digitally trigger a software specified event. Ease of use and user performance need to be understood, particularly when the triggered software event is not intuitively related to the gesture itself. This work investigates the use of haptic feedback to aid in user performance of sEMG input devices.

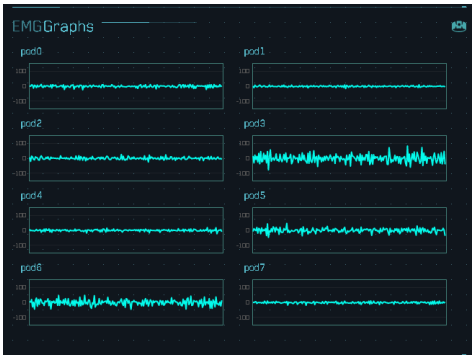


Figure 8.1: Typical Signals from Myo sEMG

8.2 Experimental Setup

A one-dimensional sEMG-controlled slingshot game was created as a testbed. The user is able to use an LCD monitor for visual feedback, the Myo sEMG for game control input, and the Sensable ® PHANToM^{TM}Omni for haptic feedback. This workflow is shown below in a block diagram as Fig 8.2, and a screen shot of the visual interface is shown in Fig 8.3.

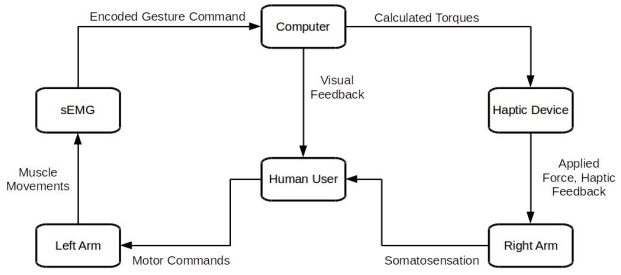


Figure 8.2: Slingshot Game Flow

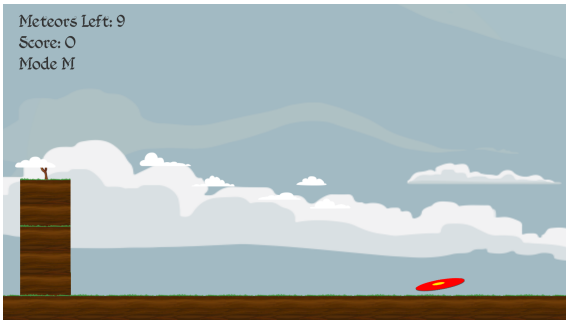


Figure 8.3: Slingshot Game Screen

The goal of the user is to control the slingshot game via the Myo sEMG interface to break a series of targets. Left and right wave motions stretch the slingshot to a desired distance, while a fist gesture releases a projectile. Scores and completion time are recorded.

8.3 User Study

A total of 51 subjects were recruited for this between user study. They were assigned to one of three different groups:

1. Visual Feedback Only

2. Haptic Feedback Only

3. Haptic and Visual Feedback

The haptic feedback provides a force proportional to the stretch of the slingshot in-game in the opposite direction. There were 17 subjects in each group. In total, there were 30 females and 21 males between the ages of 18 and 33, with an average age of 22.5 years. The subjects played their assigned game modes (one of either Haptic, Haptic + Visual, and Visual) for ten iterations of ten shots (100 shots total), including a warm up mode, as well as two randomly interspersed Visual only modes.

8.4 Results

During the studies, the users performed ten trials total with eight in their own feedback mode and two randomly dispersed visual feedback only trials. The visual only feedback mode serves as a baseline for integrating sEMG input. For each feedback category, the mean score completion time [s] are recorded. The scores are shown below in Tables 8.1

	Mean Score	Standard Dev
Haptic	4.622	1.205
Visual	5.622	1.419
Haptic&Visual	5.790	1.034

Table 8.1: Mean sEMG Game Scores

and the times here in Table 8.2.

	Mean Time [s]	Standard Dev
Haptic	111.22	37.97
Visual	117.62	37.12
Haptic&Visual	96.66	18.74

Table 8.2: Mean sEMG Game Times

A total of six pairwise comparisons were made (three experiment groups and two metrics), and the statistical p-values from two-sample t-tests are shown here in Table 8.3

	Score Comparison	Time Comparison
Haptic vs. Visual	0.0340	0.6233
Haptic vs. Haptic&Visual	0.0048	0.1658
Visual vs. Haptic&Visual	0.6957	0.0462

Table 8.3: sEMG Game Pairwise Comparison p-values

8.5 Discussion

From the score comparison data, the task is difficult to accomplish with haptic feedback alone. In the Haptic Only mode, the slingshot is obscured visually, effectively removing visual feedback prior to launch. Visual feedback is a critical factor in precisely controlling the slingshot stretch. However, the results also show that haptic feedback in combination with visual feedback has potential to reduce completion time while not affecting game score. However, because of the exploratory nature of this work and multiple comparisons, statistical significance cannot be claimed. Repeating the study evaluating the use of haptic augmentation of sEMG input for this game for reducing task time could yield significance.

Chapter 9

FUTURE DIRECTIONS: A FRAMEWORK FOR GENERAL SYNTHESIS OF HAPTIC VIRTUAL FIXTURES

The previously mentioned works provide tangible evidence and cases for building virtual fixtures with information from sensors auxiliary to RGB-D data. However, these implementations do not overcome the manual tuning and ad hoc approach to their creation. Instead, conjectures and formulations about the methodology need to be made. Because the search space for optimizing haptic virtual fixtures is quite complex and large, an evolutionary computation approach is proposed. With genetic algorithms, the search space is explored through crossover, mutation and selection of 'organisms' designed in such a matter that against a certain performance metric, the best individuals with desired traits survive and persist. The experimental arrangement and environmental setup for the genetic algorithm is critical to its success. Here, since operator input is difficult to model, assumptions about the operator's intentions are made. The genetic algorithms will be used to develop relations between relevant environmental and task parameters in efficient haptic virtual fixture creation.

9.1 Introduction

Current methodologies for implementing haptic assistance modes in teleoperation, such as forbidden regions and guidance fixtures, are largely done in an ad hoc manner. Oftentimes, the characteristics of the virtual fixtures can depend on the quality of sensor data. The problem therefore arises when new hardware, particularly sensing elements, provide new parameters that affect the haptic assistance modes - a new ad hoc tuning is needed. This limitation in the current state-of-the-art prevents rapid and cross platform deployment of even basic sensor-based haptic virtual fixtures, which could enhance and improve efficacy.

To address this, we are concerned with finding generalizable relations between basal sensor and virtual fixtures parameters for 3-DOF forbidden region haptic assistance and point-cloud based data. In order to do so, we propose a sampling approach whereby a space of potential point-cloud generating sensor parameters is sampled. At each of the sampled locations, which represents a particular set of sensor parameters, an optimal forbidden region haptic virtual fixture is found through simulation. From these results, the relationships between haptic virtual fixture and sensor parameters will be extrapolated.

9.1.1 Assumptions

For this work, we are particularly interested in a special subset of potential sensors, workspace, task type and virtual fixture methods that could be involved in a telerobotic system. Extrapolating the impact of these different system parameters on virtual fixture characteristics, within this particular subset, provides a stepping stone towards general synthesis of haptic virtual fixtures. A major assumption from this viewpoint is that dimensions within the factors of sensors, workspace, and task type can inform the creation of an effective virtual fixture.

Sensor

First consider the types of sensors that may be included in a robotic system. There are myriad of measurands that are of interest, including biological markers, physical geometries, temperatures, chemical levels etc. Oftentimes a physical location is associated with a sensor measurement (e.g. pixels of an RGB image), and when dealing with augmenting the robot's motion and its interaction with the environment, physical location of sensed relevant measurands are pertinent.

For the purpose of this work, consider sensors which provide either a binary, digital or analog sensor reading. Further, assume that this reading can provide the encoded measure-

ment and be registered with a physical point in 3D space, or an array of points in 3D space. Examples include a single reading chemical sensor, whose measurement can extrapolate the concentration of a particular chemical at the sensor location, or a 2D laser scanner which provides ranging information along one physical dimension. These two examples provide very direct methods for physical registration of the sensor data (in the first case, the sensor location itself provided location, while the second case provided depth information relative to the sensor location). When a sensor provides physical registration of the measurement with insufficient spatial dimensionality (e.g. an RGB image or thermal image), raycasting and voxelization can be used to achieve 3D registered sensor data [85,86]. To summarize, we are concerned with sensors, that with specificity in 3D localized point location, can provide measurements. We will henceforth term these sensors as point-based sensors. With this in mind, here are the factors regarding point-based sensors that we have identified which could be critical to haptic virtual fixtures:

- noise levels
- bias
- sensor drift
- latency
- spatial resolution
- temporal frequency

Teleoperated Task and Human Operator

A simple teleoperated task that may be amenable to haptic assistance involves both obstacle avoidance and navigation. Suppose that an object is obstructing direct motion to a goal configuration. In the simplest case, the object is not moving, but its surface geometry is observed via sensor data. It is the teleoperator's goal to control the end effector around the object to the goal location.

Consider that a teleoperator's motions and intentions greatly influence the effectiveness of haptic virtual fixtures. For example, a sophisticatedly trained teleoperator in a repeatable task may not elicit any useful feedback from a forbidden region or guidance haptic virtual fixture simply because of user experience. In fact, when the human operator is so sure of his or her decision, a haptic virtual fixture which opposes the desired motion of the operator can be detrimental. Thus, it would be helpful to simulate user input signals, yet modeling human input, experience and intent is an arduous and complex task.

To address this, consider the contention that haptic virtual fixtures are best suited for situations in which the operator is under duress, for example from task overload, inexperience or lack of proper visual feedback. In such a situation, the user would benefit and potentially rely heavily from intelligent haptic assistance. To that end, make the assumption for this that the user imparts no motion commands; rather the operator input acts solely as a point mass. The user is still able to monitor the procedure, and in the case of unforeseen events can intervene and take immediate control.

Haptic Virtual Fixtures

Now consider potential variables in virtual fixture design. Again, we are interested in point cloud haptic interaction, and thus consider the proxy method for streaming point clouds [39]. The variables to consider include:

- proxy radii, r_1, r_2, r_3
- proxy step sizes
- spring constant, forbidden region and guidance

9.2 Genetic Algorithm

The development, adaptation and evolution of natural organisms in changing environments gives inspiration to several types of heuristic search mechanisms classified as evolutionary

computation. Genetic algorithms and application towards real-world optimization problems are of particular interest. Genetic algorithms were invented by John Holland, as he studied the process of biological adaption and sought to incorporate these natural mechanisms into computer systems. Holland proposed this method in [87], a formulation coined as a simple genetic algorithm. In this, a population of organisms are described by their chromosomes encoded in binary representation. Environment pressures help inform selection, crossover and mutation in a simulation of the population generation after generation, with organisms with highest payoff or performance having higher chances of survival and reproduction.

Because of the versatility and application to complex, multimodal problems, genetic algorithms have given rise to unique and innovative solutions in game theory [88], mathematical objects such as cellular automatons [89], predicting DNA protein structures [90] [91], ecological modeling [92], sensor fusion [93] [94], image processing [95] and robotics [96] [97] to name a few.

Challenging computer problems can require findings solutions within a complex and vast search space, such is the case with optimal haptic virtual fixture generation. Sometimes, the ideal solution does not have a definite form. To that end, an ideal approach would perform a guided search instead of a brute force approach, and parallelization can further improve efficiency. Genetic algorithms possess several characteristics which make them amenable to computational problems [98]:

- adaptive
- innovative
- complex
- parallel application

A basic genetic algorithm process can be represented in pseudo-code as shown below:

Algorithm 4 Basic Genetic Algorithm Outline

1: The objective function to optimize is given as f
2: Initialize a population X_0, of n random bit strings of length l
3: **for** each generation m **do**
4: **for** each $x \in X_m$ **do**
5: compute $f(x)$
6: **end for**
7: **while** $|X_{m+1}| < n$ **do**
8: Select x_1, x_2 for crossover with probability of selecting x increasing with $f(x)$
9: Crossover x_1, x_2 with probability p_c at crossover points to form x'_1, x'_2
10: Mutate x'_1, x'_2 at each bit with probability p_m
11: Append x'_1, x'_2 to X_{m+1}
12: **end while**
13: **end for**
14: **if** Stopping criteria are not met with latest generation **then**
15: Repeat 3-13
16: **end if**

First sampling the relevant parameter space, in a Monte Carlo fashion, sets up an array of optimization problems. At each parameter space location, we propose simulating the above described haptic interaction task and evaluating a genetic algorithm to obtain optimal virtual fixture parameters. Once optimized for each parameter location, trends and relations between sensor and task parameters and haptic virtual fixture variables can be extrapolated.

9.2.1 Anticipated Outcome

These theoretical contributions and methods will provide initial work towards codifying the integration and future work with telerobotic systems involving multi-modal sensor haptic virtual fixtures. At the present, no such generalizable strategy which can be deployed to arbitrary bilateral teleoperation tasks and setups exists.

BIBLIOGRAPHY

[1] A. Maimone and H. Fuchs, "Reducing interference between multiple structured light depth sensors using motion," in *Virtual Reality Short Papers and Posters (VRW), 2012 IEEE*, March 2012, pp. 51–54.

[2] D. Pai, "Acme, a telerobotic measurement facility for reality-based modelling on the internet," in *IROS Workshop on Robots on the Web*, 1998.

[3] G. Kahn, P. Sujan, S. Patil, S. Bopardikar, J. Ryde, K. Goldberg, and P. Abbeel, "Active exploration using trajectory optimization for robotic grasping in the presence of occlusions," pp. 4783–4790, 2015.

[4] J. Aleotti, D. L. Rizzini, and S. Caselli, "Perception and grasping of object parts from active robot exploration," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 401–425, 2014.

[5] A. Bettini, S. Lang, A. Okamura, and G. Hager, "Vision assisted control for manipulation using virtual fixtures," in *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 2, 2001, pp. 1171–1176 vol.2.

[6] F. Wang and S. Payandeh, "A study of hybrid virtual fixtures in assistive path following problems," in *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on*, Nov 2012, pp. 1–2.

[7] M. Mast, M. panl, G. Arbeiter, V. tancl, Z. Materna, F. Weisshardt, M. Burmester, P. Smr, and B. Graf, "Teleoperation of domestic service robots: Effects of global 3d environment maps in the user interface on operators cognitive and performance metrics," in *Social Robotics*, ser. Lecture Notes in Computer Science, G. Herrmann, M. Pearson, A. Lenz, P. Bremner, A. Spiers, and U. Leonards, Eds. Springer International Publishing, 2013, vol. 8239, pp. 392–401.

[8] T. B. Sheridan, *Telerobotics, automation, and human supervisory con trol*. MIT press, 1992.

[9] D.-S. Kwon, J.-H. Ryu, P.-M. Lee, and S.-W. Hong, "Design of a teleoperation controller for an underwater manipulator," in *Robotics and Automation (ICRA), 2000 IEEE International Conference on*, vol. 4. IEEE, 2000, pp. 3114–3119.

[10] D. Hainsworth, "Teleoperation user interfaces for mining robotics," *Autonomous Robots*, vol. 11, no. 1, pp. 19–28, 2001.

[11] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.

[12] H. H. King, B. Hannaford, K.-W. Kwok, G.-Z. Yang, P. Griffiths, A. Okamura, I. Farkhatdinov, J.-H. Ryu, G. Sankaranarayanan, V. Arikatla *et al.*, "Plugfest 2009: Global interoperability in telerobotics and telemedicine," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1733–1738.

[13] A. Sekmen, A. B. Koku, and S. Zein-Sabatto, "Human robot interaction via cellular phones," in *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, vol. 4. IEEE, 2003, pp. 3937–3942.

[14] G. Kamsickas, "Future combat systems (fcs) concept and technology development (ctd) phaseunmanned combat demonstrationfinal report," *Boeing Company, Seattle, WA, Tech. Rep*, 2003.

[15] T. Fong and C. Thorpe, "Vehicle teleoperation interfaces," *Autonomous robots*, vol. 11, no. 1, pp. 9–18, 2001.

[16] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich, "Common metrics for human-robot interaction," in *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*. ACM, 2006, pp. 33–40.

[17] J. Y. Chen, E. C. Haas, and M. J. Barnes, "Human performance issues and user interface design for teleoperated robots," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 6, pp. 1231–1245, 2007.

[18] J. H. Park and T. B. Sheridan, "Supervisory teleoperation control using computer graphics," in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*. IEEE, 1991, pp. 493–498.

[19] Y. Tsumaki, Y. Hoshi, H. Naruse, and M. Uchiyama, "Virtual reality based teleoperation which tolerates geometrical modeling errors," in *Intelligent Robots and Systems (IROS), 1996 IEEE/RSJ International Conference on*, vol. 3. IEEE, 1996, pp. 1023–1030.

[20] K. Salisbury, F. Conti, and F. Barbagli, "Haptic rendering: introductory concepts," *Computer Graphics and Applications, IEEE*, vol. 24, no. 2, pp. 24–32, 2004.

[21] C. B. Zilles and J. K. Salisbury, "A constraint-based god-object method for haptic display," in *Intelligent Robots and Systems 95.'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on*, vol. 3. IEEE, 1995, pp. 146–151.

[22] D. C. Ruspini, K. Kolarov, and O. Khatib, "The haptic display of complex graphical environments."

[23] M. Minsky, O.-y. Ming, O. Steele, F. P. Brooks Jr, and M. Behensky, "Feeling and seeing: issues in force display," *ACM SIGGRAPH Computer Graphics*, vol. 24, no. 2, pp. 235–241, 1990.

[24] G. Cirio, M. Marchal, M. A. Otaduy, and A. Lécuyer, "Six-dof haptic interaction with fluids, solids, and their transitions," in *World Haptics Conference (WHC), 2013*. IEEE, 2013, pp. 157–162.

[25] W. A. McNeely, K. D. Puterbaugh, and J. J. Troy, "Six degree-of-freedom haptic rendering using voxel sampling," in *ACM SIGGRAPH 2005 Courses*. ACM, 2005, p. 42.

[26] E. Ruffaldi, D. Morris, F. Barbagli, K. Salisbury, and M. Bergamasco, "Voxel-based haptic rendering using implicit sphere trees," in *Haptic interfaces for virtual environment and teleoperator systems, 2008. haptics 2008. symposium on*. IEEE, 2008, pp. 319–325.

[27] Y. Li, M. Tang, S. Zhang, and Y. J. Kim, "Six-degree-of-freedom haptic rendering using translational and generalized penetration depth computation," in *World Haptics Conference (WHC), 2013*. IEEE, 2013, pp. 289–294.

[28] M. Ortega, S. Redon, and S. Coquillart, "A six degree-of-freedom god-object method for haptic display of rigid bodies with surface properties," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 3, pp. 458–469, 2007.

[29] F. Ryden and H. Chizeck, "A method for constraint-based six degree-of-freedom haptic interaction with streaming point clouds," in *Robotics and Automation (ICRA) 2013 IEEE International Conference on*, May 2013, pp. 2353–2359.

[30] A. Bettini, S. Lang, A. Okamura, and G. Hager, "Vision assisted control for manipulation using virtual fixtures: experiments at macro and micro scales," in *Robotics and Automation (ICRA), 2002 IEEE International Conference on*, vol. 4, 2002, pp. 3354–3361 vol.4.

[31] J. J. Abbott and A. M. Okamura, "Virtual fixture architectures for telemanipulation," in *Robotics and Automation (ICRA), 2003 IEEE International Conference on*, vol. 2. IEEE, 2003, pp. 2798–2805.

[32] R. Kumar, A. Kapoor, and R. H. Taylor, "Preliminary experiments in robot/human cooperative microinjection," in *Intelligent Robots and Systems (IROS), 2003 IEEE/RSJ International Conference on*, vol. 4. IEEE, 2003, pp. 3186–3191.

[33] M. Li and R. H. Taylor, "Optimum robot control for 3d virtual fixture in constrained ent surgery," in *Medical Image Computing and Computer-Assisted Intervention-MICCAI 2003*. Springer, 2003, pp. 165–172.

[34] J. W. Park, J. Choi, Y. Park, and K. Sun, "Haptic virtual fixture for robotic cardiac catheter navigation," *Artificial organs*, vol. 35, no. 11, pp. 1127–1131, 2011.

[35] T. L. Gibo, L. N. Verner, D. D. Yuh, and A. M. Okamura, "Design considerations and human-machine performance of moving virtual fixtures," in *Robotics and Automation (ICRA) IEEE International Conference on*. IEEE, 2009, pp. 671–676.

[36] J. Ren, R. V. Patel, K. A. McIsaac, G. Guiraudon, and T. M. Peters, "Dynamic 3-d virtual fixtures for minimally invasive beating heart procedures," *Medical Imaging, IEEE Transactions on*, vol. 27, no. 8, pp. 1061–1070, 2008.

[37] N. V. Navkar, Z. Deng, D. J. Shah, K. E. Bekris, and N. V. Tsekos, "Visual and force-feedback guidance for robot-assisted interventions in the beating heart with real-time mri," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 689–694.

[38] F. Ryden, S. Kosari, and H. Chizeck, "Proxy method for fast haptic rendering from time varying point clouds," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, 2011, pp. 2614–2619.

[39] F. Ryden and H. Chizeck, "Forbidden-region virtual fixtures from streaming point clouds: Remotely touching and protecting a beating heart," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, Oct 2012, pp. 3308–3313.

[40] C. W. Nielsen, M. A. Goodrich, and R. W. Ricks, "Ecological interfaces for improving mobile robot teleoperation," *Robotics, IEEE Transactions on*, vol. 23, no. 5, pp. 927–941, 2007.

[41] D. Labonte, P. Boissy, and F. Michaud, "Comparative analysis of 3-d robot teleoperation interfaces with novice users," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 40, no. 5, pp. 1331–1342, 2010.

[42] D. J. Bruemmer, D. A. Few, R. L. Boring, J. L. Marble, M. C. Walton, and C. W. Nielsen, "Shared understanding for collaborative control," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 35, no. 4, pp. 494–504, 2005.

[43] B. Willaert, J. Bohg, H. Van Brussel, and G. Niemeyer, "Towards multi-dof model mediated teleoperation: using vision to augment feedback," in *Haptic Audio Visual Environments and Games (HAVE), 2012 IEEE International Workshop on.* IEEE, 2012, pp. 25–31.

[44] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*, 2011, pp. 127–136.

[45] D. A. Butler, S. Izadi, O. Hilliges, D. Molyneaux, S. Hodges, and D. Kim, "Shake'n'sense: Reducing interference for overlapping structured light depth cameras," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '12. New York, NY, USA: ACM, 2012, pp. 1933–1936. [Online]. Available: http://doi.acm.org/10.1145/2207676.2208335

[46] N. Senin, B. M. Colosimo, and M. Pacella, "Point set augmentation through fitting for enhanced icp registration of point clouds in multisensor coordinate metrology," *Robot. Comput.-Integr. Manuf.*, vol. 29, no. 1, pp. 39–52, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.rcim.2012.07.003

[47] I. Lysenkov, V. Eruhimov, and G. Bradski, "Recognition and pose estimation of rigid transparent objects with a kinect sensor," in *in Proc. of Robotics: Science and Systems*, 2012.

[48] U. Klank, D. Carton, and M. Beetz, "Transparent object detection and reconstruction on a mobile platform," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 5971–5978.

[49] L.-T. Jiang and J. Smith, "Seashell effect pretouch sensing for robotic grasping," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, 2012, pp. 2851–2858.

[50] B. Mayton, L. LeGrand, and J. Smith, "An electric field pretouch system for grasping and co-manipulation," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, pp. 831–838.

[51] F. Ryden and H. Chizeck, "A proxy method for real-time 3-dof haptic rendering of streaming point cloud data," *Haptics, IEEE Transactions on*, vol. 6, no. 3, pp. 257–267, 2013.

[52] I. Ihrke, K. N. Kutulakos, H. P. A. Lensch, M. Magnor, and W. Heidrich, "State of the art in transparent and specular object reconstruction," 2008.

[53] F. Mriaudeau, R. Rantoson, D. Fofi, and C. Stolz, "Review and comparison of non-conventional imaging systems for three-dimensional digitization of transparent objects," *Journal of Electronic Imaging*, vol. 21, no. 2, pp. 021 105–1–021 105–5, 2012. [Online]. Available: http://dx.doi.org/10.1117/1.JEI.21.2.021105

[54] A. Petrovskaya and O. Khatib, "Global localization of objects via touch," *Robotics, IEEE Transactions on*, vol. 27, no. 3, pp. 569–585, June 2011.

[55] N. Gorges, S. Navarro, and H. Worn, "Haptic object recognition using statistical point cloud features," in *Advanced Robotics (ICAR), 2011 15th International Conference on*, 2011, pp. 15–20.

[56] K. Hsiao, S. Chitta, M. Ciocarlie, and E. Jones, "Contact-reactive grasping of objects with partial shape information," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, Oct 2010, pp. 1228–1235.

[57] L.-T. Jiang and J. Smith, "A unified framework for grasping and shape acquisition via pretouch sensing," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, May 2013, pp. 999–1005.

[58] A. Leeper, K. Hsiao, M. Ciocarlie, I. Sucan, and K. Salisbury, "Methods for collision-free arm teleoperation in clutter using constraints from 3d sensor data," in *IEEE Intl. Conf. on Humanoid Robots*, Atlanta, GA, 10/2013 2013.

[59] D. Katz, J. Kenney, and O. Brock, "How can robots succeed in unstructured environments?" 2008.

[60] A. Peon and D. Prattichizzo, "Reaction times to constraint violation in haptics: comparing vibration, visual and audio stimuli," in *World Haptics Conference (WHC), 2013*, April 2013, pp. 657–661.

[61] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison *et al.*, "Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera," in *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 2011, pp. 559–568.

[62] A. Petrovskaya, O. Khatib, S. Thrun, and A. Y. Ng, "Bayesian estimation for autonomous object manipulation based on tactile sensors," in *Robotics and Automation (ICRA), 2006 IEEE International Conference on*. IEEE, 2006, pp. 707–714.

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

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

[65] Z. Jiang, Y. Liu, H. Liu, and J. Zou, "Flexible virtual fixture enhanced by vision and haptics for unstructured environment teleoperation," in *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, Dec 2013, pp. 2643–2648.

[66] A. Saxena, L. Wong, M. Quigley, and A. Y. Ng, "A vision-based system for grasping novel objects in cluttered environments," in *Robotics Research*. Springer, 2011, pp. 337–348.

[67] C. Goldfeder, M. Ciocarlie, J. Peretzman, H. Dang, and P. K. Allen, "Data-driven grasping with partial sensor data," in *Intelligent Robots and Systems (IROS), 2009 IEEE/RSJ International Conference on*. IEEE, 2009, pp. 1278–1283.

[68] D. Rao, Q. V. Le, T. Phoka, M. Quigley, A. Sudsang, and A. Y. Ng, "Grasping novel objects with depth segmentation," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 2578–2585.

[69] A. Hornung, K. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: an efficient probabilistic 3d mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013. [Online]. Available: http://dx.doi.org/10.1007/s10514-012-9321-0

[70] A. M. Okamura, "Haptic feedback in robot-assisted minimally invasive surgery," *Current opinion in urology*, vol. 19, no. 1, p. 102, 2009.

[71] L. Rosenberg, "Virtual fixtures: Perceptual tools for telerobotic manipulation," in *Virtual Reality Annual International Symposium, 1993., 1993 IEEE*, Sep 1993, pp. 76–82.

[72] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, 2012. [Online]. Available: http://www.mdpi.com/1424-8220/12/2/1437

[73] P. Marayong, M. Li, A. Okamura, and G. Hager, "Spatial motion constraints: theory and demonstrations for robot guidance using virtual fixtures," in *Robotics and Automation (ICRA), 2003 IEEE International Conference on*, vol. 2, Sept 2003, pp. 1954–1959 vol.2.

[74] J. Abbott, P. Marayong, and A. Okamura, "Haptic virtual fixtures for robot-assisted manipulation," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, S. Thrun, R. Brooks, and H. Durrant-Whyte, Eds. Springer Berlin Heidelberg, 2007, vol. 28, pp. 49–64.

[75] F. Rydén, H. J. Chizeck, S. N. Kosari, H. King, and B. Hannaford, "Using kinect and a haptic interface for implementation of real-time virtual fixtures," in *Proceedings of the 2nd Workshop on RGB-D: Advanced Reasoning with Depth Cameras (in conjunction with RSS 2011)*, 2011.

[76] *Proximity and ambient light sensing (ALS) module*, STMicroelectronics, 8 2014, rev. 6.

[77] C. M. Jones and S. D. Healy, "Differences in cue use and spatial memory in men and women." *Proc Biol Sci*, vol. 273, no. 1598, pp. 2241–7, 2006. [Online]. Available: http://www.biomedsearch.com/nih/Differences-in-cue-use-spatial/16901845.html

[78] L. Susperregi, J. M. Martínez-Otzeta, A. Ansuategui, A. Ibarguren, and B. Sierra, "Rgb-d, laser and thermal sensor fusion for people following in a mobile robot," *International Journal of Advanced Robotic Systems*, vol. 10, 2013.

[79] A. Kim and R. M. Eustice, "Real-time visual slam for autonomous underwater hull inspection using visual saliency," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 719–733, 2013.

[80] F. S. Hover, R. M. Eustice, A. Kim, B. Englot, H. Johannsson, M. Kaess, and J. J. Leonard, "Advanced perception, navigation and planning for autonomous in-water ship hull inspection," *The International Journal of Robotics Research*, vol. 31, no. 12, pp. 1445–1464, 2012.

[81] F. Rydén, A. Stewart, and H. J. Chizeck, "Advanced telerobotic underwater manipulation using virtual fixtures and haptic rendering," in *2013 OCEANS-San Diego*. IEEE, 2013, pp. 1–8.

[82] D. Doroftei, A. Matos, and G. de Cubber, "Designing search and rescue robots towards realistic user requirements," in *Applied Mechanics and Materials*, vol. 658. Trans Tech Publ, 2014, pp. 612–617.

[83] A. Birk, S. Schwertfeger, and K. Pathak, "A networking framework for teleoperation in safety, security, and rescue robotics," *Wireless Communications, IEEE*, vol. 16, no. 1, pp. 6–13, 2009.

[84] K. D. Katyal, C. Y. Brown, S. A. Hechtman, M. P. Para, T. G. McGee, K. C. Wolfe, R. J. Murphy, M. D. Kutzer, E. W. Tunstel, M. P. McLoughlin *et al.*, "Approaches to robotic teleoperation in a disaster scenario: From supervised autonomy to direct control," in *Intelligent Robots and Systems (IROS), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 1874–1881.

[85] S. Vidas, P. Moghadam, and M. Bosse, "3d thermal mapping of building interiors using an rgb-d and thermal camera," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2311–2318.

[86] J. van Baar, P. Beardsley, M. Pollefeys, and M. Gross, "Sensor fusion for depth estimation, including tof and thermal sensors," in *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*. IEEE, 2012, pp. 472–478.

[87] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press, 1975.

[88] R. Axelrod *et al.*, "The evolution of strategies in the iterated prisoners dilemma," *The dynamics of norms*, pp. 1–16, 1987.

[89] M. Mitchell, J. P. Crutchfield, R. Das *et al.*, "Evolving cellular automata with genetic algorithms: A review of recent work," in *Proceedings of the First International Conference on Evolutionary Computation and Its Applications (EvCA96)*. Moscow, 1996.

[90] C. B. Lucasius, S. Werten, A. van Aert, G. Kateman, and M. J. Blommers, "Conformational analysis of dna using genetic algorithms," in *International Conference on Parallel Problem Solving from Nature*. Springer, 1990, pp. 90–97.

[91] R. J. Parsons, S. Forrest, and C. Burks, "Genetic algorithms for dna sequence assembly." in *ISMB*, 1993, pp. 310–318.

[92] T. Jones and S. Forrest, "The ecology of echo peter t. hraber," *Artificial Life*, vol. 3, no. 3, pp. 165–190, 1997.

[93] R. R. Brooks, S. S. Iyengar, and J. Chen, "Automatic correlation and calibration of noisy sensor readings using elite genetic algorithms," *Artificial Intelligence*, vol. 84, no. 1, pp. 339–354, 1996.

[94] I. V. Maslov and I. Gertner, "Multi-sensor fusion: an evolutionary algorithm approach," *Information Fusion*, vol. 7, no. 3, pp. 304–330, 2006.

[95] M. Paulinas and A. Ušinskas, "A survey of genetic algorithms applications for image enhancement and segmentation," *Information Technology and control*, vol. 36, no. 3, 2015.

[96] K. Sims, "Evolving virtual creatures," in *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. ACM, 1994, pp. 15–22.

[97] Y. Hu and S. X. Yang, "A knowledge based genetic algorithm for path planning of a mobile robot," in *Robotics and Automation (ICRA), 2004 IEEE International Conference on*, vol. 5. IEEE, 2004, pp. 4350–4355.

[98] M. Mitchell, *An introduction to genetic algorithms*. MIT press, 1998.

VITA

Kevin was born in Seattle, Washington. In 2012 he earned his Bachelor of Science in Engineering and Mathematics from Trinity College. In 2015, he earned his Master of Science in Electrical Engineering from the University of Washington. During his time in higher education, he has been active in teaching and outreach to both primary and secondary school students.